Imperial College London

MEng Individual Project

Imperial College London

Department of Computing

# Network Effects and their limits in ride-hailing platforms

*Author:*
Remi Kaan Uzel

*Supervisors:*
Dr. Yves-Alexandre de Montjoye
Ali Farzanehfar

*Second Marker:*
Prof. Antonio Filieri

June 12, 2020

**Abstract**

Hotel companies not owning a single bed, taxi companies not owning a single car, the worlds' most diverse store not owning a single till; in the 21st century we are witnessing a rapid transformation of our way of life, greatly driven by such digital platforms. Ride hailing services in particular have been quite present in our lives, disrupting how we think about transport. As a platform business these services benefit from network effects: their value increases according to their number of users. Due to the existence of network effects, these services are believed to benefit a lot from first mover advantage. However, no one has studied whether there are limits in the network effect for ride hailing services, something crucial to competition watchdogs when deciding if they should let a service like this into the city. The current literature presents extensive analysis and modelling of various graph based methods, looking at the apparition of "small-world" or "scale-free" phenomena, as well as some structural properties. Although very useful, these models are often very abstract and rarely lend themselves to empirical falsification. Here we study the limits of NE in these services using agent based graphical models. We find the sensibilities to waiting/idle time to be decisive factors in the growth of these platforms. We validate our results using real trajectories from the New York City TLC dataset. We find that our model can accurately capture the RHPs growths both in terms of market-share and population. This work is an example of how data driven regulation can help enhance the competitive environment in the digital age.

**Acknowledgements**

I would like to express my sincerest thanks and gratitude to my supervisor Ali Farzanehfar, for his guidance, support, and for proposing this project.

# Contents

# Chapter 1

# Introduction

Since the late $18^{th}$ early $19^{th}$ centuries, the world has been seeing a surge in population of urban cities. This global scale urbanisation quickly turned into a huge economical growth perspective for multiple sectors. Today, with our the ever-increasing degree of digitisation in cities, the market for ride-hailing platforms (RHP) has exploded. In about a decade we have seen dozens of such companies (Uber, Lyft, Kapten, ViaVan, ...) grow to form a multi-billion dollar industry.

Uber-style platforms typically benefit from some form of network externality which results in a rich-get-richer dynamic (Matthew effect) [1], also called preferential attachment [2]. This occurs when an increase in the number of users of a product or service, directly increases its value. For example, additional users riding with a RHP will mean a richer market for drivers which incentivises them to join Uber. With this increase in drivers, waiting times are reduced which in turn attract additional riders to join Uber.

This property of a platform business is referred to as a network effect. Intuition suggests that for businesses like Uber, there could be some inherent limit to how much they can grow. For example, for Uber passengers, the difference between a waiting time of 30s and 1 min is not significant, and so, once Uber reaches a saturation such that waiting times are down to 1 minute, any growth in the number of drivers, will only have a marginal effect on the number of riders.

We propose here an agent based model that can simulate the network effects acting on various such platforms in a city, influenced by its topology. We also investigate the limits of these network effects, as well as the applicability of our model using open-source individual level trip data from multiple cities across the world.

The model takes its root from a particular domain of biology: immunology. Here we make the argument that riders and drivers behave analogously to the way cancer and immune cells (respectively) interact [3]. This comparison allows us to have 5 tweak-able parameters: the agents' sensibility to waiting time ($\mu_{waiting}$); the agents' sensibility to idle time ($\mu_{idle}$); the portion of agents that are riders (usually $\pm 95\%$); the calculation of the price surging coefficient ($\eta$); and the delay between platform's entrances to market.
This makes for a versatile model that can successfully reproduce real world growth, and can reproduce a situation where new users join RHPs according to a system resembling preferential attachment [4].

This work shines a much needed, data-driven light on competition between RHPs. Furthermore, the planned empirical validation of our model opens the door for competition authorities in cities to use this research to guide their decision making regarding the introduction of RHPs in their jurisdictions.

# Chapter 2

# Background

## 2.1 What are network effects

A network effect (NE) is an economical or business term that describes a situation where the value of a product or service increases with its number of users [5]. Classical examples of such that have been studied extensively include the citation pattern of scientific papers [6], the World Wide Web (WWW), or even the collaboration graph of movie actors [4]. The term "Network Effect" is used because these observed phenomena result from certain dynamical properties that are intrinsic to some particular types of networks.

These examples naturally exhibit NEs. A scientific paper which has been cited thousands of times is present in a substantial amount of other papers, gaining more exposure and increasing its odds of being cited again. Similarly, a website that has a certain reputation and many frequent users is more likely to be shared, increasing its odds of being linked on another page. Finally, a famous actor will naturally be offered more important roles in upcoming movies and therefore will increase his number of co-stars.

In pop-culture, "The rich get richer" is often the term used to describe such apparently unlimited exponential growth. In the literature, one can often find the terms: "Matthew Effect" [7] from the biblical verse *"For to every one who has will more be given, and he will have abundance..."*, "Cumulative Advantage" [8] or more recently "Preferential Attachment" [4].

## 2.2 Local vs global network effects

The examples laid out in the previous section are all instances of *global* (or *direct*) NEs. Typically, networks that profit from a direct NE have a single type of node, and the addition of a new node directly increases the value of each user. For example, when the telephone was introduced to the public, each additional phone allowed each owner to call an additional person, increasing their individual value for the service [9].

A typical example of a global NE could be the indexing of a search engine such as Google. As far as we know, Google's indexing grossly works by expanding its graph of known (indexed) websites by crawling through these [10]. What this means is that they go through their known websites in search for new links which they have yet to index. Naturally, having more indexed websites exponentially increases the number of discoverable links, and therefore introducing a global network effect.

Not all networks benefit directly from growth alone: those exceptions are called *local* (or *indirect*) NEs. Platforms that strongly profit from these are ride-sharing or ride-hailing platforms (RHP). If we model Uber as a network, it would be a bipartite network with two different types of nodes. Once the platform has a high number of rider nodes, the business becomes quite lucrative for drivers, increasing the number of driver nodes. This new increase in drivers cuts the waiting time for riders, which in turn increases the number of rider nodes.

Although not as obvious, local network effects are far from being the exception. Take any messaging app such as Facebook Messenger, WhatsApp or Snapchat. The overwhelming majority of their users are only aware of their own contacts, they only interact through the service with people they are directly connected to. In order words, if we were to draw an undirected graph of such a platform, each node would only be aware of its immediate neighbours - essentially creating minuscule local networks [11]. Agents have comparatively low degrees, and are unaware of the overall structure of the graph.

The case of Uber is an interesting one because it also represents a "second" layer of locality: although the increase in drivers is correlated in an increase in users, this is the case only in local geographically areas. RHPs benefit from these local effects, but this has yet to be rigorously studied. In addition to geography, there might also be waiting times that come into effect, which we will study in this paper. Although Uber is a global network, it consists of many smaller local clusters of bipartite graphs, within which we observe indirect NEs.

## 2.3 The implications for competition

Network effects have always had a considerable impact on financial markets. Historically, economists and competition watchdogs viewed NEs as a significant barrier to entry and protective of strong market positions. Now, more modern economic has recognised the various limits of NEs and negative consequences of platform growth [12]. Old literature suggested that networks exhibiting direct network effects (such as the telephone network) rapidly scaled to a monopoly. The widespread view was that once a particular platform reached a certain scale, it wasn't profitable to build a competing company [13]. As a result, the former would have significant market power, and competition enforcers tended to share this view.

As the understanding of multi-sided platforms advanced, so did the economic literature on both direct and indirect network effects [14]. Rapidly, economists recognised that the existence of NEs didn't necessarily ensure a strong first mover advantage: they could suffer from negative externalises just as much as positive ones. For example, as a search engine gains users it becomes attractive to advertisers, which are beneficial for the business. On the other hand, having more advertisers (or even users) has no effect on user demand [15]. In other cases, this impact might even be negative. Take paper or online newspapers, the increase in number of readers is correlated with the number of potential advertisers, but more advertising diminishes the value for the readers [16]. For social networks, which are an ever-expanding class of NE-prone platform, growth can even invite competition [17]. As the number of profiles gets large, the number of users who are trying to use the platform for disruptive or illegal activities also becomes problematic. This also naturally comes at a higher infrastructural, network and maintenance cost.

The explosion of digital platforms also brings its own set of problem for network effects. With social media, ride-hailing and dating platforms being free to use, there is no consumer "lock-in": the cost of moving to a competing service is often close to none. Additionally, while they may have a preference, users are not exactly limited to using a single platform (e.g. always using Uber instead of Black Cabs). This concept is called "multi-homing" and has been investigated by Rochet & Tirole [14].

All things considered, profiting from a network effect is not the one single secret to a successful business. Platforms on a multi-sided market must capture as many positive externalities, while also mitigating the negative effects that naturally come with growth. Such examples would be Facebook, that all-the-while competing with the (at the time $\approx$ 2004) giants Myspace and Friendster, enforced its strict policy for "real" profiles, requiring a valid college email addresses. It also rigorously enforced its terms of service, banning what it thought was obscene and nude content [18]. On the other hand at Myspace, news reports of minors who lied about their age and child sex predators who preyed on them caused public concern. Advertisers subsequently abandoned the platform and the site floundered [12].

With the development of online platforms benefiting from NE, some may seem to hold a monopoly. This seems to be the case in the mind of most people when they think about Uber, although in cities

like London or New York, the competition is considerable. On the other-hand, this competition might not exist in smaller cities such as Clermont Ferrand in France, leading to a real monopoly. Research has yet to determine the reason for this divide, and give a model for predicting when each situation is most likely to occur.

## 2.4 Graph-based models of network effects

At first, networks of complex topology have been explained using Erdős and Rényi's (ER) random graph theory [19]. Their model starts by generating $N$ vertices, and connects each pair of vertices with a probability $p$. From this setup, the probability that a randomly chosen vertex has $k$ edges follows a Poisson distribution $P(k) = e^{-\lambda}\frac{\lambda^k}{k!}$, with parameter:

$$\lambda = N\binom{N-1}{k}p^k(1-p)^{N-1-k}$$

Unfortunately, this research dates back to 1960, where data on large networks was nonexistent. Due to this, the theory couldn't be tested on real world data. Today, we have access to virtually unlimited data about all kinds of networks. This allows to empirically validate previously proposed models.

A more recent model proposed by Watts and Strogatz (WS) is the small-world model [20]. In this model, $N$ vertices are aligned to form a $1-D$ lattice, each vertex connected to two nearest other ones. An additional edge is then drawn to any other vertex with probability $p$. Because this can spontaneously generate "shortcuts" between otherwise far-apart vertices, this process decreases the average distance between each of them, leading to a small-world phenomenon [4] (also commonly known as "6 degrees of separation" [21]).

