Optimization of Orders Assignment to Couriers for On-demand Delivery

Tomás Maria Loureiro Sarreira Mateus Rocha

Thesis to obtain the Master of Science Degree in

Industrial Engineering and Management

Supervisors:

Prof. Diana Rita Ramos Jorge

Prof. Tânia Rodrigues Pereira Ramos

Examination Committee

Chairperson: Prof. Susana Isabel Carvalho Relvas

Supervisor: Prof. Diana Rita Ramos Jorge

Member of the Committee: Prof. Nuno Antunes Ribeiro

December 2022

Acknowledgments

This Master’s Thesis would not have been possible without the invaluable support and guidance from my supervisors, Professor Tânia Rodrigues Pereira Ramos and Professor Diana Rita Ramos Jorge. Thank you for sharing with me your knowledge, expertise and helpful advice, as well as for always being available throughout the whole development of this dissertation.

I would like to extend my gratitude to my family and to my girlfriend Ana who I could always count on throughout my entire academic journey. Thank you for your unconditional support.

I cannot end this chapter without acknowledging the support of friends I met along the way. Thank you for the moments we shared.

Abstract

The on-demand economy has shaped the habits of consumers, who now expect faster deliveries. Instant deliveries gained popularity for carrying meals to urban areas, and the spectrum of products available to order has broadened. The platforms that provide these services rely on crowdsourced couriers, who use their personal vehicles, resulting in a heterogeneous fleet. Companies experience intense competition to retain both customers and couriers. Hence, it is vital to develop superior optimization models integrating multiple types of vehicles, capable of producing assignments in real-time to meet customers and couriers' expectations.

An optimization model is developed to solve the assignment problem with vehicle restrictions using the Jonker-Vogenant and Branch-and-Cut algorithms. The mathematical model is inserted into a dynamic framework that continuously solves it, while controlling the arrivals of orders, couriers' shifts and performing position updates. The model also contemplates dynamic traffic congestion and regional speed limits for different types of vehicles. Besides a myopic assignment approach, the implementation of policies is investigated – extended assignment policy and bicycle policy – in order to improve performance along different metrics.

Based on real-world instances, the proposed model with the extended assignment policy achieves a decrease in total delivery time of approximately 4.5% and an increase of 9.6% in balanced courier utilization when compared with the real assignment, thus improving the solution from the customer and couriers' perspectives. The bicycle policy achieves an increase in balanced courier utilization of 3pp, at the expense of a 0.3% increase in delivery time compared to the baseline model.

Keywords: on-demand, instant delivery, courier assignment, dynamic assignment problem, heterogeneous fleets, mixed-integer programming.

Resumo

A economia *on-demand* tem moldado hábitos dos consumidores, que exigem entregas mais rápidas. As entregas instantâneas ganharam popularidade no transporte de refeições para áreas urbanas, e o espetro de produtos disponíveis para encomendar tem-se expandido. As plataformas que disponibilizam estes serviços dependem de estafetas *crowdsourced*, que usam veículos pessoais, originando frotas heterogéneas. Verifica-se uma intensa concorrência para reter consumidores e estafetas. Assim, é vital desenvolver modelos de otimização que integrem múltiplos tipos de veículos, capazes de produzir atribuições em tempo real para satisfazer expetativas de consumidores e estafetas.

É desenvolvido um modelo de otimização para resolver o problema de atribuição com restrições de veículos, recorrendo aos algoritmos Jonker-Vogenant e Branch-and-Cut. O modelo matemático é inserido num *framework* dinâmico, que continuamente o resolve, enquanto controla chegadas de pedidos, turnos e atualiza posicionamentos. O modelo contempla o trânsito e limites de velocidade para diferentes tipos de veículos. Para além do modelo de atribuição instantâneo, é investigada a implementação de políticas – política de atribuição e política de bicicletas – para melhorar o desempenho em diferentes métricas.

Com base em instâncias reais, o modelo proposto considerando a política de atribuição alcança uma redução no tempo de entrega de, aproximadamente, 4,5% e um aumento de 9,6% na utilização equilibrada de estafetas em comparação com a atribuição real, melhorando a solução na perspetiva dos consumidores e estafetas. A política de bicicletas alcança um aumento de 3pp na utilização equilibrada de estafetas, à custa de um aumento de 0,3% no tempo de entrega em comparação com o modelo base.

Palavras-chave: *on-demand*, entregas instantâneas, atribuição de estafetas, problema de atribuição dinâmico, frotas heterogéneas, programação inteira mista.

Contents

List of Tables

List of Figures

Acronyms

AAM: Algorithm Adaption Method 30

ABM: Agent-Based Modeling 30

AGV: Autonomous Ground Vehicle 11, 15

AI: Artificial Intelligence 30

ALNS: Adaptive Large Neighborhood Search 29, 30

AP: Assignment Problem 19, 20, 22, 29, 30, 31, 32, 33, 34, 35, 38, 39, 41, 42

API: Application Programming Interface 5

B2B: Business-to-Business 11

B2C: Business-to-Consumer 6

BB: Branch-and-Bound 29, 33, 42

BC: Branch-and-Cut 41, 42, 48, 49, 50, 51, 53, 58, 59, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70

C2C: Consumer-to-Consumer 6

CG: Column Generation 30

DBSCAN: Density Based Spatial Clustering of Applications with Noise 25

GAP: Generalized Assignment Problem 32, 33

GH: Greedy Heuristic 30

HA: Hungarian Algorithm 34, 41, 50

ID: Instant Delivery 1, 2, 4, 7, 8, 9, 10, 11, 12, 13, 14, 16, 18, 19, 20, 22, 23, 24, 25, 26, 27, 29, 31, 32, 33, 34, 35, 36, 37, 69, 70

JV: Jonker-Volgenant 33, 41, 48, 49, 50, 51, 53, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70

KPI: Key Performance Indicators 38, 49, 50, 54, 55, 57, 58, 61, 64, 65

LP: Linear Programming 42

LS: Local Search 30

MDP: Markov Decision Process 30

MDRP: Meal Delivery Routing Problem 23, 24, 26, 29, 34

MILP: Mixed-Integer Linear Programming 38

ML: Machine Learning 30, 31

NP: Nondeterministic Polynomial Time 32

NSGA-II: Nondominated Sorting Genetic Algorithm II 30

O2C: Outlet-to-Consumer 12, 13, 14, 15, 17, 19

ODD: On-Demand Delivery 2, 3, 6, 12, 16, 24, 51, 69

OSRM: Open Source Routing Machine 45, 46

P2C: Platform-to-Consumer 12, 13, 14, 15, 17, 18, 69

PCA: Principal Component Analysis 30

PTM: Problem Transformation Method 30

R2C: Restaurant-to-Consumer 12

RL: Reinforcement Learning 30

1 Introduction

This chapter introduces the motivation behind the dissertation. Subchapter 1.1 contextualizes and justifies the subject of the study. Subchapter 1.2 defines the objectives of the study. Subchapter 1.3 details the methodology followed. Subchapter 1.4 presents the outline of the document.

1.1 Background and Motivation

The demand for fast delivery options has surged in recent years. The technological developments that followed the advent of the internet and the mass adoption of smartphones had a profound impact on the consumers' behavior and expectations. The same technologies connected businesses and customers, frequently cutting the middleman and allowing for the coordination of self-employed workers. Specially in urban areas, there has been a push for faster deliveries whose highest expression is the instant delivery (ID) that takes 45 minutes or less (Dablanc et al. 2017). Recently, the lockdown response to the Covid-19 pandemic made more people open to try IDs especially for meals, resulting, for some countries, in a seven time increase in sales (Ahuja et al. 2021) compared to 2018 levels. With the end of restrictions, the growth has slowed down, however the habit stuck and now a broader audience regularly uses these services.

The exponential growth IDs experienced in the last decade was sustained largely on the low investment required due to crowdsourced workers bringing their own vehicles. The other side of this coin is that couriers can choose schedules, reject requests and decide where to wait. The fleet is heterogeneous with vehicles with different carrying capacity, speed, range, susceptibility to congestion and restrictions to circulation. The literature concerning IDs is scarce and directed to the meal delivery niche. Most works do not contemplate heterogeneous fleets, or do so incompletely, and do not consider the dynamic nature of couriers' schedules or congestion. However, all these factors influence the real-life problem and must be pondered to ensure the best assignment decisions are made.

Given that the industry has not yet consolidated and competition for customers, partners and couriers is intense, having assignments that bring value to all parts involved is of vital importance to the short to long-term success of businesses. It is important to build models that generate results in near-real time for large instances of data, but also to study how various policies affect the often-conflicting objectives. The success in building better models has the potential to reduce costs to both the company and customers and ensure that the workforce is retained and motivated.

The present dissertation is propelled by a practical case study of an ID platform operating in London Metropolitan Area. The problem studied concerns the assignment of orders to couriers subject to vehicle restrictions, that is affected dynamically over the course of a day. Therefore, developing an optimization model to answer the aforementioned problem is the motivation behind this research.

1.2 Research Objectives

The primary goal of the dissertation is to develop an assignment model that can be generalized for most ID companies, considering vehicle restrictions and incorporating regional and dynamic elements to study the consequences of different policies. Therefore, the following secondary objectives are delineated:

- Comprehend the environment that surrounds IDs, the evolution of the industry and analyze trends that are likely to dominate the field.
- Formalize the concepts of ID and on-demand delivery (ODD).
- Identify agents, elements and features of instant delivery operations, as well as their relationships and trade-offs.
- Characterize the delivery process of the company that is the focus of the case study.
- Describe the challenges modelling ID in general and the case study in particular pose.
- Learn how similar problems have been modelled and solved in the past and what lessons could be adapted and applied to the problem at hand.
- Study the impacts different policies have on performance and key objectives compared to the current operation and provide recommendations.
- Analyze different scenarios and study the sensibility of the model to variations in parameters.

1.3 Research Methodology

A five-step methodology, represented in Figure 1, is followed to achieve the proposed objectives.

1. **Problem Context and Definition:** Based on ID companies' websites, newspaper articles, magazines, scholarly journals, statistical databases and companies' reports, contextualization of the problem within the delivery environment; definition of ODD, Same-day Delivery (SDD) and ID concepts; characterization of the operations of ODD platforms and of the case under study; and, identification of the main challenges.

2. **Literature Review:** Study common model formulations, objectives and solution approaches used for ID and similar problems. Complement this information by studying general approaches for solving traditional assignment problems with the aim of guiding the development of the model. Both these steps were performed by searching scholarly journals and databases, such as ScienceDirect, ResearchGate and Semantic Scholar, using as keywords: "on-demand delivery", "meal delivery", "assignment problem", among others. Further research was enabled by using the platforms Connected Papers and Litmaps, which suggested additional scientific papers based on the similarity with selected articles.

3. **Model Development:** Formulate the assignment model according to the specificities of the case study. Build a dynamic framework with Python to control the passage of time, make necessary updates and invoke a mathematical optimization software (Gurobipy or Scipy) to solve the model with the objective of reducing the total delivery time.

4. **Model Validation:** Test the model using real instances as input and compare results.

5. **Experiments and Analysis:** Perform computational experiments for the various models, policies and parameters and analyze the performance along multiple metrics. Critically discuss the results and derive conclusions that can add value to the case under study.

***Figure 1**. Dissertation's Methodology*

1.4 Document Structure

Following the research methodology, the contents of the dissertation were compiled into six chapters:

1. **Introduction:** The current chapter. Presents the background and motivation that led to the work, the objectives, the methodology employed to accomplish them and the layout of the document.

2. **Problem Definition:** Contextualizes instant deliveries within the broader on-demand economy, characterizes instant delivery operations in general and specifically for the case study and lists the main challenges in addressing and modelling the problem.

3. **Literature Review:** Describes how were ODD problems modeled in the past, which methodologies were applied by various authors in those problems and how are traditional assignment and routing problems modelled and solved.

4. **Methodology:** Formally describes the problem, presents a dynamic framework, and formulates the assignment mathematical model grounded on the literature insights.

5. **Experiments and Results:** Outlines the data treatment processes to obtain the inputs, compares the model's results to the real assignment, tests the sensitivity of the model to key parameters and analyses the effects of different policies.

6. **Conclusions and Future Research:** Summarizes the most important insights of the study, identifies its limitations and suggests directions for future research.

2 Problem Definition

This chapter characterizes the problem at hand. Subchapter 2.1 introduces the concept of ID and contextualizes it within the broader on-demand environment. Subchapter 2.2 further delves into the operational aspects surrounding this type of delivery. Subchapter 2.3 describes the case study and frames the model in question. Subchapter 2.4 presents the main obstacles and trade-offs, stating the problem under study. Subchapter 2.5 summarizes the main insights of this chapter.

2.1 On-demand Logistics

The term logistics has been widely used in military circles since the XIX century. With the end of World War II and the advent of the contemporary period, it started to be treated as a scientific subject and later gained the attention of the business world resulting in many accepted definitions. Among these are the "seven rights" of logistics, that consist in delivering the right product, for the right customer, at the right price, in the right quantity and condition, to the right place at the right time (Swamidass 2000).

On-demand, as it pertains to logistics, is intimately related with the above-mentioned time dimension, since customers expect to see their orders fulfilled soon after they are placed (Taylor 2018). As a result, on-demand is commonly applied to describe a fast distribution system with different timeframes depending on the industry, from SDD to IDs that usually take 45 minutes or less.

2.1.1 On-demand Economy

Logistics is but one component of the *on-demand economy* which should not be mistaken with the *sharing economy* – characterized by individuals getting temporary access to resources that they do not own – or with the *crowdsourced* or *gig economy* – used to describe the hiring of independent freelancers to perform single jobs –, even though these concepts might overlap (Belk 2014, Petriglieri et al. 2019). The interchanged use of these terms, although erroneous, is not accidental, the reason being that all three economies grew side-by-side and, in many cases, are inextricably linked. As an example, Uber would be unable to operate if their cars were not shared and their drivers crowdsourced. Figure 2 frames popular on-demand companies in relation to the gig and sharing economies and locates IDs as a subgroup of on-demand that can borrow elements from the other.

The desire to instantly receive a product or service is not novel. In the Inca Empire, *chasqui* runners delivered messages and gifts between cities, while in Mumbai the *Dabbawalla* has been used for more than 125 years as a meal delivery service. As it is conceived today, the on-demand economy is a new phenomenon. The first boom happened in the 1990s and included, among others, grocery and product delivery services such as Webvan, Kozmo or Urbanfetch. The boom, however, was short-lived and many unprofitable companies heavily dependent on outside investment crumbled with the popping of the Dot-Com bubble. Only a decade later would all factors be set for the thriving of the so-called on-demand 2.0

businesses. What differed between the two revolutions was, on one side, the changes in consumers and employees' behavior and, on the other, technological advances and leaner operations (Hackett 2014).