A common point between both the ER and WS models is that the probability of finding a highly connected node decreases exponentially with $k$, i.e. vertices with high degree are virtually non-existent. This conflicts with what we can observe in empirical data. Indeed, in (very) large networks, such highly connected vertices actually have a large chance of occurring, following a power-law tail. Barabási and Albert (BA) [4] explain that two generic aspects of real-world networks are missing in these two models.
First, both networks start with $N$ vertices, and no new one are ever added. The models simply attach edges between them using two different methods. In contrast, real-world networks are open, they form by the constant addition of new vertices from their environment. New actors join the industry quite often; the WWW grows exponentially over time with the addition of new websites; and research papers are constantly being published.
Secondly, random networks assume a random and uniform probability for two vertices to be connected. In the real-world we commonly see what BA coin as "Preferential Attachment". Newly published papers are much more likely to cite well known and peer-reviewed research rather than unknown works. This means that the probability of a new paper citing one that already has many citations (a higher degree node) is much higher than citing a paper with few citations (a low degree node). The variety of such existing examples illustrate that the way a new vertex links to existing ones is far from being uniform.

Barabási and Albert base their model on exactly these two features:

- Continuous growth (of the population size $N$),
- Preferential Attachment (of new nodes to high degree nodes).

Their network starts with a small number of vertices ($m_0$) and grows by the addition of a single vertex at each time step. New nodes are then connected by $m(\leq m_0)$ edges to $m$ other (different) vertices. Preferential attachment is then implementing by saying that each new node has a probability $\Pi$ to be linked to node $i$ depending on the degree $k_i$ of that vertex. That is:

$$\Pi(k_i) = \frac{k_i}{\sum_j k_j}$$

This network evolves into a scale-invariant state with the probability that a vertex has degree $k$ follows a power-law distribution with exponent $\gamma_{model} = 2.9 \pm 0.1$.
From comparing these models, we can see that BA's work related best to real-world networks. Thanks to this, we will take inspiration in his work to hopefully bring a new insight on NE profiting businesses and their limits.

## 2.5 Local network effect models

Current local NE models often focus on precise modelling of a complex network, by setting specific rules for individual clusters within them. Arun Sundararajan (AS) bases his model on the fact that individual vertices have no knowledge at all about the underlying structure of the network they are a part of [11]. The model is graph based and contains $N$ fixed vertices which represent agents. Each agent is associated to a set $G_i$, the neighbour set of vertex $i$. This can be seen as the friends, or contacts of the agent. If $j \in G_i$, we can say that $j$ is a neighbour of agent $i$. This creates an undirected graph of agents connected to their respective "close" acquaintances. A visualisation of such a neighbourhood can be seen in Figure 2.1. In addition, each vertex is initialised with an unchangeable parameter $\theta_i \in [0, 1]$ which influences what AS describes as its payoff value. At each time-step, agents make a binary decision $a_i$ to adopt or not a new abstract network good. The decisions are influenced by their neighbours and modifies their reward, or payoff $\pi_i$:

$$\pi_i(a_i, a_{i-1}, G_i, \theta_i) = a_i[u([\sum_{j \in G_i} a_j], \theta_i) - c]$$

where this payoff is dependent on a value function $u(x, \theta_i)$ that quantifies how many neighbours have adopted the good.

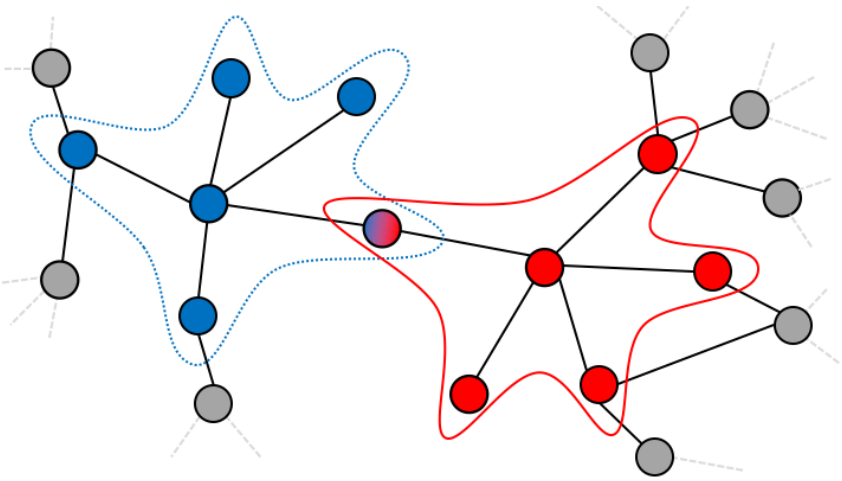


Figure 2.1: Two agents and their neighbourhood clusters $G_{blue}$ and $G_{red}$

AS's model successfully represents the fact that if your close family, friends or co-workers adopt something, you will to some extent be incentivised to do the same. His goal was to identify the overall adoption trend of the network, and how local adoption can influence the global result, taking a game theory-like approach.

One notable feature that AS describes in the "future work" section of his paper is that agents could adopt one of many incompatible goods, in a dynamical/evolutionary way. In our model we propose just that, where users are able to join one of $n$ RHPs based on a localised measure of the city's topology.

## 2.6 Lack of empirical validation

Throughout each of the models that we've seen this far, there has been no empirical work on competition and ride hailing platforms. Furthermore, BA verified their model on networks of at

most a few hundred-thousand vertices, and the same goes for AS and WS [4, 20, 11]. While this can seem like a substantial size, real world networks now easily go above tens of millions of nodes with a good amount of this data now being collected on astronomical scale.
Obviously, most of this real-world data is not made public, but some have used public datasets of millions of records to build alternative simulations.

R. Tachet et al. have built a "shareability" model of rides and tested their findings on empirical data from metropolises ranging from San Francisco, London and Singapore [2]. Their argument is that our increasingly connected urban areas drastically increase the number of unique trips (made in a personal car) that are similar enough to be merged into one car-pooling, all the while keeping the time delay to a minimum. This would reduce congestion as well as societal, environmental and economical cost in a city.
In their model, two trips are defined to be shareable if they would incur a sharing delay of no more than $\Delta$ minutes, relative to a single ride. The authors of the paper suggest a formula for the sharability ($S$) of rides for any given city:

$$S = 1 - \frac{1}{2L^3}(1 - e^{-L})(1 - (1 + 2L)e^{-2L}) \qquad \text{with} \quad L = \lambda\Delta^3 \frac{v^2(C)}{|\Omega(C)|}$$

with $v$ the average traffic velocity of the city, $\lambda$ the average rate at which taxi rides are available, and $\Omega$ the city's area.
What they were able to demonstrate was that in all the metropolises/megalopolises that they studied, sharability rapidly saturated to near 100% as both the average of trips/h or $L$ grew.

## 2.7 Contributions

Through this review we've seen how rich the existing literature is in terms of network modelling and theoretical results. Nonetheless, none have yet to study local NEs in the context of ride-hailing platforms using publicly available empirical data. The interesting feature of RHPs is that they naturally exhibit NEs, and it is not known whether or not this growth has a limit.

# Chapter 3

# The Model

## 3.1 The agent population model

Ride-hailing platforms (RHPs) can be modelled from two components: the riders and the drivers that make-up their user-base. Each agent has a specific set of criteria that they look for when choosing which platform to join. By considering only a handful of these criteria, our model allows us to analyse some key emergent properties of RHPs.

The dynamics of the two populations are modelled through a system of structured equations where $f_{rider}$ and $f_{driver}$ are the population functions and $R_{rider}$ and $R_{driver}$ model the net growth rates of rider and driver populations. We will look into each separately.
**[New paragraph is necessary here]**

- Mention that we are using a graph based model
- include a diagram of an RHP graph based model
- Mention that the two equations describe the rates of platform growth for riders and drivers

### 3.1.1 The rider agent intuition

$$R_{rider}(t, u_i) := \kappa_{driver}(t, u_i) - \mu_{waiting}\rho_{rider}(t, u_i) - \eta\rho_{rider}(t, u_i) \tag{3.1}$$

$$\rho_A(t, u) = \frac{f_A(t, u)}{f_{rider}(t, u) + f_{driver}(t, u)} \qquad \kappa_A(t, u_i) = \frac{\omega_A(t, u_i)}{\sum_{j\neq i}^{U} \omega_A(t, u_j)} \qquad A \in \{rider, driver\} \tag{3.2}$$

When a rider considers which RHP to join, they mainly consider a combination of three factors, which are defined in equations (3.1) and (3.2).

Riders typically have an inherent sensitivity to waiting time. Ask them to wait too long for a ride, and they are likely to check another app, or get to their destination differently. This is modelled through the $\mu_{waiting}$ parameter, and here we assume that this can be modelled as a single parameter for each platform (i.e. $\mu_{waiting}$ is the average sensibility of users from that platform). Almost more importantly, a RHP that has a higher number of riders w.r.t. their drivers will necessarily make their riders wait longer for each ride. This is modelled by the $\rho_{rider}(t, u_i)$ and is completely analogous to over-crowding. As this applies to all riders, the multiplicative scaling is justified.

Next, users of RHPs are naturally attracted by more "well-known" platforms. For many years, if you thought of ordering a taxi from your smart-phone, odds were you would think of Uber first. It has been shown that market-share is a correct description of all the factors that come into play when this phenomenon happens. In our model, we specifically take into account the drivers' market-share. This way, if a platform is currently booming amongst drivers, this would mean a relatively high driver-per-rider ratio, thus being attractive to riders for many reasons. This is all taken into account in the $\kappa_{driver}(t, u_i)$ term. Note that the way this function is being computed is completely identical to the probability of a new edge being attached to a particular node in the Barabási and Albert (BA) model [4]. This has been studied in detail and has been shown to be a

key element in exhibiting preferential attachment.

Last and definitely not least, the price of each ride is critical. Everywhere riders are unanimously attracted by low prices. This justifies for the last negative term of our model, $\eta\rho_{rider}(t, u_i)$. This term can be seen as a comparative choice of the prices offered by platform $i$ and all other platforms.

Note that all these terms share an equal weighting in the model. This is a simplifying assumption as the underlying relationship might account for other weightings or even non-linearities. As a first model, we choose not to make further assumptions about the underlying dynamics. Further, more complex models could segment rider and driver populations based on their price elasticity among a number of other metrics.

### 3.1.2 The driver agent intuition

$$R_{driver}(t, u_i) := \kappa_{rider}(t, u_i) - \mu_{idle}\rho_{driver}(t, u_i) + \eta\rho_{rider}(t, u_i) \tag{3.3}$$

Quite similarly to riders, when a driver chooses which RHP to work with, they consider a combination of factors which can define how attractive the RHP is to drivers.

Drivers will always choose to serve the ride which minimises their idle time, modelled through $\mu_{idle}$. This is once more considered to be the average sensitivity to idle time throughout a platform's drivers. This is also obviously negatively affected by over-crowding, and applies to all drivers. The multiplicative term $\rho_{driver}(t, u_i)$.

The next important factor that boosts a platforms' attractiveness is having a substantial pool of active riders. This is a positive effect and is represented by the $\kappa_{rider}(t, u_i)$ term. It describes what share of all riders the platform is able to offer the driver.

Lastly, and almost identically to the rider case, $\eta\rho_{rider}(t, u_i)$ involves the price of each ride a driver is able to charge. While this was a negative for riders, who aim to minimise cost, here drivers aim to maximise this same positive amount.