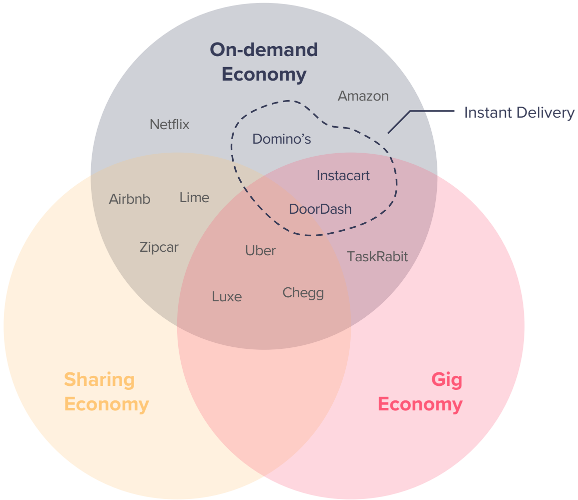

Figure 2. *Overlap between on-demand, sharing and gig economies.*

The key for understanding the behavioral shift that took place in less than a decade, lies in the embracing of the internet by the public at large. The internet was by no means unpopular by the turn of the century, with 442 million users worldwide; however, this figure is overshadowed if compared with the almost 2 billion users that were connected only 10 years later. Furthermore, the internet became more widespread within the countries themselves, with 58 countries having a share of their population superior to 50% online, when in 2000 only Canada and Norway passed that mark (Roser et al. 2015). The fact that this technology was not restricted to enthusiasts, but rather available to the public, allowed for almost perfect communication and a constant connection between individuals, as well as a much more direct channel between companies and customers. This trend was further exacerbated by the invention of the smartphone, that gradually took over as the dominant tool through which consumers interact with digital media, being responsible for 56% of current online traffic (StatCounter Global Stats 2022). The emergence of social media meant that more and more people flocked into the internet, and it became intrenched in the routines of most people in developed countries. This familiarity, coupled with easier and safer payment methods, brought trust to the on-demand services that would only be strengthened by real-time tracking and rating systems for both customers, businesses, and couriers (Dudas 2014a).

Competition among firms together with better processing power, Software Development Kits (SDKs) and Application Programming Interfaces (API) meant that the consumer experience would become increasingly convenient and efficient, which in turn would snowball into consumers demanding shorter and shorter fulfillment. In the case of logistics, the raw power of competition brought a more effective fleet management, smoother supply chain management and overall leaner operations (Dudas 2014b).

Meanwhile, some companies figured out how to tap into underused capacity by connecting people that owned unused resources with people interested in buying or renting them while creating value for both parties. The company would profit with added revenue or fees depending on the business model, the owner would capitalize on the asset, and the buyer or tenant would fulfill a need. Other companies realized that some resources were used by each person for a limited time and the company could provide them as a service, achieving a higher rotation on the underlying asset and benefiting from economies of scale. Consumers would get value in part from not having to invest in an asset that they would only use a handful of times. Both models, Consumer-to-Consumer (C2C) and Business-to-Consumer (B2C), became known as the sharing economy (Osztovits et al. 2015).

A significant cause for the success of this type of service were the changes in the labor market. In the 1990s unemployment was low, especially in North America, that was close to perfect employment, and where the first ODD companies popped up. As a result, businesses had a hard time attracting workers and had to resort to aggressive hiring practices offering a higher compensation, health insurance and retiring savings plans along with branded equipment. In the awakening of the global financial crisis, unemployment skyrocketed and only returned to 2000's levels around 2018 for OECD countries (International Labour Organization 2021). The conditions set greatly favored the crowdsourced work. Employers were able to offer high wages without other fixed costs usually incurred with fulltime workers, and people that were unemployed or wanted to have a second income gained a flexible part-time job.

All previous factors contributed, one way or the other, for the exponential development of the on-demand economy that, unlike the first iteration, appears to be robust enough to withstand future hazards, mainly because of the new habits cultivated and elevated expectations on the part of the consumers.

On-demand services rapidly grew to encompass almost every industry, to the point that being the "Uber of X" became a prevalent catchphrase used by entrepreneurs while pitching their ideas despite none reaching the same level of success, in part because not all fields of activity are equally relevant. For instance, online marketplaces account for 27% of all customer spending on the on-demand economy in the United States (US). Food delivery is the second largest piece with 16%, being the category that saw the steepest growth between 2016 and 2019 after retail deliveries, which now represent 10%. Transportation accounts for 14% and includes ride-sharing and ride-hailing, making it the fourth largest sector of on-demand, even though its growth has stalled in past years (Colby 2019). Logistics plays a big role on the on-demand economy, as evidenced by Figure 3, since food and grocery deliveries combined with transportation make around 44% of the on-demand pie. Moreover, online marketplaces normally require deliveries, further driving the point that the on-demand economy is heavily reliant on logistics.

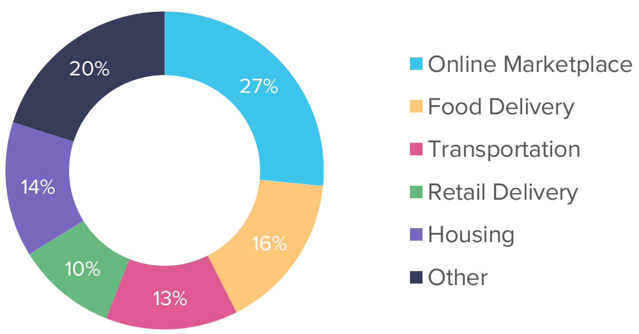

Figure 3. *Composition of the on-demand industry by sector in the US (adapted from Rockbridge, 2019).*

2.1.2 Instant Deliveries

The term instant describes fast deliveries that take less than 2 hours, though most are performed under 45 minutes, and some can even take just 15 minutes. Such stringent deadlines usually entail immediately start processing the order and having a dedicated courier, meaning fewer opportunities for consolidation. As previously remarked, the two categories that experienced a greater growth were food and retail deliveries. Not coincidently, these are some of the main sectors to which IDs have catered. To meet the customer's expectation, goods must arrive in appropriate condition meaning that the transportation must either be fast or require refrigeration or heating equipment, which is impractical and expensive. In the case of meals, it is almost mandatory that the delivery is done in minutes, to preserve not only temperature, but also flavor and texture. For groceries and other utilities, it is not evident that IDs are the optimal alternative. Even though every consumer prefers to have orders replenished sooner rather than later, there are other concerns that might take precedence over the fastest delivery. Despite quick deliveries being a priority for 40% of the grocery shoppers, considerably more people – around 60% –, view low or even no delivery fees as a more pressing issue (Coresight Research 2021). Not only that, but the preference for urgent delivery is also very age striated, with younger generations being more open to pay an increased price for a higher speed of delivery (Outbrain 2021).

Perhaps more importantly, the type of product is correlated with the preferred delivery timeline. This is mainly due to the fact that customers have varying urgencies for different goods. It might be acceptable to wait three days for furniture, but not for food. Figure 4 summarizes the results of a survey conducted in the US revealing that only a fraction of the respondents demand home deliveries within the hour; nevertheless, two categories – groceries and alcohol – stand out for a considerable preference for less than an hour and SDDs. The demand for quick deliveries of furniture, appliances, clothes and footwear is almost non-existent, but stationery supplies and health and beauty products are required in less than an hour by 5% (Salesforce Research 2021). This study should be taken with a grain of salt, since it was conducted in the US and reflects this country's reality, which has a big-box retailer culture and, as a result, tends to lag behind European countries when it comes to the fast delivery of such supplies. Despite the shortening up of distance in recent years, it should be expected that the preference for within the hour, same-day and next-day deliveries to be greater in European countries (Aull et al. 2021).

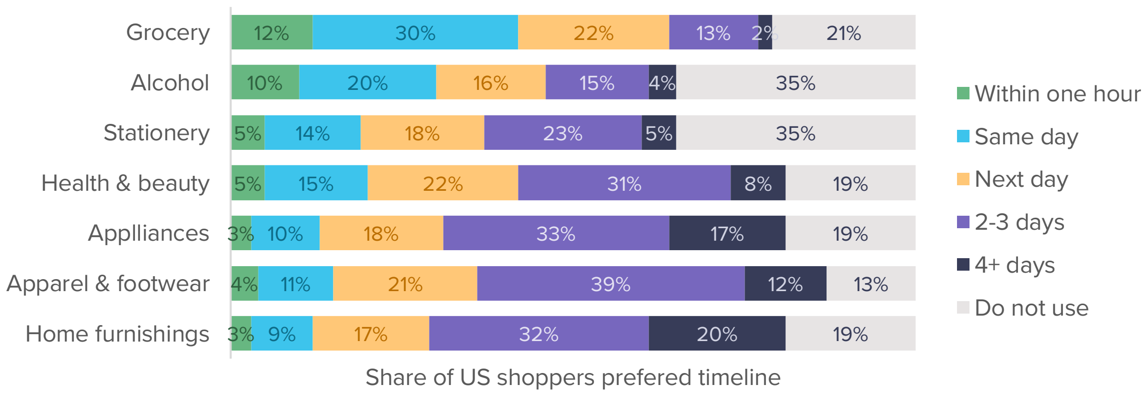

Figure 4. *Popular home delivery timelines worldwide, by category (adapted from Salesforce Research, 2021).*

Since its inception, the on-demand market has been expanding in phases, having started with e-commerce, being followed by streaming services and ridesharing, and finally IDs. Within this type of delivery different categories became more mainstream in stages, first with prepared meals then groceries and now with an uptick in convenience store items (Edison Trends 2021). This trend is reflected in Figure 5 that highlights the fraction of instant orders per type of item in the US. Even though all categories saw an increase during the one-year period, convenience store items supplanted the other two in terms of growth and now represent a small niche.

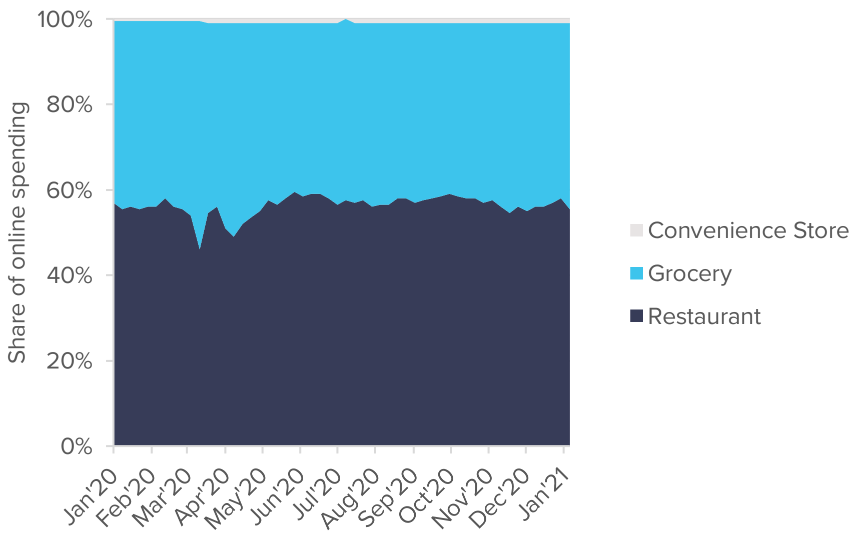

Figure 5. *Share of online spending in pickup and delivery by category (adapted from Edison Trends, 2021).*

Worldwide, the global food delivery market was worth more than USD 230 billion in 2020 and is expected to reach USD 450 billion by 2025 (Statista 2021). In comparison, the grocery last-mile delivery market had reached USD 25 billion by 2020 worldwide and it is still expected to grow at a faster rate, with forecasts predicting it to reach USD 72 billion by 2025 (Frost & Sullivan 2021). Despite having different orders of magnitude, in more mature markets, the contrast between meal and grocery delivery is much smaller and, as a result, it is expected to see an approximation of the two markets.

On-demand food delivery has seen an exponential growth during the last four years, being one of the major boosting factors the Covid-19 pandemic, which had impacting repercussions in worldwide economies. In fact, according to McKinsey, some of the most mature food delivery markets correspond to the US, Canada, Australia, and United Kingdom (UK) and recent data shows that they have experienced a growth between four to seven times higher in comparison to their size in 2018, as shown in Figure 6 (Ahuja et al. 2021). Although, this growth effect was intensely affected by the lockdown restrictions put into place due to the Covid-19 pandemic, it is believed that the food delivery industry is established in solid ground, keeping its growth trend steady after many countries have lifted lockdown measures and relieved restrictions.

2.1.3 Future Trends

The same behavioral and technological factors that aligned for IDs to appear in the first place are not set in stone, and current trends give a hint to where the industry is moving as well as suggestions worth considering incorporating into the model.

Behavioral and demographic

Younger generations are the main users of these services. As the demographic pyramid changes, it is expected that newer generations will be at least as much if not more reliant on IDs, meaning that with time a greater percentage of the population will demand these services. As younger people age and decide to build families, they tend to move to suburban areas where housing is cheaper. Compounding the remote working effect, further exacerbated by the pandemic restrictions, a move towards more remote areas is not only expected but is already manifesting and figuring out how to run IDs in sparsely populated areas is a major concern for the industry (Ahuja et al. 2021, Gorenflo 2017). Figure 7 highlights this trend between 2019 and the first quarter of 2021, with rural and especially suburban food deliveries taking a bigger share of transactions.

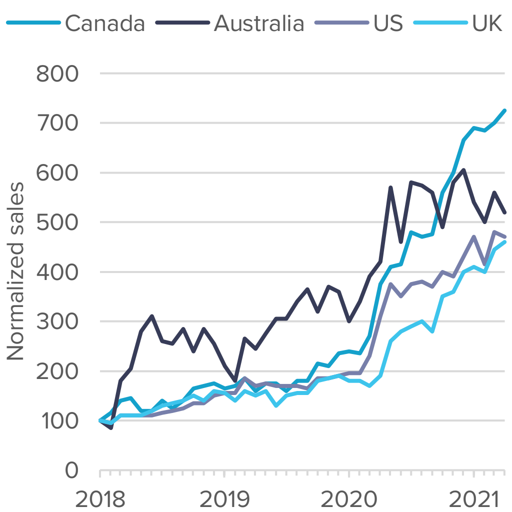

Figure 6. *Normalized delivery-platform sales growth (adapted from Edison Trends, 2021).*

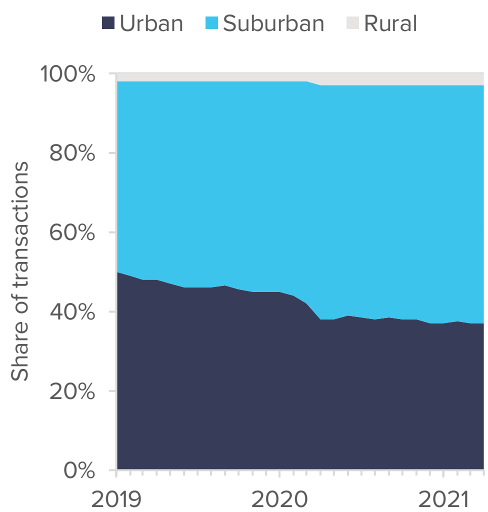

Figure 7. *Food-delivery transactions by population density (adapted from Edison Trends, 2021).*

Dark stores and kitchens

While platforms that connect restaurants and shops to customers are in a consolidation phase, a new phenomenon – dark stores and dark kitchens – has emerged. Dark stores are physical spaces that operate as micro-fulfillment centers. Unlike traditional brick-and-mortar stores, their goal is not to draw customers in, but, instead, to deliver the order as soon as possible. Companies like Getir, Gorillas or Gopuff promise to take 15 minutes from the moment a customer presses the order button, to deliver groceries and convenience products to their door, beating the traditional platform intermediary apps. This is only possible by having a network of centers dispersed through a city, ready to answer the call, and a well-optimized picking operation. On the other hand, dark or ghost kitchens are spaces that prepare delivery-only takeaway meals. They can be part of the sharing economy in some cases and allow for a restaurant to expand the reach for deliveries without the investment cost that setting-up a new site usually requires. Both trends are growing and can contribute for a more massive adoption of on-demand deliveries, even though some experts question their long-term success.