#### Rates as joining probabilities

$R_{rider}$ and $R_{driver}$, intuitively represent how attractive a particular RHP is to an agent at a specific time. We have seen previously how $R_{rider}$ and $R_{driver}$ were being calculated. Taking into account all the terms, and considering a negative attractiveness rate as being just as bad as a rate of 0, we find that $R_A \in [0, 1]$ for $A \in \{rider, driver\}$. This makes for an intuitive translation to probabilities. In the single platform case ($|U| = N = 1$) we consider this unique $R_A$ to be the probability that a newly generated agent of type $A$ joins the RHP, the alternative being joining no platform. In the multi-platform case ($|U| = N \geq 2$) each individual rate is understood as the probability that the newly generated agent joins the corresponding platform.

### 3.1.3 Reproduction of known behaviour

Ride-hailing platforms have particularly intuitive behaviours that our model is able to reproduce. First we will see that the exponential growth, typical of businesses that benefit from network effects exhibit, is well reproduced here. Then we will see how increasing the sensibility to idle and waiting time reduces the total population of their respectively affected agents (i.e. high waiting time sensibility in a platform implies a lower population of riders). Finally an increase in ride price should have a noticeable effect on the number of rider and driver agents, as this is negative for the former and positive for the latter.

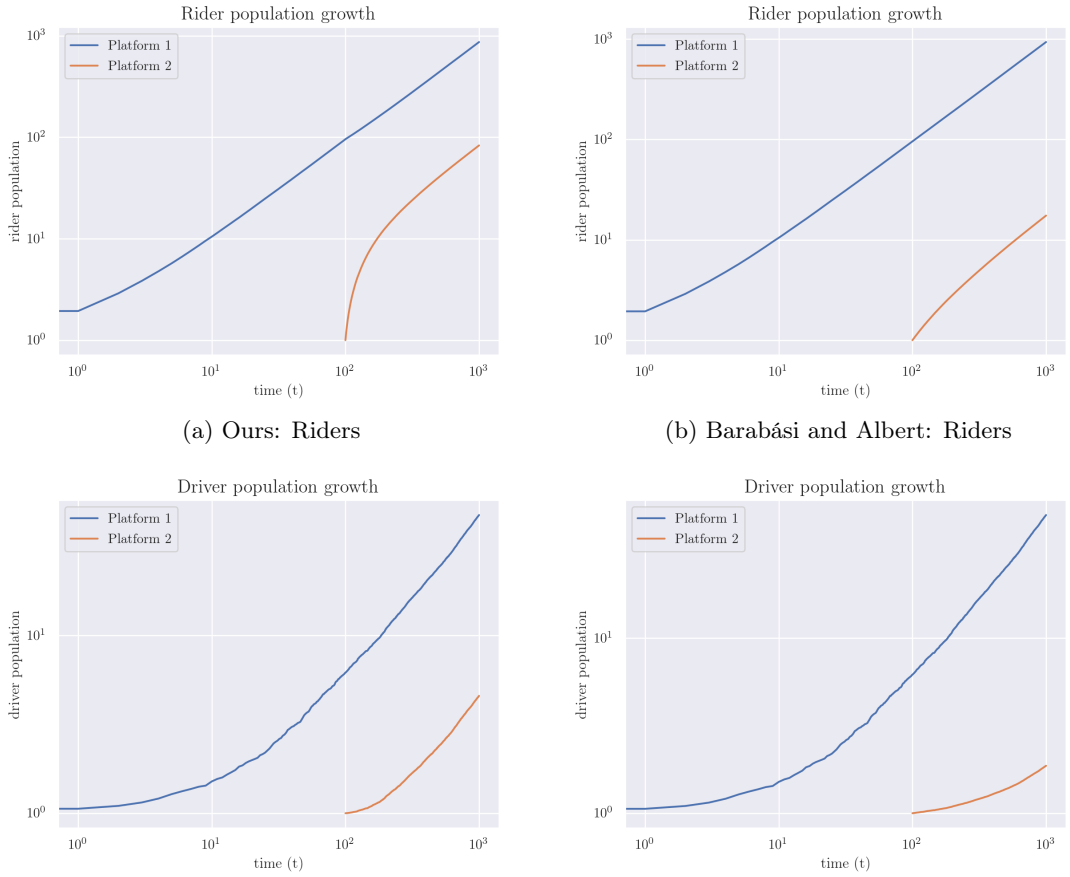
#### Exponential Growth

It has been well shown and documented that exponential growth, led by network effects, arises when two conditions are met: the population $N$ increases at every time step, and the probability

that a specific node in the network grows is proportional to that node's degree. Specifically, the probability that a new edge is drawn on node $i$ should be:

$$\Pi(i) = \frac{k_i}{\sum_{j \neq i} k_j} \qquad \text{where degree}(i) = k_i$$

Our simulation reproduces the first aspect naturally, with new agents being added to the environment at each time-step. Instead of modelling an entire graph, we consider the market-share of each platform to be equivalent to the degrees of a graph's node. Explicitly, one could model each platform as a graphs' cluster. The centroids would be the platforms, and each edge connecting it to a unique agent. The market-share of each platform is thus the degree of the centroid over the total number of edges in the graph.
Figure 3.1 (b) shows the population growth of our agents when their decision to join a platform is led solely by the platform's overall market-share, compared to figure 3.1 (a) where this decision is based on the rate questions from our model. The trends of both graphs are almost identical, with (b) having a slightly higher number of riders (+6%) and drivers (+10%).




(a) Ours: Riders

(b) Barabási and Albert: Riders

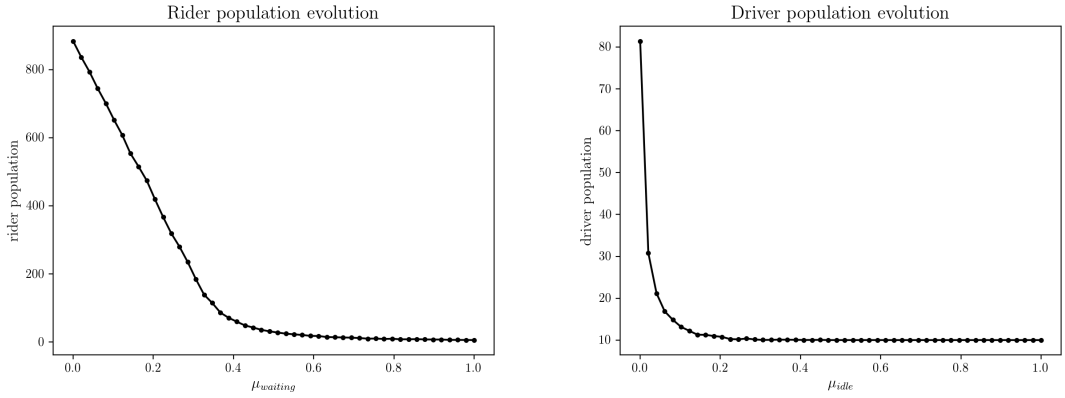
(c) Ours: Drivers

(d) Barabási and Albert: Drivers

Figure 3.1: Agent population growth comparison between our model and a purely market-share based model.

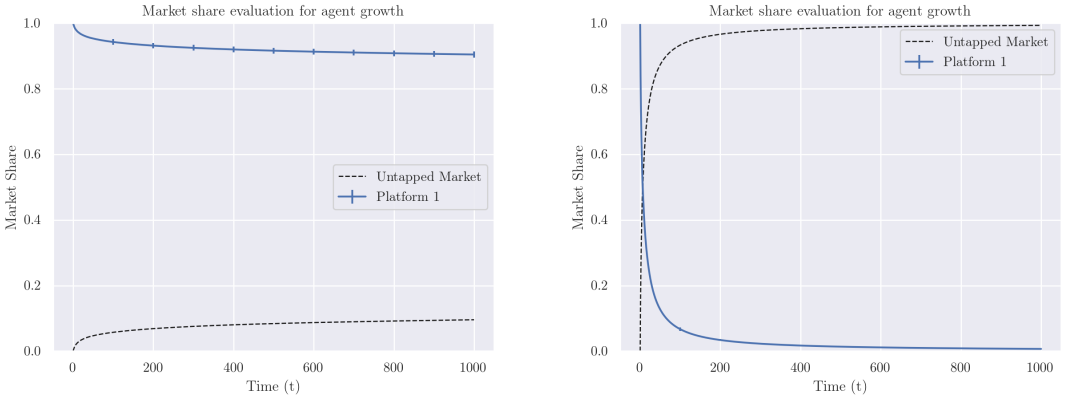
### Sensitivity to waiting/idle time

Figure 3.2 shows how varying $\mu_{waiting}$ and $\mu_{idle}$ impacts the total number of riders that join a single RHP. As expected, the sensibility to waiting time is inversely proportional to the final population, and this is also true for drivers and their sensibility to idle time.
Looking at this in more detail, figure 3.3 illustrates the average population growth of both the rider and driver population for $\mu_{idle} = \frac{1}{2}$ and $\mu_{waiting} \in \{0.01, 0.99\}$ over 100 runs. The impact of rider waiting-time sensibility is clear. With high sensibility, nearly all agents choose to not join the platform after only 200 time-steps. With a low value, the market-share converges to $\pm 90\%$.




(a) Final rider population w.r.t. $\mu_{waiting}$ (b) Final driver population w.r.t. $\mu_{idle}$

Figure 3.2: The impact of $\mu_{waiting}$ and $\mu_{idle}$ on the final population of the respective agents.




(a) $\mu_{waiting} = 0.01$ (b) $\mu_{waiting} = 0.99$

Figure 3.3: Agent population evolution with extreme values of $\mu_{waiting}$

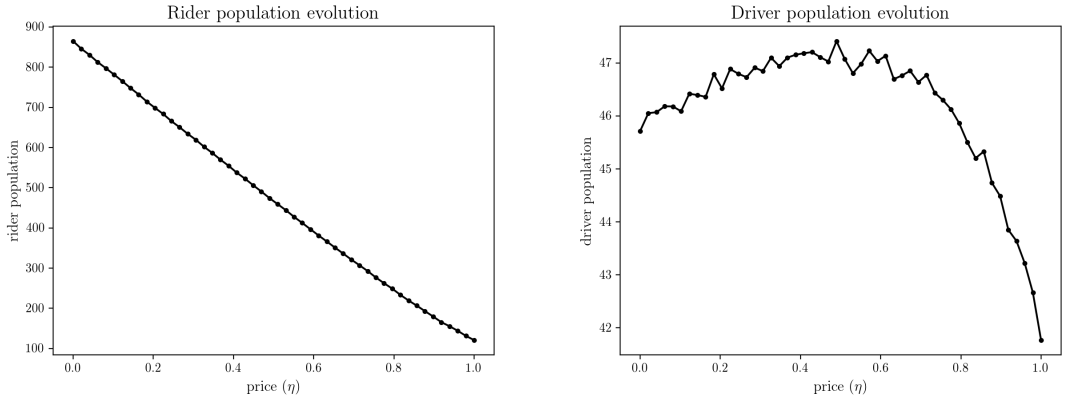
**Sensitivity to ride pricing**

Figure 3.4 illustrates the behaviour of the total rider and driver population as we change $\eta$. The linear decrease in rider population does validate our first intuition, but the way the driver population evolves is much more intriguing.
From the plot, we can see that the increase in price is initially favourable for the driver population, rising slowly, but once it hits $\pm 0.6$, the population rapidly declines. At first glance this seems counter intuitive, higher prices should be more attractive to drivers and shouldn't lower their population. What we are seeing here is actually that once the price goes past a certain limit, the rider population suffers so greatly, that the loss in customers out-weights the gain from each individual ride price. This indicates an "optimal" price which maximises the driver population, and of course makes perfect sense.