Moving warehouses

The push for shorter delivery deadlines is clear, with same-day parcel delivery being a reality in many cities and 12-hour delivery expected to become a staple in less than a decade (Huang 2019). Also, companies might want to offer IDs to customers willing to pay. Complying with such delivery windows can be done to some extent by companies with a large network of warehouses. However, for smaller retailers, the traditional parcel delivery method of having to visit every location poses a dilemma when it comes to delivering under two hours, as presented in Figure 8: (a) a single fully loaded van cannot visit all locations under the time limit; (b) dividing the parcels by a larger fleet can reduce the delivery time but drastically increases the operational cost. A solution that some companies propose (c) is having a “moving fulfillment center”, that integrates multiple couriers in a single delivery. This is done by assigning a van to bring the load from the client’s warehouse closer to the destination. Then, couriers usually driving two-wheeled vehicles can go and pick a smaller number of parcels that are delivered in short notice. This solution is all the more essential the further the retailer’s site is from the delivery zone, because the long distance is traveled only by the van and the final part is done simultaneously by various couriers, thus combining low-cost bulk shipping with quick delivery. The operation is also decentralized, since the fleet that goes to the moving fulfillment center can be used for the regular meal and grocery IDs which has the added advantage of being able to absorb peaks and troughs in demand by aligning deliveries of goods that run counter cyclical to each other (#42 - Bassel El Koussa - CEO @ Quiqup 2021).

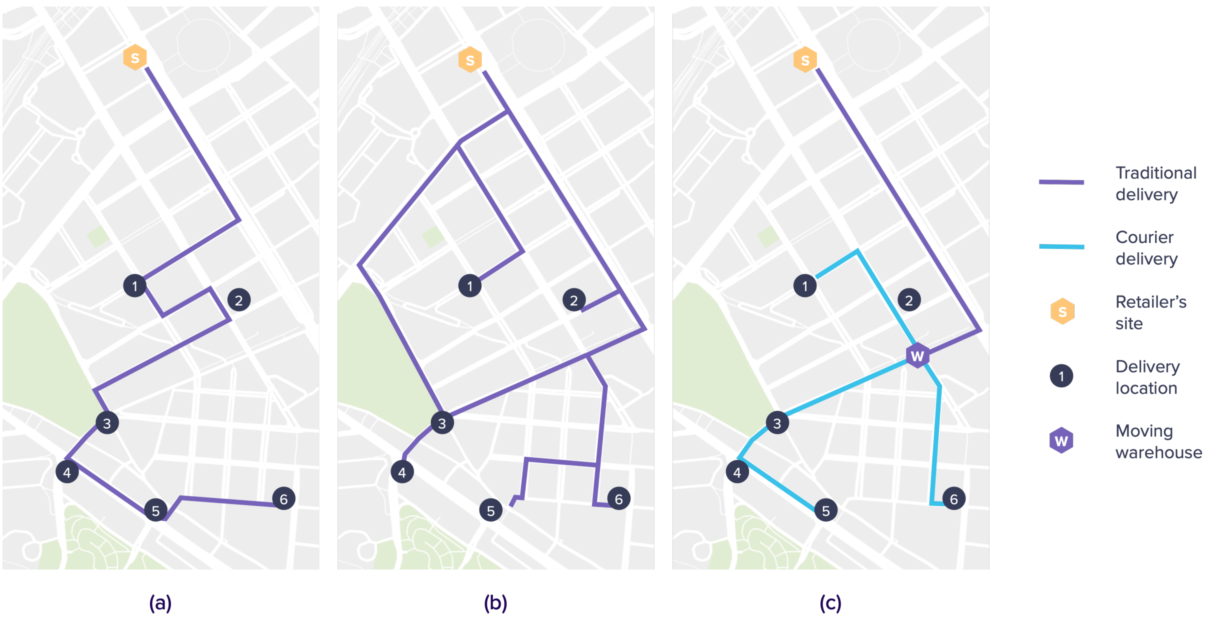

Figure 8. *Comparison of delivery systems: (a) Traditional; (b) Large fleet; (c) Van distribution center models.*

Automation

Automation is set to change all aspects of work and IDs are no exception, with many companies greenlighting Unmanned Aerial Vehicle (UAV), Autonomous Ground Vehicle (AGV) or droid delivery as proofs of concepts in some areas for limited uses. Nevertheless, some analysts speculate that, in the future, around 78% of all items will be delivered by autonomous vehicles (Joerss et al. 2016). According to the same analysts, only Business-to-Business (B2B) deliveries will follow the current model as shown in Figure 9. The use of UAVs, despite highly anticipated, is unlikely to become a reality for IDs and its deployment might be restricted to rural regions. Weather conditions, air space restrictions, required landing areas and the danger of hurting human beings are among the factors that might forever stall the mass adoption of these technologies. Moreover, at least in urban settings, there is the need to carry the goods to apartments, which is challenging for drones but trivial for couriers. Consequently, droids – for small parcel sized single IDs – and AGVs with lockers – for parcel sized or larger pooled SDDs – show signs of being a better substitute to couriers, or even being capable of working symbiotically alongside them in fulfilling on-demand deliveries.

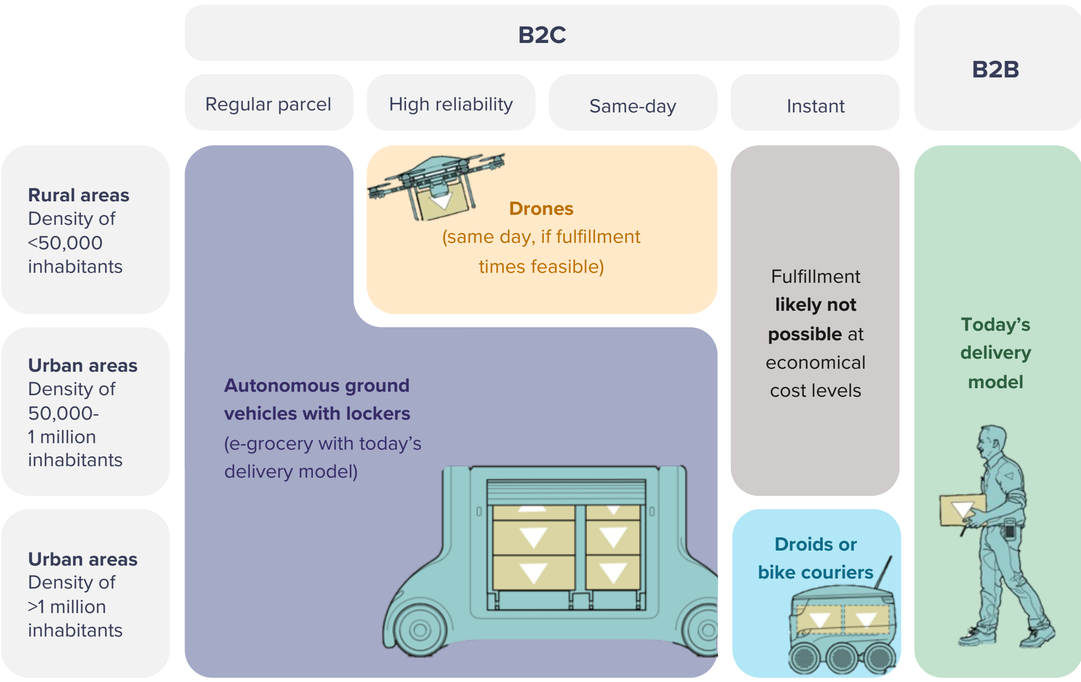

Figure 9. *Models likely to dominate last-mile delivery (adapted from McKinsey & Company, 2016).*

Legal factors

The legal framework in place plays a crucial role in determining the long-term success of any business. Concerning IDs there is a mix-bag of legislations that can either be viewed as opportunities or threats. On one side, there is a push for greener metropolis that, in many cases, entails limiting the circulation of heavy trucks and can be viewed as an opportunity for smaller and electric vehicles, such as scooters and bicycles, usually employed for IDs. On the other hand, there have been complaints associated with noise at late hours. If heavy-handed regulations are put in place, this can mean a shift for IDs and a factor that should be considered when assigning couriers by instead assigning electric vehicles or even bicycles at night. Labor is another nuisance that distresses especially crowdsourced businesses. There has been a push to consider drivers as employees rather than couriers connected to work by a platform. This, combined with the demand for higher salaries and the difficulty in recruiting new drivers, might be an impetus driving towards more automated alternatives.