## 3.2 The links between ride-hailing and biology

Our model as described above, takes its roots from a population model of competition between cancer and T-cells [3]. Here we go into more detail about the reasoning behind this peculiar origin.

Riders and drivers in RHP benefit from a two sided feedback loop which takes into account competition for resources. This kind of phenomenon has been studied extensively, and specifically in the biological literature. One such model comes from immunology and studies the population growth rates coming from the interactions between cancer and T-cells. We leave to the reader any




(a) Final rider population w.r.t. $\eta$

(b) Final driver population w.r.t. $\eta$

Figure 3.4: The impact of $\eta$ on the final population of the respective agents.

philosophical considerations about comparing RHP users to cancer cells.

### 3.2.1 The original model

Here we will dive into the details of the original model, what they were modelling and what each term represents in immunological terms.

Consider a population of cancer cells structured by $u \in U \subset \mathbb{R}^+$ that represent their antigenic expressions, and a population of activated T cells structured by $v \in V \in U$ that represent those antigens that T cells can effectively attack.

The local densities of cancer cells and T cells are modelled by the functions $f_C(t,u) \geq 0$ and $f_I(t,v) \geq 0$. The related total densities are:

$$\rho_C(t) = \int_U f_C(t,u)du \qquad \rho_I(t) = \int_V f_I(t,v)dv$$

The model aims to model the growth of both cells using therapeutic agents boosting proliferation and immune memory: $c_P(t) \geq 0$ and $c_M(t) \geq 0$ respectively.

Cancer cells proliferate, net of apoptosis with rate $\kappa_C > 0$. Furthermore since cellular proliferation is hampered by the competition for resources, they assume that interactions lead to cancer death with rate $\mu_C > 0$.

Next T-cells undergo rapid clonal expansion which is accounted for by including binary interactions between cancer cells of trait $u$ and activated T cells with trait $v$: $\eta_{\theta_E}(|u-v|) > 0, \eta'_{\theta_E}(.) \leq 0$. Again due to limited resources, T cells die on average according to a rate $\mu_I > 0$.

To model the effects of T cells killing cancer cells, a second binary interaction is described by $\eta_{\theta_I}(|u-v|) > 0, \eta'_{\theta_I}(.) \leq 0$.

Again the proliferation rate of immune T cells is modelled by $\kappa_P > 0$, and the actions of therapeutic agents that boost immune memory is modelled through a reduction in death rate related to homeostatic regulation by parameter $\mu_M > 0$.

The dynamics of the two cell populations are described by the follow set of equations, where $R_C$ and $R_I$ model the net proliferation rates of cancer cells and T cells respectively:

$$R_C(t,u) := \qquad (\kappa_C - \mu_C \rho_C(t)) \qquad - \int_V \eta_{\theta_I}(|u-v|) f_I(t,v)dv \tag{3.4}$$

$$R_I(t,v) := \left[\int_U \eta_{\theta_E}(|u-v|) f_C(t,u)du + \kappa_P c_P(t)\right] - \frac{\mu_I}{1+\mu_M c_M(t)}\rho_I(t) \tag{3.5}$$

### 3.2.2 This parallelism

There are many direct parallelisms between the way riders and drivers behave within the world of RHPs, and how immune cells react to the presence of cancer.

The most important being competition for resources. Just as cells compete with themselves for the body's resources, riders compete for a valuable resource: the driver which will take them from their location, to their destination in the least amount of time. On the flip side, drivers also compete with each-other to serve the most customers, thus increasing their revenues. Both of these resources, interestingly being the agents themselves, are indeed finite and variable. Fundamentally, both cells and riders/drivers compete amongst themselves for a limited shared resource, leading to similar dynamics.

Another point of similarity is over-crowding, and it is a direct consequence from the competition for resources. As the density of a single cell-type increases, this also hinders their reproduction and is therefore self-limiting (in a situation with equal resources). This is analogous to a RHP that sees its number of riders explode, while its drivers do not. Eventually, each individual rider will have such a poor experience, that it will lead to slower growth [and identically for a growing number of drivers, with a constant number of riders].

While these comparisons may seem to be direct, there does seem to be subtleties. In particular, the death-rate of cells and the way that this factor is increased through over-crowding is known/can be measured empirically in a lab. Importantly, all cells of the same type can be assumed to behave identical. For ride-hailing platforms, this is not true. Each individual agent is unique, and while effects can be measured on the population, it is hard to say anything specific of individuals. An assumption that is necessarily made here is the homogeneity of these rider/driver agents, which must be kept in mind.

### 3.2.3 Modification to the original model

As our model is treating an entirely different domain, some modifications of the original model are necessary.

This first modification to be made was altering the way each agent is understood to grow. The basic cell proliferation rates can be considered constant, and this is simply not the case for a RHP. Its attractiveness is variable, and thus we have increased complexity by changing the constants $\kappa$ with functions $\kappa(t,u)$. These take into consideration the time ($t$) and the specific platform ($u$) in order to give a positive growth rate element, dependent on the RHPs' market-share.

The next change was regarding the way cell-death was amplified due to over-crowding, $\rho_A(t)$. This was originally defined as $\rho_A(t) = \int_U f_A(t,u)du$ $A \in \{C, I\}$, the density of cell type $A$. In our model, we consider different RHPs where they considered different antigenic expressions of a cancer or T-cell. This meant that evaluating over-crowding wasn't as straight forward as such an integral. Our solution was to consider the per-platform rider-per-driver (resp. driver-per-rider) ratio as this indicator. A high value would translate to over-crowding of riders (resp. drivers), which would amplify the negative rate term.

### 3.2.4 The limits of this comparison

As hinted as previously, this comparison is not perfect. The main issue being that in reality T-cells kill cancer cells, removing them completely from the population. In the RHP case, this comparison is obviously non existent. Whilst some agents might negatively affect others, pushing them away from a certain platform at time $t$, that agent does theoretically have the capability of returning to this platform at a later time $t_1 > t$. Note that the current model does not account for this.

Another smaller point is that the model made extensive use of the population function $f(.)$ and its density $\rho(.)$ by considering the "area" of the cells' possible antigenic expressions, represented on the $[0, 1]$ segment. In our case, the analogous space would be the different platforms that are being considered. This is a problem because while the original paper considered 400 antigenic expressions, we would only ever consider (at most!) a handful of RHPs, never going past the dozens. This lead to open ended terms in their model, and we used our expertise to fill them with what seemed to be the most appropriate substitutes.

# Chapter 4

# Results

## 4.1 Reproducing real-world data

### 4.1.1 The New York City data

In order to empirically validate our model, we decided to reproduce the market-share values and growths of major real-world RHPs competing in the same environment. To do this, we use the New York City's ride-hailing and taxi rides dataset provided by the Taxi and Limousine Commission [22]. This dataset contains completed rides dated from January 2015 to December 2019 and was made available thanks to a freedom of information act request.

Our data set contains 782 million. From these we extract market-share values for Uber, Lyft as well as Via and Juno (that we classify as other) and use this as ground-truth values for the benchmarking of our simulations. The specifics of how we go from the raw data to the data used here can be found in section 6.2.

### 4.1.2 Simulation results

To generate a model that corresponds to the real-world data, we only consider the parameters $\vec{\mu}_{waiting}$ and $\vec{\mu}_{idle}$, setting the price parameter $\vec{\eta}$ to 0 as most ride hailing platforms tend to offer similar prices [23]. This is both to reduce the size of the parameter search space (which is already substantial) and for simplicity. The details on the simulation itself can be found in section 6.1.

In order to score each model, we used the root mean-squared error (RMSE) metric between each of the generated market-shares and its corresponding ground truth value. The details of what parameters were tested, and the methods used can be found in section 6.1.4. To test the generalisability of our model, we first score each candidate using only 70% of the data, and observe how it extends onto the rest.

Figure 4.1 illustrates the ground-truth data obtained from the TLC dataset as markers (triangles: Uber, cross: other). The generated market-shares using our simulation are the full-lines of corresponding colour (blue: Uber, orange: other). The dashed line, market as Untapped, corresponds to the fraction of generated agents that chose to not join uber at the time where it was the only available platform, between January and April 2015.

The "bump" that can be seen at the early stage of the simulation is a consequence of how we chose to model monopolies differently from multi-platform scenarios. In a monopoly, agents join the unique platform w.r.t. the rate equations. This means that if the growth rate of a platform is $< 1$, some agents are essentially choosing to join no platform at all. This changes suddenly as a new platform enters the market, and now all generated agents must join a platform. Nonetheless, given that the fit was performed on only 70% of the data, this captures the approximate intuition that was aimed for and is overall a good result.

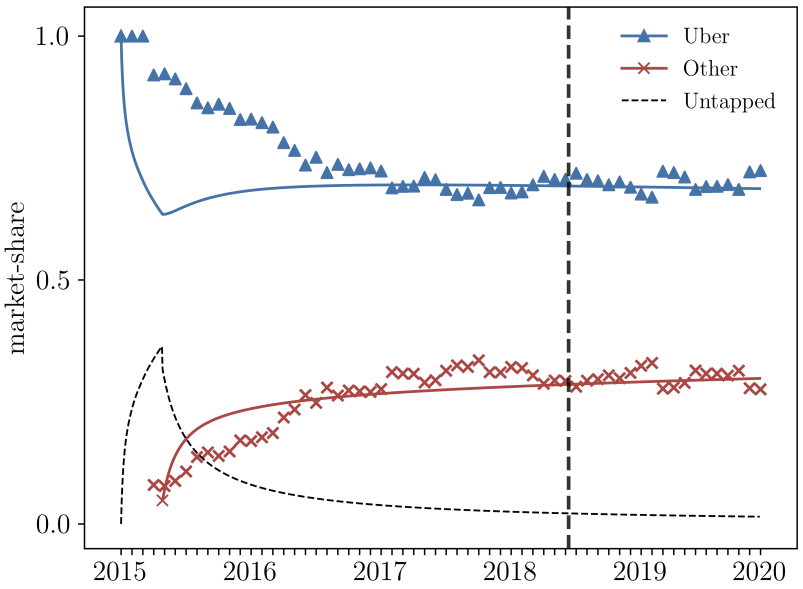



Figure 4.1: Optimal model fit on the first 70% of the NYC TLC dataset. The parameters are $\vec{\mu}_{waiting} = (0.25, 0.05)$ and $\vec{\mu}_{idle} = (0.9, 0.2)$

To show how how our model is able to fit long-term data, we also attempted to score each generation with the entire dataset. Upon doing so, the optimal parameters were independently found to be identical to those previously found.