2.2 Operational Model

The newness of IDs meant a first-mover advantage since the players that established themselves before everybody else could grow to dominate the industry in the future. Adding to this there were practically no barriers to entry, stemming from a lack of regulation and the inexpensive equipment required to run such a business. The attractiveness of the industry resulted in many competing firms entering and exploring different business models. The purpose of this subchapter is to shed a light on the different operational models, as well as to identify the commonalities between them. In the end of the subchapter, Table 1 is presented which summarizes the operational model of several ODD firms.

2.2.1 Business Model

Long before online deliveries were a reality, pizza and Chinese food were almost the only options available to receive at the doorstep. Such products became popular options for home delivery for being inexpensive, meaning that even incorporating a delivery fee on the final price still made them affordable to the consumer, but also portable and easy to pack. Businesses employed their own fleets and took care of the delivery, ensuring that the operation was carried out in an efficient and quick manner, to which the classical Domino's slogan – “30 minutes or it's free” - is attestant to. Not surprisingly, pizza chains were the first adopters of mobile applications, even before the crowdsourced delivery platforms, and by the late 2000s were already conducting around 20% of their orders through the online channel (Miller 2021).

The type of delivery employed by the pizza chains, where the restaurant handles the delivery, is sometimes referred to as Restaurant-to-Consumer (R2C) delivery. This does not mean that the order needs to be placed solemnly on the restaurant's app. In some cases, the restaurant might have a digital place in a third-party app for visibility, but the delivery will be carried out by the restaurant's fleet. For this dissertation, a broader term will be applied to cover both restaurants and grocery or convenience shops – Outlet-to-Consumer (O2C) – to describe the businesses that handle their own IDs. An alternative to this model is the Platform-to-Consumer (P2C) delivery, in which a third-party platform takes care of the delivery by either employing its own work force or by tapping into a pool of crowdsourced workers. This modality has experienced a significant growth in recent years and, in the case of online food delivery accounts for 64% of the revenue (Miller 2021). For groceries and convenience store deliveries, this figure is likely higher due to the lack of a previously established ID system. Figure 10 illustrates the two models: O2C where goods are transported using their own means and platforms are only used in some cases to facilitate ordering and transitioning; P2C where platforms are indispensable and act as an intermediary to connect couriers to orders that can be crowdsourced or not.

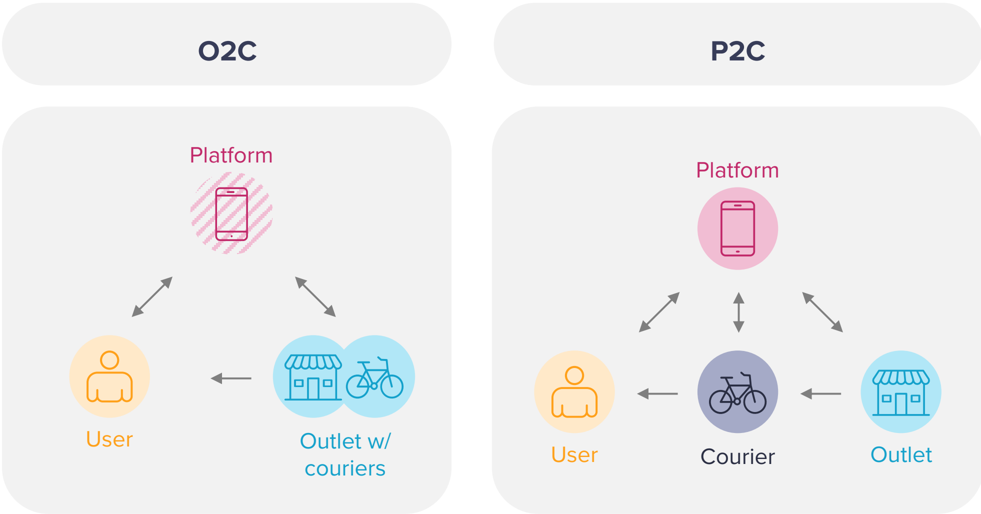

Figure 10. *O2C and P2C model structure.*

2.2.2 Pickup and Delivery

IDs require a fast pickup and delivery. As a result, there is little room for aggregating orders, which in turn means that the delivery share of the cost paid by the consumer is higher compared to traditional delivery. This also means that, at least under the current courier models, IDs are mostly restricted to individuals instead of bulk customers such as businesses that generally require larger volumes, hence the delivery point for IDs is predominantly a home. This should be considered especially in cities with tall buildings because couriers are often required to go up to deliver the loads, spending more time when compared with delivering at a house. On the other hand, having many apartments increases the chance for combining requests in the same trip if ordered around the same time. Furthermore, houses are common in sparsely populated areas, which increases the distance travelled and, consequently, the cost. The pickup point, by contrast, is more linked with the type of business. The couriers of O2C firms might always use the same pickup site or a small number of locations, while the carriers of P2C businesses must travel to different places, consuming more time especially if the rider is not familiar with a neighborhood.

The integration with IDs differs from outlet to outlet. O2C sites are frequently built from the ground up with deliveries in mind and, therefore, tend to be more efficient at streamlining orders through this channel than shops that take deliveries as a secondary job. This competitive advantage is one of the reasons why dark stores and dark restaurants have been gaining notoriety on the ID industry (Barnes 2019). On the opposite end of the spectrum are the P2C firms that must serve requests to all places available in the platform. Despite every place having an incentive to improve its operations, the reality is that, in many cases, these places are created with other purpose in mind, such as serving a family at a table or having people cover all the alleys on a store and cross the monthly grocery list. As a result, P2C platforms tend to be less efficient with the pickup when compared with O2C. This effect might be counterbalanced by the same platforms offering solutions and building relationships with key partner restaurants and shops responsible for a large sales volume.

2.2.3 Load

The type of cargo handled can either be homogenous – meaning that all items are equal or at least of the same type –, or heterogenous. In general, O2C businesses handle more homogenous products, while P2Cs handle a larger array of products despite the existence of many counter examples. The products demanded for IDs tend to be small and/or lightweight. Common items include meals, groceries, alcohol, clothes, medicine, beauty, office supplies, gifts and even personal couriering, which consists of taking messages or items from individuals to other individuals such as forgotten keys. There are also some companies that provide on-demand deliveries for larger than parcel size products, including furniture and appliances and there are even companies that provide bulk or freight transport on-demand.

Regarding the handling equipment, this might be supplied by the company – usually for O2C deliveries – or bought by the courier, which is more prevalent in P2C businesses. The most common transporting equipment are large backpacks. If the platform provides meal delivery, these bags are likely to have thermal insulation to ensure that food stays warm and that frozen products remain that way. For a traditional transportation operation, this type of insulation would be insufficient, however, since the delivery part of the order is performed in around 15 minutes, the thermal insulation is enough to prevent undesired melting or cooling (Ahuja et al. 2021, Simonite 2019). Some additional equipment might be used, such as storage compartments in scooters or a front basket in bicycles. The packaging is the responsibility of the outlet but usually the product is presented in a box or in a bag which the only handling restriction is being moved with care and with the correct side up.

2.2.4 Vehicles

Once more, the means through which deliveries are carried are dependent on the business model of the company. For instance, the question of the ownership of the vehicle, most O2C platforms opt for employing their own fleet, while P2Cs often rely on couriers to provide their own vehicles.

Another poignant decision that businesses must come to terms with is the range of transportation means to employ. Some companies choose to only use bicycles and electric vehicles since low emissions and sustainability are core parts of their visions. Other companies only employ bicycles and motorcycles for the increased flexibility with parking, while others might still employ all sorts of vehicles for different types and quantities of cargo. The most common vehicle today for quick deliveries is the scooter, however, this does not mean that other means cannot be used; in fact, most P2C companies employ a wide range of transports, including bicycles, cars or scooters, and some can even use commercial vehicles. Other aspects of the delivery operation favor one type of vehicle over another, such as the fuel consumption. In cases where the deliveries are small and can be fulfilled by bicycles, cars or light commercial vehicles will be at a disadvantage. However, if the order is large enough or if orders can be grouped together, the larger capacity of automotives and vans can make this method of transportation more appropriate. By contrast, two-wheeler vehicles have a significant advantage in their favor for the increased flexibility, ability to access narrow roads and easier parking.

As previously mentioned, it is widely accepted that drones will have a crucial role on the future of the industry. Most established players and some new entrants have invested in the research and development of drones capable of performing home deliveries. Starship Technologies is an example of a company that is pioneering deliveries with droids, firstly in university campuses and now in some cities (Geha 2022). For now, the droids are set to go from a specific store to homes and are not decentralized in the sense of being able to answer calls from multiple establishments, but that might change in the future. Another limitation of droids is the limited carrying capacity and the fact that it can only serve a customer at a time. The AGVs with parcel lockers are an alternative that solves the referred problem, since they can carry orders from different customers in the same trip. This technology is not new and has been tested in the last couple of years, being Ocado, an online grocery retailer, one of the first to test AGVs for deliveries with the CargoPods (Boxall 2017). With the lockdowns at full swing, companies like Meituan, JD.com and Neolix greenlit the testing of such technologies in Beijing, while Domino's Pizza launched its own self-driving robot in Houston (Liao 2021, Lyons 2021).

2.2.5 Couriers

In the future, automation will likely play a role in deliveries but, for now, the people that move the goods from pickup to the delivery point are indispensable. Couriers can be employees of the company, more prevalent for O2C, or be self-employed or crowdsourced workers that use the app, more common for P2Cs. In this case, the workers are not directly employed by the firm, even though the legal designation can change in future years. Contrary to popular belief, most gig workers of the on-demand economy are part-time workers that use these jobs to get a supplementary income and not as a primary source (Gorenflo 2017). Delivery platforms became an interesting option by providing the flexibility of schedule.

Platforms assign orders to drivers in different ways. To ensure that the delivery is fast and efficient, most do not allow drivers to scroll and pick their favorite option, but rather connect an order and send a notification to accept or reject it (Reuters 2016). Apart from this, it is common practice among platforms to employ rating systems to all sorts of components of the service, to which the drivers are no exception. For most P2C companies, when proximity to the pickup point is comparable, the scores of the riders, based on past deliveries, will determine who gets the order. There are multiple aspects that might factor in the algorithms of each company. Policies like surge pricing during demand peaks, driver's tips or the means of transportation used by the driver can all be part of the assigning decision. The extent to which the driver is aware of the journey ahead can also vary from company to company, but usually drivers have limited information regarding the order. As an example, DoorDash only shows their drivers the restaurant's name, the order total and the tip placed by the customer. Such procedures have the goal of reducing the bias that drivers might have towards certain orders.

2.2.6 Delivery Options

On-demand deliveries are, undoubtedly, the main area of expertise of online ID companies, which does not mean that they are the only type of service provided. A delivery option common to most platforms is the scheduled delivery. The difference from the previously analyzed alternative is that the customer places the order in advance. The specific delivery option varies by company, but usually ranges from an hour and can go up to a week in advance and a short delivery window, around half an hour is provided. This type of delivery is convenient for certain people that might have tight schedules or do not want to be distracted before mealtime. Whichever the reason might be, scheduled deliveries are even more welcomed by the companies that usually are restricted to using surge pricing or prediction models, as with scheduled deliveries they can get an actual picture of a part of the requests, thus having a better ability to match supply and demand (Panneerselvam 2021).

The time windows when orders are placed are not evenly spread out. Restaurant orders spike during mealtime, while grocery requests concentrate after working hours (Blake 2022). Combined with the fact that drivers, in most cases, do not have a fixed schedule, this means that at times there might be a shortage of workers. One work-around solution, that also optimizes the travelled distance per delivery, is to aggregate different orders into a single delivery with multiple stops, also known as pooling. The orders can all be from the same place, or from different outlets that are geographically close to each other. The number of orders that can be pooled differs with all sorts of factors, like the relative size of the order and the capacity of the vehicle/equipment, the type of load in question, the ability to create a route or the relative distance between the points (Li et al. 2022). Furthermore, certain loads cannot be in transit over long periods of time, as food may get cold or defrost, hence meal delivery pooling tends to be restricted to just two orders. Pooling orders in IDs has the caveat of displeasing customers if the waiting time is long or if the product is not in the best conditions. As a result, drivers can get blamed unfairly and get a bad review. This jeopardizes both the loyalty of the customer and the will of the courier to accept these orders.

Another feature that has recently came to ODD platforms is the ability to make an order take precedence over the other, with the priority functionality. With this setting, a customer can pay an extra fee and ensure that the order will be delivered in less time. This means that the order will not be batched or, in the case of it being batched, it will be the first in line to be delivered (Labuschagne 2022).

Table 1. Operational model description of on-demand services and companies.