#### RMSE Evaluation

In the above simulations, we have used the RMSE metric between each generated curve and the corresponding platforms' original data. The details of this can be seen in section 6.1.4. For these specific fits, the average RMSE value is 0.0818 on the "training" and 0.0701 on the "testing". Specifically, Uber had scores of 0.1072 and 0.0904, and Other had scores of 0.0565 and 0.0498.

## 4.2 Sensitivity Analysis

From the previous results, it would be natural to conclude that a strong first-mover advantage exists in our simulations, as it does somewhat in the real-world. In this section, we will investigate the robustness of this phenomenon, and discover under which conditions it breaks / arises.

Throughout we will compare the end of simulation market-share values of two RHPs, $u_1$ and $u_2$. In each simulation, we fix three of the four parameters that make up the platforms ($\vec{\mu}_{waiting}$ and $\vec{\mu}_{idle}$) and analyse the effect of varying the fourth as well as the delay after which the $2^{nd}$ platform is introduced to the market.

Note that throughout we will use the $\vec{\mu}$ vectors to represent the sensitivity to waiting and idle times of each platform, where the $i^{th}$ entry corresponds to $u_i$.

### 4.2.1 Sensitivity to waiting and idle times

Let us begin by observing the impact that agents' sensitivities to both waiting and idle time have on the competitiveness of a platform.

In these simulations we will fix a set of parameters:

$$\vec{\mu}_{waiting} = \begin{pmatrix} \mu^1_{waiting} \\ \mu^2_{waiting} \end{pmatrix} \quad \vec{\mu}_{idle} = \begin{pmatrix} \mu^1_{idle} \\ \mu^2_{idle} \end{pmatrix} \quad \vec{\eta} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad \text{with } \mu^i_{waiting/idle} \in [0, 1]$$

From these, we will choose one to be variable, and will also vary the delay after which platform 2 ($u_2$) enters the market (w.r.t. platform 1, $u_1$). This will give us a heat-map of values where the heat represents the market-share advantage (or disadvantage!) of $u_1$. As such, our first intuition might hint towards the fact that the longer this delay, the lower chances for $u_2$ to gain any users at all.

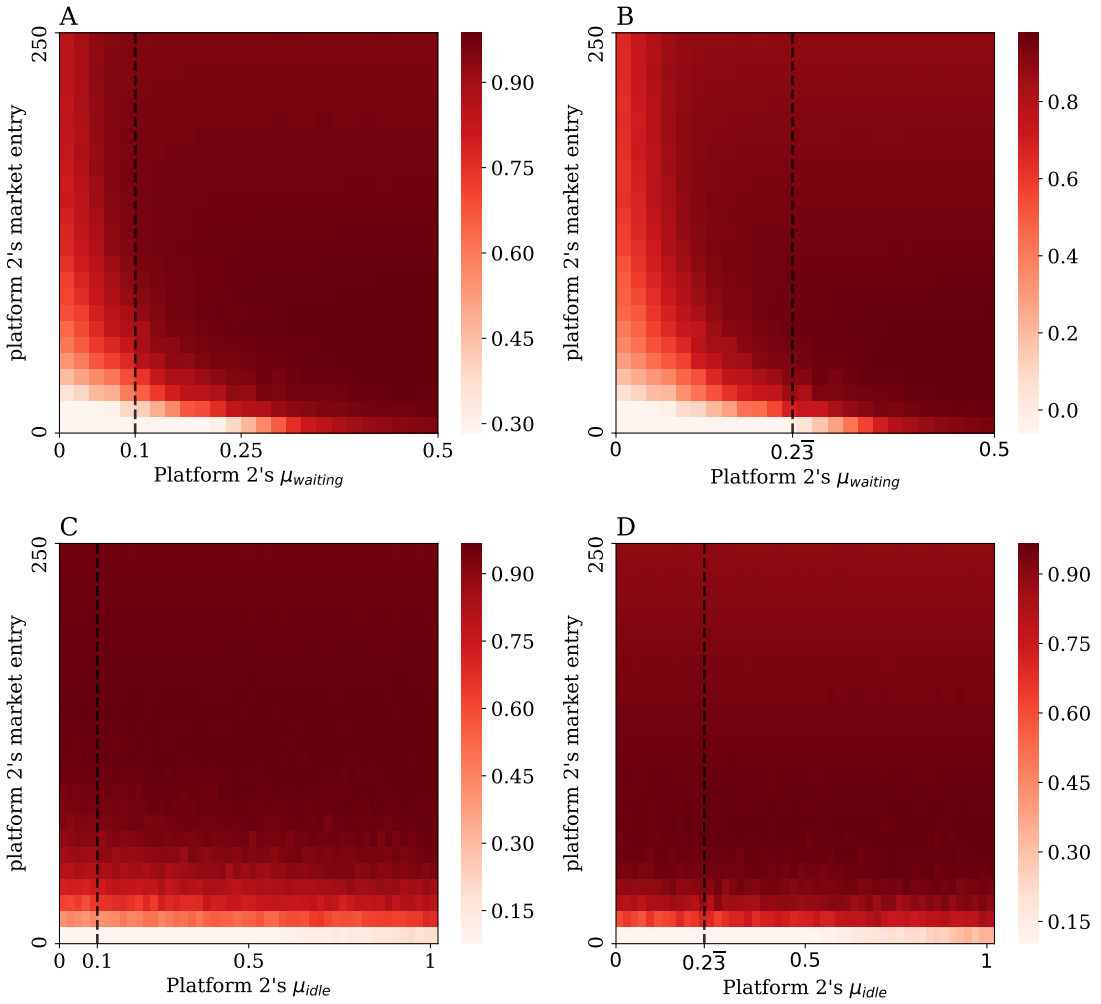



Figure 4.2: Difference in market-share $\kappa(u_1) - \kappa(u_2)$ where three of the four $\mu$ parameters are fixed. The dotted line represents the fixed parameters' value. **A**: $\vec{\mu}_{waiting} = (0.1, X)$ and $\vec{\mu}_{idle} = (0.1, 0.1)$. **B**: $\vec{\mu}_{waiting} = (0.2333, X)$ and $\vec{\mu}_{idle} = (0.2333, 0.2333)$. **C**: $\vec{\mu}_{waiting} = (0.1, 0.1)$ and $\vec{\mu}_{idle} = (0.1, X)$. **D**: $\vec{\mu}_{waiting} = (0.2333, 0.2333)$ and $\vec{\mu}_{idle} = (0.2333, X)$.

Figure 4.2 shows in red how much of an advantage platform 1 has compared to platform 2 in terms of market-share. The black-dashed lines here represent the values that the three other (non-varying) parameters are fixed at. In Figures 4.2 **A** and **B** a curve can clearly be seen that marks a boundary on which platform 2 must aim to be within. Joining the market late, if it does not have an exceptionally good waiting times, means agents will certainly favour platform 1. Its only chance is to minimise both delay to market entry, and this $\mu_{waiting}$. This is also the case, although with less importance, for the idle time as shown in figures 4.2 **C** and **D**. Here the parameter doesn't have much of an effect on the overall outcome (due to the low population of drivers). The key factor is to arrive into market as soon as possible in order to mitigate the first-mover-advantage penalty, as was somewhat expected.

### 4.2.2 Sensitivity analysis on real-world fitted model

Now that we have analysed a somewhat arbitrary case, how does this extend to the real world? In this section we will take the parameters that mirror the real-world situation in New York City (as per section 4.1.2). That is, we will set $\vec{\mu}_{waiting} = (0.25, 0.025)$ and $\vec{\mu}_{idle} = (0.9, 0.2)$. From the data, we also know that Uber's competitor entered the market at the equivalent of the $65^{th}$ time-step in our simulation. To investigate on the sensitivity of each of these variables, we will generate four more heat-maps as previously. The $X$ will represent one of the four above parameters, and the $Y$ will always be the competitor's market-entry time.

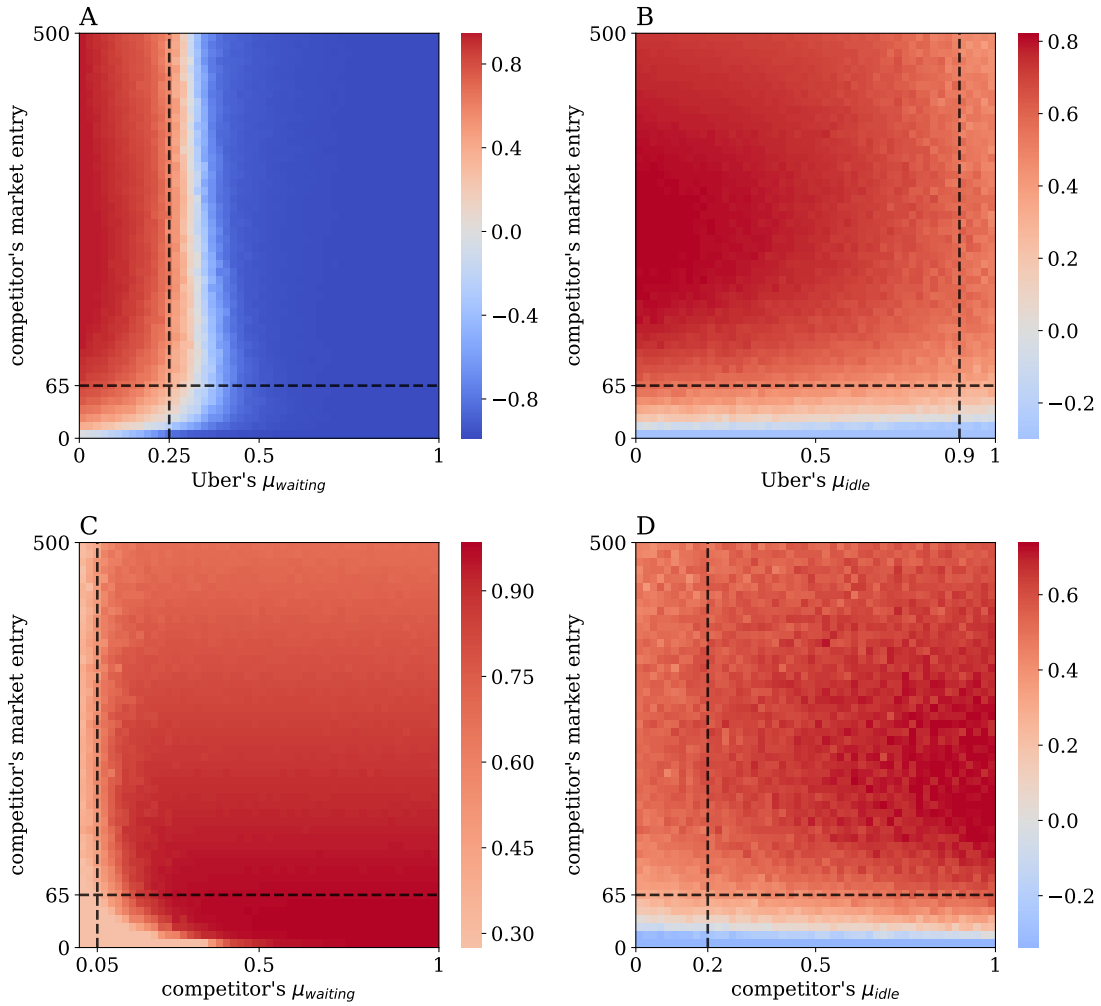



Figure 4.3: Difference in market-share $\kappa$(Uber)$-\kappa$(competitor) where three of the four $\mu$ parameters are fixed. The dotted line represents the fixed parameters' value. **A**: $\vec{\mu}_{waiting} = (X, 0.05)$ and $\vec{\mu}_{idle} = (0.9, 0.2)$. **B**: $\vec{\mu}_{waiting} = (0.25, 0.05)$ and $\vec{\mu}_{idle} = (X, 0.2)$. **C**: $\vec{\mu}_{waiting} = (0.25, X)$ and $\vec{\mu}_{idle} = (0.9, 0.2)$. **D**: $\vec{\mu}_{waiting} = (0.25, 0.05)$ and $\vec{\mu}_{idle} = (0.9, X)$.

Figure 4.3 shows this analysis. The dotted lines in each plot correspond to the real-world values of the corresponding variables. As an example, in figure 4.3 **A**, $Y = 65$ corresponds to the real time at which Uber's competitors come into market, and $X = 0.25$ corresponds to Uber's real $\mu_{waiting}$ (as found by our model). The heat in red shows areas where Uber comes out ahead in terms of market-share, and those in blue correspond to situation where its competitors come out ahead.

As we have seen previously, each of these figures contain a boundary around which healthy competition occurs. Furthermore, each intersection of the dashed lines (which correspond to the exact situation that reproduces the NYC market) is located almost exactly on this boundary! Again this is more visible in figures 4.3 **A** and **C** due to the abundance of rider agents v.s. driver agents.

It also tells us that if Uber's $\mu_{waiting}$ was slightly higher, its competitors would rapidly dominate the market. Similarly we can see that if the competitors had joined the market earlier, they would have an increasing domination of the NYC market-share.

### 4.2.3 Model limitations

In this section we will observe the behaviour that arises from the rate equations when we fix relatively high values for the $\mu_{waiting}$ and $\mu_{idle}$ times. This is not expected to reproduce intuitive, real-world applicable data, but is nonetheless interesting.

As previously, in each simulation we fix three of the four parameters appearing in $\vec{\mu}_{waiting}$ and $\vec{\mu}_{idle}$.

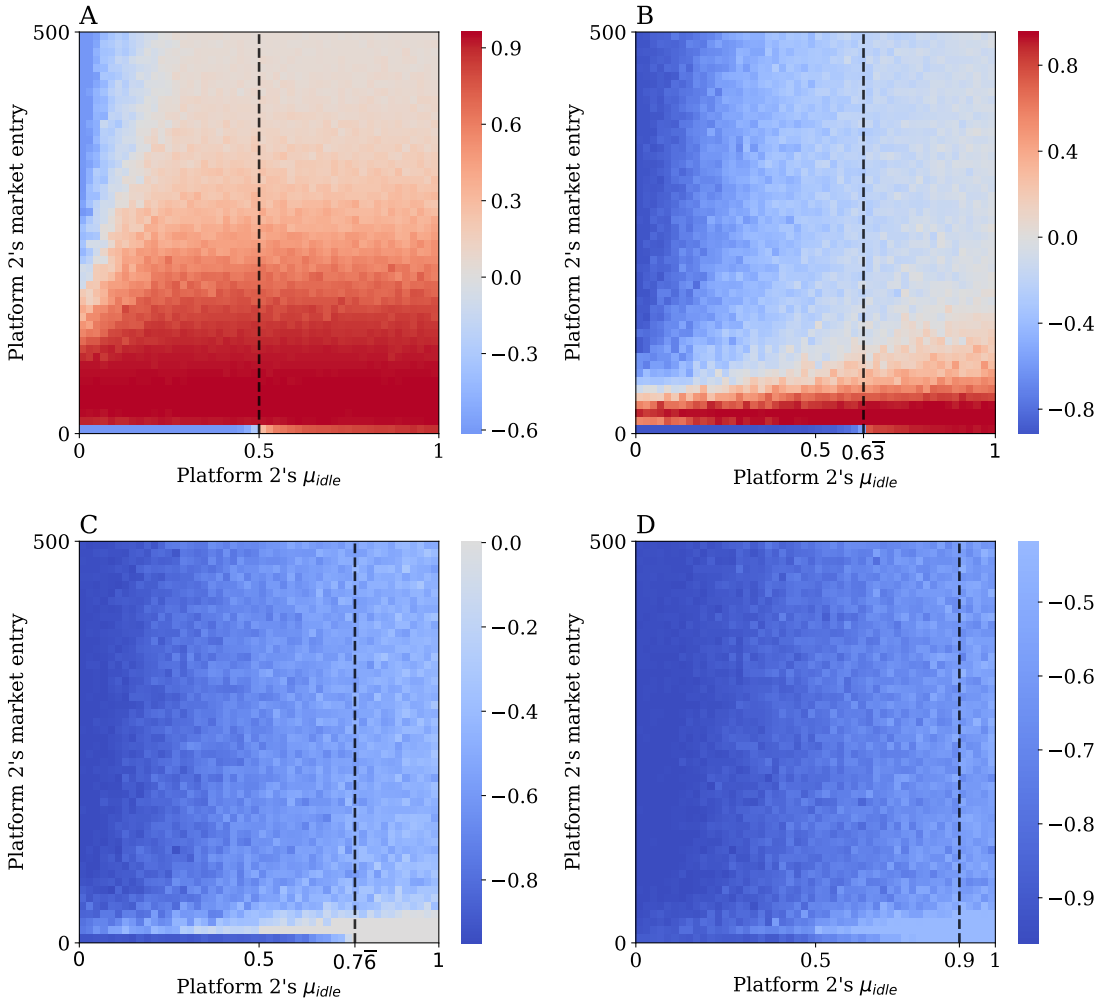



Figure 4.4: Difference in market-share $\kappa(u_1) - \kappa(u_2)$ where three of the four $\mu$ parameters are fixed. The dotted line represents the fixed parameters' value. **A**: $\mu_{waiting} = (0.5, 0.5)$ and $\mu_{idle} = (0.5, X)$. **B**: $\mu_{waiting} = (0.6333, 0.633)$ and $\mu_{idle} = (0.6333, X)$. **C**: $\mu_{waiting} = (0.7666, 0.7666)$ and $\mu_{idle} = (0.7666, X)$. **D**: $\mu_{waiting} = (0.9, 0.9)$ and $\mu_{idle} = (0.9, X)$.

In figure 4.4 we can start seeing a trend. As we increase the base value, the boundary at which these platforms can coexist in a duopoly moves somewhat from left to right. In Figure 4.4 **A**, when all parameters are about average we see something interesting happening. We begin to clearly see somewhat of a border between the red and new blue regions. This area corresponds to the parameter space in which these platforms are able to coexist by splitting the market 50-50 in a relatively stable manner. We are also able to clearly see situations where platform two not only is able to achieve competition, but also dominate the market. The reason the boundary is so volatile in

figures 4.4 **B**, **C** and **D** is that $\mu_{idle}$ has a notoriously low impact on the final market-shares of the platforms, due to their relatively low population size compared to riders. The boundary movement here is due to the base values of the $\mu_{waiting}$ more than the variation of the $\mu_{idle}$

Going into detail, it seems always better for platform 2 to enter the market as late as possible. What is going on here? It turns out that if platform 1 isn't attractive enough in terms of waiting/idle time, in a monopoly, agents would rather not join a platform at all. This leaves $\kappa(u_1)$ to be strictly decreasing throughout the simulation, and the longer this goes on, the easier it would be for platform 2 to come out ahead thanks to its more favourable parameters. Furthermore, even when platform 2's parameters are not much better than $u_1$'s, the market converges to a duopoly where both platforms share about 50% of the market each.

Overall these figures seem to suggest that the FMA is quite weak in these scenarios. Even though as seen previously $\mu_{idle}$ has a relatively low impact on the overall platforms' market-share, we can see that the FMA can still be reversed with low values of $\mu_{idle}$ for an appropriately "bad" base values.

Next we will perform the same analysis using $\mu_{waiting}$, and will identify similar results, in a much stronger fashion.

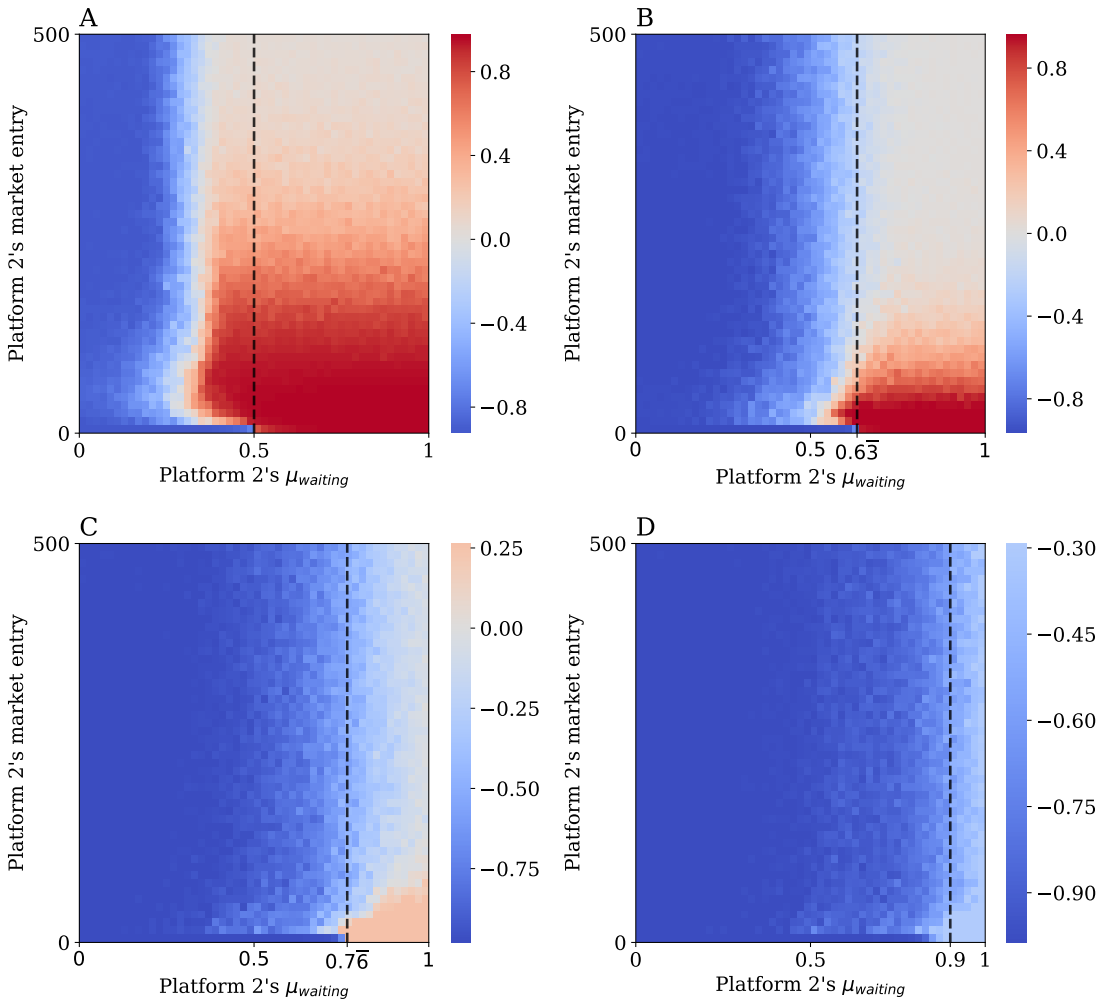



Figure 4.5: Difference in market-share $\kappa(u_1) - \kappa(u_2)$ where three of the four $\mu$ parameters are fixed. The dotted line represents the fixed parameters' value. **A**: $\mu_{waiting} = (0.5, X)$ and $\mu_{idle} = (0.5, 0.5)$. **B**: $\mu_{waiting} = (0.6333, X)$ and $\mu_{idle} = (0.6333, 0.6333)$. **C**: $\mu_{waiting} = (0.7666, X)$ and $\mu_{idle} = (0.7666, 0.7666)$. **D**: $\mu_{waiting} = (0.9, X)$ and $\mu_{idle} = (0.9, 0.9)$.

Figure 4.5 **A** is extremely intriguing. We can again clearly identify a curve flowing between the regions where platforms 1 and 2 respectively dominate. This boundary corresponds to an equilibrium where in those conditions both RHPs are able to exist in a duopoly, splitting the ride-hailing market 50-50.

In addition to this, almost irrespective of the delay to market entry, for values of $X < 0.3$, $u_2$ systematically comes out ahead in terms of market share. For the other values, we see a completely different version of the story: $u_1$ comes out well ahead, but increasing the delay to market entry is actually detrimental to $u_1$! Again in all scenarios, it is in platform 2's best interest to join the market as late as possible.

Now from Figure 4.5 we can see how this boundary which we discussed earlier washes through the plot and disappears almost completely in Figure 4.5 **C**. Once the default sensibilities to waiting and idle times are high for platform 1, it continuously bleeds users by offering a poor service. This effect is so strong that regardless of when platform 2 comes in, it wins by a landslide. Note also that although $u_2$ wins systematically, as soon as its own sensitivity to waiting time ($X$) goes above 0.7 (and even more-so after that), the win isn't so monopolistic, with the far right sides of the plots being lighter coloured, indicating that platform 1 is able to retain some of its market-share, coexisting with platform 2.

Overall, these plots seem to suggest that $\mu_{waiting}$ in particular is extremely sensitive and powerful in this model. Keeping it to a low value seems to best reproduce real-world scenarios, where exponential growth still exists although in a controlled fashion. Increasing its value leads to situation where agents would rather not join a platform, even if it is the only one offering its service. Naturally this allows any new platform that enters the market with slightly better service to quickly and immediately dominate the market.

# Chapter 5

# Discussion

**Achievements**

From our model we find that $\mu_{waiting}$ especially has a formidable impact on the growth of each platform (more than $\mu_{idle}$ due to the population differences). Indeed, by varying $\mu_{waiting} \in [0, 1]$ we are able to reproduce one of multiple scenarios which all exist in the real world. In the case where $\mu_{waiting} = 0$ we saw that this allows for the reproduction of exponential growth due to the usual network effects. Then, as we increase this value slightly, we start to be able to control this growth, lowering it bellow the heavy monopoly. This area is extremely sensitive and increasing $\mu_{waiting}$ too far tips it into a third phase, where the exponential growth is completely cancelled out. From this point forward, larger values lead to a larger and larger bleed in users from the platform, even in a monopoly, and even goes to hindering the platform from future growth as a new platform enters the market.

We have also shown that our model is able to accurate reproduce the market-share evolution of the two biggest ride-hailing platforms in arguably the most important competitive environment of the United States (New York City). We have also shown that given partial data, our model is able to generalise to a lesser extend, but nevertheless brings useful information and is able to capture the general trends of the platforms' growths.

Through a sensitivity analysis we have also shown that the first mover advantage, whilst strong in some settings, is not indefinite and can be easily counteracted by the agents' sensibilities to waiting/idle times.

**Limits of the first mover advantage**

After our initial simulations, it seemed as though being late to join the RHP market would condemn the platform to a low market-share. With this in depth analysis for the 2-platform scenario, this seems increasingly unlikely.

It has been clear throughout this work that the sensibility of riders to the waiting times of their platforms was one of the main driving factors to achieve high attractiveness. We have now empirically shown that it is also stronger than the FMA advantage. Platforms that join the market late, even half-way into the simulation, are still able to completely take over the market. How is this possible?

One interesting feature that we observe in these specific simulations is that as soon as the $2^{nd}$ platform enters the market, the way the simulation works is slightly changed. Up-to that point, agents join the single RHP following their "raw" rate equations. This meant that if the platform had relatively unfavourable parameters, it would quite rapidly loose in effective market-share. This loss in market share is then capitalised on by the second platform as soon as it enters the market. At that point, agents must join one of the 2 platforms as their rates are normalised. This is where the previously low market-share for platform 1 brings it to an equal with the entering platform, and if its parameters are more attractive, it will swiftly take over the market. In this sense, the FMA is essentially void if the platform's parameters are unfavourable.

This also means that if the first platform's agents are not sensitive to waiting/idle time, the FMA is indeed strong, and allows it to dominate the market with little resistance from the entering platform.

## 5.1 Limitations of current study

Whilst this work finds strong results, it does has its limitations.

We recognise here that one major flaw of the simulation is how it behaves differently in a monopoly from an oligarchy. Given that agents have a positive chance to join no platform, with the correct parameters, the way the market-share represents the platforms growth has its limits. Platforms in a monopoly can essentially bleed users due to their bad service, reducing their effective market-shares greatly as time goes on. Although this make sense in some situations as we discussed previously, the range of values for the $\mu$ in which the exponential growth is controlled is small. A mitigation of this could be that this range, although small with respect to its domain, can be extended infinitely (up-to floating point errors).

Another limitation from this study is the fact that unlike many previous models on ride-hailing platforms, we do not model the number of rides directly. By modelling the growth of each agent type, we forgot the details of ride-matching, a key component of RHPs. This effectively limited our options as far as empirical validation goes as more data is available on number of rides, rather than number of riders/drivers (which are often bundled as "users"). Again a mitigation of this could be an argument along the lines that from the total number of riders and drivers, one could extrapolate from known data how many rides may happen per unit of time. We do not go into these details here.

Lastly, we acknowledge that the simplicity of some parts of the model might come as a shortcoming. Firstly, both $\mu_{waiting}$ and $\mu_{idle}$ could be modelled differently, in a way that makes them both related (as they could be in the real-world). Secondly, the price coefficient $\eta$ is quite simplistic and hasn't been expanded much upon in this research. In the real world, this price is not-only variable, but is modelled very specifically and varies quite a lot from platform to platform. These three improvements would drastically increase complexity but would lead to an even more realistic simulation. Here we have omitted these largely to favour the simplicity of this model.

## 5.2 Future research directions

As hinted towards in the limitations of this study, there are multiple avenues one could go down in terms of extending this work.

Firstly, increasing the complexity of $\mu_{waiting}$, $\mu_{idle}$ and $\eta$ would greatly improve the performance of this model. What's more, perhaps unifying the definitions for the $\mu$ might drastically reduce the parameter space which currently grows exponentially with the number of additional platforms.

Another way this model can be improved is by considering the geography of a particular city in order to better refine the over-population metrics that are already used. Having this geography introduced to each agent would allow for a global simulation. Simulating the entire Uber company, and observing the growth of different clusters in response to other platforms would be an extremely interesting extension of this work.

### 5.2.1 Wider world implications

As much as this work brings an interesting new light to the limits of network effects, it can also be used by policy makers and competition authorities. We will illustrate this here with a couple of imaginary scenarios where our work could come into play.

In the metropolis of Uberopolis, competition authorities are having their quarterly market review to assert that no company is taking advantage of anti-competitive practices. In doing so, they pay particular attention to the ride-hailing market, as this one is known to greatly benefit from

network effects. Although it is expected for their growth to be exceptionally high, something seems off. A particular nameless company seems to be completely taking over the market, quicker than has historically been recorded. Using a newly written model, they quickly ask for a specific set of information from all existing platforms. One such parameter is the average time after which their riders decide to cancel the rides, the $\mu_{waiting}$ in our model (similarly for $\mu_{idle}$ and $\eta$). Using all of this information, they are able to simulate the current market and track market-shares as well as rider and driver population growths. From this, and using other investigative methods, they are able to make a more informed judgement regarding this platform's fate.

# Chapter 6

# Methods

## 6.1 The Simulation

Here we will go into the details of how the simulation works exactly, and how we are able to extract results from it.

The code is entirely written using Python 3.7 and the only dependencies are Numpy [24] for data-structure manipulation and Matplotlib [25] for the generation of plots.

### 6.1.1 Agents

At the heart of our simulations are our agents. These are intentionally kept as very simple classes that are created only when a platform is set to grow. On their creation (as part of a variable cluster size), they are assigned to be either riders (with 95% probability) or drivers (with 5% probability). This large difference in population is parametric, but is fixed to these values in order to match virtually all empirical data on ride-hailing platforms. Once they are created, the simulation supplies them with a set of information regarding each platform that is currently on the ride-hailing market. This includes the number of riders and drivers within each platform at the given time, as well as the proportion of riders/drivers that those correspond to within the market. Using this information, the agents apply the respective rate equations (equation (3.1) or (3.3)) in order to generate a joining rate corresponding to each platform. If there is a single platform, then this rate corresponds to the direct growth for the platform, otherwise these rates are first normalised (making them sum to 1) before being sent as growths.

### 6.1.2 Ride-Hailing Platforms

Around each of these agents and at a higher level we can find ride-hailing platforms (RHPs). These hold most of the actual data for the simulation. Each platform keeps track of its own number of agents (both drivers and riders separately) as well as the corresponding market-share. These are all stored indefinitely, meaning that platforms store their entire growth history and makes it easy to extract at the end of the simulation. Upon their creation, each platform also immediately starts with both 1 rider and 1 driver. Platforms are also given information about the current market. As part of their constructors, they receive the current total number of riders and drivers throughout all other platforms. This means that each platform always starts with a market-share of $\frac{2}{N}$, where $N$ is the total number of agents currently in the simulation. As just hinted at, the market-share of a platform is defined to be the number of users it has, over the total number of users in the market. Although this isn't typically how market-share is calculated, this metric should be equivalent to the real-world market-share for our purposes (and would only be off by a constant factor regardless).

### 6.1.3 The Simulator

Finally comes the simulator class. This object has the responsibility of creating and running a single iteration of a simulation, from end to end. It is created directly from the main program loop and is given all the appropriate parameters. These include:

- $N$: the simulation length (in number of new agent generation waves),
- $U$: the number of RHPs to generate in the simulation,
- $\mu_{waiting}$: as a list it corresponds to the sensibility to waiting time for each of the platforms in the simulation,
- $\mu_{idle}$: similarly, the sensibility to idle time of each platform,
- $\eta$: this price parameter for each platform, this is an indication of how expensive each ride will be,
- *delays*: as a list, the time at which (during the simulation) each individual platform is to enter the market. At least one platform must be active at time 0,
- $n\_joins$: the number of agents to release onto the market at each time-step.

From these parameters, the simulator's first task is to generate all the platforms that need to start at time 0, as well as to schedule all the next platform creations / entries to market. Then a simple global clock counts through the $N$ steps, generating $n\_joins$ new agents and making them decide which platform to attribute growth to.

This entire process is then repeated *it* number of times in the main program loop. This allows us to extract a 95% confidence interval for all the interesting metrics, namely: number of riders, number of drivers and market-shares throughout the simulation.

### 6.1.4 Constraint optimisation based grid-search

In order to find the optimal set of parameters which describe real data as accurately as possible, we perform a grid-search on the parameter space.

In a scenario with $P$ platforms, each simply having a pair of $\mu_{waiting}$ and $\mu_{idle}$ parameters, leads to $N^P \times M^P$ possible combinations (where $N$ and $M$ are the granularities of the $\mu$). Running each combination at least 10 times in order to get an average RMSE score is obviously not tractable if we wish to consider a granularity of less than $10^{-1}$. In order to circle this problem, let us consider four possible scenarios. An individual platform can have a combination of:

- Many drivers, for few riders: in which case $\mu_{waiting}$ would be low and $\mu_{idle}$ should be high,
- Few drivers, for many riders: in which case $\mu_{waiting}$ would be high and $\mu_{idle}$ should be low,
- Many drivers, for many riders: in which case both $\mu_{waiting}$ and $\mu_{idle}$ would be low,
- Few drivers, for few riders: again in which case both $\mu_{waiting}$ and $\mu_{idle}$ would be low.

As we can see, it is never possible for a real RHP to have both a high $\mu_{waiting}$ and a high $\mu_{idle}$. This means we can apply constraint optimisation to our parameter space. Hence whilst generating it, we can simply apply this rule (limiting the average value of the $\mu$) to greatly reduce the number of parameters. In order to generate our main results, we used this technique with $N = 19, M = 9, P = 3$ and $L = 0.6$. Limiting $\frac{\mu_{waiting}+\mu_{idle}}{2} \leq L$ lead to a reduction of the search-space by 70%.

## 6.2 New York City's Taxi and Limousine Commission (TLC) dataset

The New York City Taxi and Limousine Commission (TLC) is a local government agency that licenses and regulates the medallion taxis and for-hire vehicle industries. They have been publishing (and somewhat updating) a tremendous amount of data regarding all of their recorded rides within the city: a total of 2.63 billion trips are recorded. Importantly for our purposes, this data is not limited to just cab rides. It is initially split into three parts: data bout yellow cabs, green cabs, and for-hire-vehicles. The latter is of interest to us as this represents rides offered by our

ride-hailing platforms.

This data on RHPs is available monthly from January 2015, and monthly thereafter up-to December 2019. These records are extremely detailed:

- `dispatching_base_number`: A unique identifier attributed to each RHP. These identifiers are also tied to a geographic area,
- `PUdatetime` and `DOdatetime`: The pick-up and drop-off time and dates (accurate to the second),
- `PUlocationID` and `DOlocationID`: The pick-up and drop-off location IDs which correspond to various metropolitan areas. These are broken up into "Borough" and "Zone" (e.g. Manhattan / Lincoln Square East),
- `SR_flag`: Binary value (0 or 1), determines whether or not the ride was shared (e.g. Uber Pool).

*Note that for years 2015 and 2016, the only information is the* ***`dispatching_base_number`*** *as well as* ***`datetime`*** *and* ***`locationID`*** *of either pick-up or drop-off (not specified).*

Readers who are interested in seeing ways in which this entire dataset can be analysed are invited to read Todd Schneider's article "Analyzing 1.1 Billion NYC Taxi and Uber Trips, with a Vengeance" [26].

For our purposes, we only need a single piece of information for each individual ride: which RHP offered it. This can directly be cross-referenced with a mapping from base number to RHP, which is provided by the TLC. Unfortunately, there are approximately 126 million rows that either contain no base number, or one that isn't documented. These rides will thus be ignored. This leaves us with a total of 782 million rides that have been offered by one of: Uber, Lyft, Juno or Via between 2015 and 2019.

We will use this data in two ways, one of which was particularly experimental. First, we grouped the total number of rides that each platform offered by date, and extracted monthly market-share data on these platforms. In order to simplify the analysis (and to focus on the two main platforms), we have aggregated all data corresponding to Via and Juno and labelled it as "other" for all of our simulations. Secondly we also regroup and count the number of rides offered by each platform in order to generate an accumulated number of rides offered by them.

Lastly, in order to use RMSE as a metric to compare the market-share points generated by our simulations and the this real-world data, it was necessary to either down-sample the $N$-dim market-share vectors generated (1 entry per time-step), or to up-sample the real data. For no particular reason, we chose to up-sample the real market share values, by extrapolating the values between each month. Starting with 60 data points, this increase the resolution to 1889 points. Therefore we simply increased the length of the simulations to match the length of the data.

# Appendix A

# Additional Plots

## A.1 Model fully fitted on the NYC TLC dataset

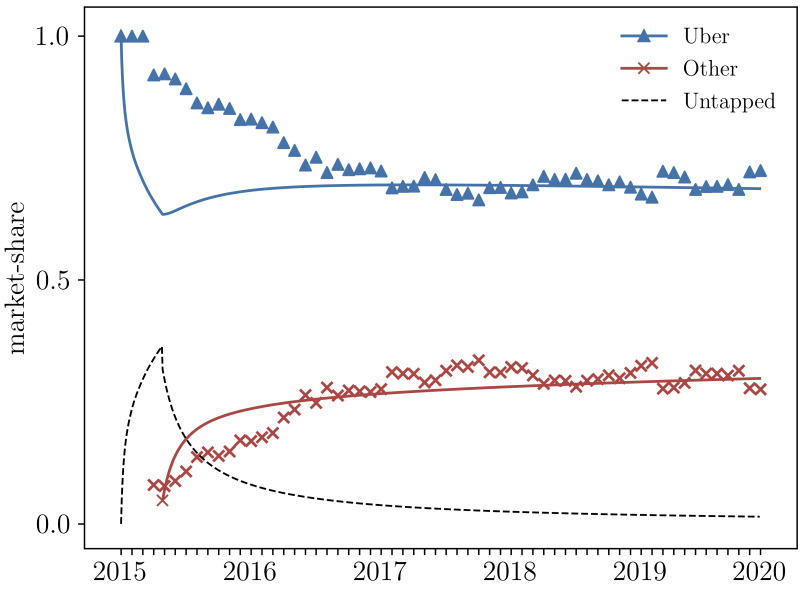



Figure A.1: Optimal model fit on the entire NYC TLC dataset. The parameters are $\vec{\mu}_{waiting} = (0.25, 0.05)$ and $\vec{\mu}_{idle} = (0.9, 0.2)$

# Bibliography


[1] Mark EJ Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[2] Remi Tachet, Oleguer Sagarra, Paolo Santi, Giovanni Resta, Michael Szell, SH Strogatz, and Carlo Ratti. Scaling law of urban ride sharing. *Scientific reports*, 7:42868, 2017.

[3] Marcello Delitala, Tommaso Lorenzi, and Matteo Melensi. A structured population model of competition between cancer cells and t cells under immunotherapy. In *Mathematical Models of Tumor-Immune System Dynamics*, pages 47–58. Springer, 2014.

[4] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[5] Carl Shapiro, Shapiro Carl, Hal R Varian, et al. *Information rules: a strategic guide to the network economy*. Harvard Business Press, 1998.

[6] Sidney Redner. How popular is your paper? an empirical study of the citation distribution. *The European Physical Journal B-Condensed Matter and Complex Systems*, 4(2):131–134, 1998.

[7] Robert K Merton. The matthew effect in science: The reward and communication systems of science are considered. *Science*, 159(3810):56–63, 1968.

[8] Derek J De Solla Price. Networks of scientific papers. *Science*, pages 510–515, 1965.

[9] Matthew T Clements. Direct and indirect network effects: are they equivalent? *International Journal of Industrial Organization*, 22(5):633–645, 2004.

[10] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. 1998.

[11] Arun Sundararajan. Local network effects and complex network structure. *The BE Journal of Theoretical Economics*, 7(1), 2007.

[12] Daniel O'Connor. Understanding online platform competition: Common misunderstandings. *Internet Competition and Regulation of Online Platforms (May 2016), Competition Policy International*, 2016.

[13] Jeffrey Rohlfs. A theory of interdependent demand for a communications service. *The Bell journal of economics and management science*, pages 16–37, 1974.

[14] Jean-Charles Rochet and Jean Tirole. Platform competition in two-sided markets. *Journal of the european economic association*, 1(4):990–1029, 2003.

[15] Giacomo Luchetta. Is the google platform a two-sided market? *Journal of Competition Law and Economics*, 10(1):185–207, 2014.

[16] E Glen Weyl. A price theory of multi-sided platforms. *American Economic Review*, 100(4):1642–72, 2010.

[17] Andrei Hagiu and Simon Rothman. Network effects aren't enough. *Harvard business review*, 94(4):64–71, 2016.

[18] INQUISITR. Lindsay lohan banned by facebook. https://www.inquisitr.com/10537/lindsay-lohan-banned-by-facebook/, 2008. [Online; accessed 22-January-2020].

[19] Paul Erdős and Alfréd Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60, 1960.

[20] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440, 1998.

[21] John Guare. *Six degrees of separation: A play*. Vintage, 1990.

[22] NYC Taxi Limousine Commission. TLC trip record data. https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page, 2020. [Online; accessed 21-January-2020].

[23] Alyx Gorman. Didi, Uber, Ola and Bolt: compare which rideshare app offers passengers and drivers the best deal. https://www.theguardian.com/technology/2020/feb/21/easy-rider-compare-which-rideshare-app-offers-passengers-and-drivers-the-best-deal, 2020. [Online; accessed 11-June-2020].

[24] Open-Source. Numpy: The fundamental package for scientific computing with python. https://numpy.org/, 2020. [Online; accessed 09-June-2020].

[25] Open-Source. Matplotlib: a comprehensive library for creating static, animated, and interactive visualizations in python. https://matplotlib.org/, 2020. [Online; accessed 09-June-2020].

[26] Todd Schneider. Analyzing 1.1 Billion NYC Taxi and Uber Trips, with a Vengeance. https://toddwschneider.com/posts/analyzing-1-1-billion-nyc-taxi-and-uber-trips-with-a-vengeance/, 2015. [Online; accessed 09-June-2020].