Name	Operating Area	Range	Year	Business Model	Unit Size	Type of Goods	Vehicle	Couriers	Delivery Time	Reference
Amazon Fresh	US, Europe, Asia	Urban	2008	O2C	Parcel	Grocery	Truck	Employed	Two-hour, same-day, next day	Dablanc et al. (2017)
Amazon Prime Now	US, Europe, Asia	Urban, Suburban	2014	O2C	Parcel	Retail, grocery	Bicycle, car, scooter, van	Employed	Two-hour, same-day, next day	Porter (2021)
Bolt Food	Europe, Africa	Urban	2019	P2C	Parcel	Meal	Bicycle, car, scooter	Crowdsourced	40 minutes	Bolt Food (2018)
Convoy	US	Urban, Suburban, Rural	2015	P2C	Pallet	Bulk	Freight truck	Crowdsourced	Variable	Convoy (2021)
Deliveroo	Europe, Asia, Australia	Urban	2013	P2C	Parcel	Mean, grocery	Bicycle, car, scooter	Crowdsourced	40 minutes	Deliveroo (2022)
Domino's	Global	Urban, Suburban	2007	O2C	Parcel	Pizza	Scooter	Employed	30 minutes	Domino's Pizza (2021)
DoorDash	North America, Germany, Japan, Australia	Urban, Suburban	2013	P2C	Parcel	Meal, grocery, convenience, alcohol, pets, retail	Bicycle, car, scooter	Crowdsourced	40 minutes	DoorDash (2020)
Ele.me	China	Urban	2008	P2C	Parcel	Meal	Bicycle, scooter	Crowdsourced	40 minutes	Zhang and Hu (2020
Getir	Europe, US	Urban	2015	O2C	Parcel	Grocery	Electric bicycle, scooter	Crowdsourced	15 minutes	Lomas (2021)
Glovo	Global	Urban	2011	P2C	Parcel	Meal, grocery, convenience, alcohol, retail, personal couriering	Bicycle, car, scooter	Crowdsourced	40 minutes	Glovo (2017)
Goodfood	Canada	Urban, Suburban	2014	O2C	Parcel	Meal kit, grocery	Truck, van	Employed	30 minutes	Goodfood (2020)
Gorillas	Europe, US	Urban	2020	O2C	Parcel	Grocery	Electric bicycle	Crowdsourced	15 minutes	Faoite (2021)
Instacart	North America	Urban	2012	P2C	Parcel	Grocery, alcohol, office and home supplies, beauty	Bicycle, car, scooter, van	Crowdsourced	One-hour	Instacart (2020)
Pickup	US	Urban, Suburban	2015	P2C	Large Parcel	Retail, furniture, appliances	Pickup truck	Employed	Same-day	Prevost (2018)
Pizza Hut	Global	Urban, Suburban	1994	O2C	Parcel	Pizza	Scooter	Employed	30 minutes	Pizza Hut (2021)
Starship	US, Europe	College Campus	2014	P2C	Parcel	Meal, drinks, stationary, tools	Droid	-	40 minutes	Starship (2021)
Swyft	Canada, US	Urban	2020	P2C	Parcel	Retail	Truck, van	Employed	Same-day, next day	Swyft (2021)
Trusk	France	Urban	2016	P2C		Furniture, appliances	Truck, van	Employed	Two-hour, same-day, next day	Trusk (2021)
Uber Eats	Global	Urban	2014	P2C	Parcel	Meals, groceries, convenience, alcohol, health & beauty	Bicycle, car, scooter	Crowdsourced	40 minutes	Lomas (2020)
Uber Freight	North America, Europe	Urban, Suburban, Rural	2016	P2C	Parcel, Pallet	Bulk	Freight truck, pickup, van	Crowdsourced	Variable	Uber Freight (2021)
Zipline	Africa, US, Japan	Rural	2014	P2C	Parcel	Blood, medicine, retail	UAV	-	15 minutes	Zipline (2019)
Zoom	UK	Urban, Suburban	2018	O2C	Parcel	Grocery	Bicycle	Employed	One-hour	Bourke (2021)

2.3 Case Study

The case under study concerns a P2C delivery company operating in London Metropolitan Area. The platform enables customers to order from partner restaurants and stores and also provides personal couriering services. Requests are picked up and delivered by crowdsourced couriers that use their personal vehicles. This results in a heterogeneous fleet where couriers have different carrying capacities, speed, susceptibility to congestion and parking or accessibility limitations. After a delivery, couriers decide to relocate to another part of the city or to wait in the same place. Couriers are also able to choose their working hours by selecting a shift, accept or reject assigned jobs, or to follow the platform's advice and move to areas with higher demand. Figure 11 represents the initial location of the company's couriers for a day in April 2018. The image highlights that most couriers concentrate on the inner city and, in the outer city, motorized vehicles, especially cars, are more prevalent. In the case of London, only couriers with bicycles, cars and scooters operate, despite the platforms also allowing cargo bikes and vans.

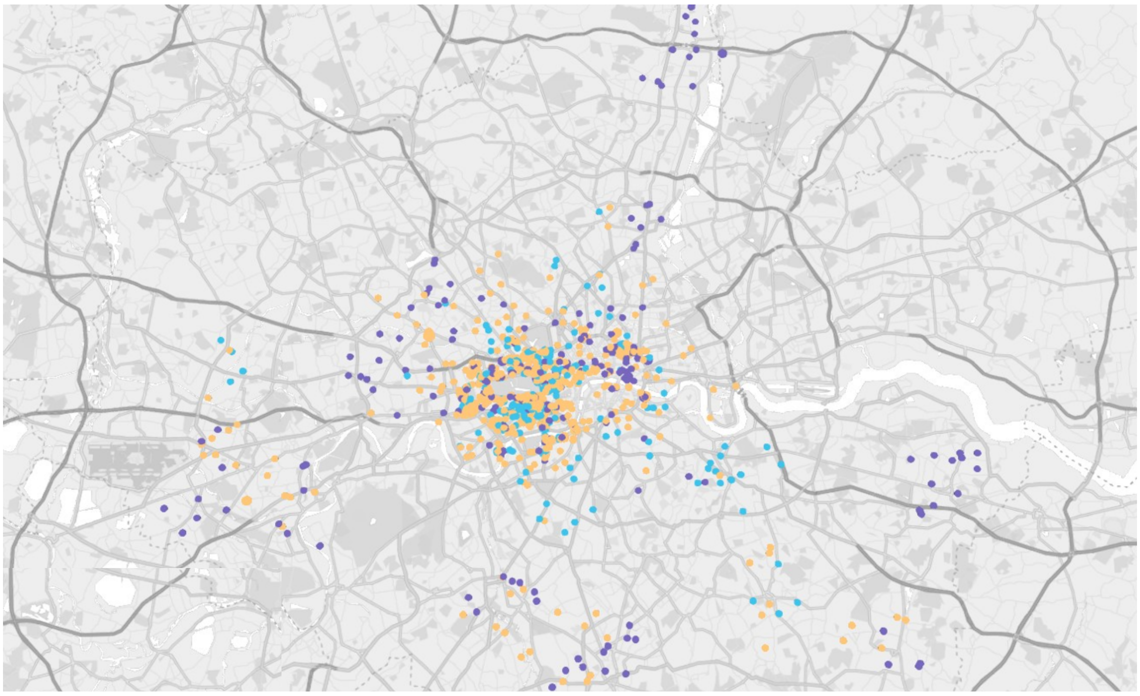

Figure 11. *Map with bicycle (blue), car (purple) and scooter (yellow) courier's initial location.*

In terms of the loads that can be transported, there is a lot of variety including meals, groceries or retail items. There is also the option of transporting larger items or having large orders with many identical items that might require a special vehicle. The company automatically restricts the vehicle types that can handle certain orders based on the vehicle type, *e.g.*, large orders are restricted to cars or even vans, and on the availability of parking space, *e.g.*, if parking is problematic in the pickup or drop of an order it is restricted only to bicycles or scooters.

When it comes to deliveries, both on-demand and scheduled options are allowed. The company operates in an urban setting under the same-hour for the on-demand IDs. The scheduled deliveries can also be instantly fulfilled or can be bundled with other orders and delivered in a more efficient route. Pooling orders from multiple customers is allowed even for IDs, but there are restrictions when it comes to bundling hot and cold orders and in delivering all parcels in a reasonable time, which in turn means that few instant orders can be grouped together.

2.4 Operational Challenges

The fundamental shift from traditional to on-demand logistics, together with the relentless competition, constitute a need for smooth and efficient operations (Ahuja et al. 2021, Kammerer et al. 2020). Some of the major constituents of such operations are the assignment of couriers to orders and the generation of an efficient route that guides the courier to the destination. These two phases represent in-and-of-themselves large problems and each carry their own idiosyncrasies, described in this subchapter.

The Assignment Problem (AP) consists of deciding which courier should be allocated to a given order (Bozanta et al. 2022). At first glance, the problem might look relatively straightforward, however, factors such as the pickup and delivery location, size of the product, vehicle type, traffic limitations and courier availability at any time introduce complexity to this process. Contrary to traditional logistics operations and even O2C deliveries, the fleet is neither homogenous, constant or defined *a priori*. This means that not only orders should be ascribed to vehicles based on their features, but also this attribution might be conditioned by the number or couriers logged in the system in a specific moment.

One of the main factors influencing the assignment is the geographical proximity to the pickup location, with couriers closer to the depot having an advantage. However, the delivery location or rather the distance between the two points can alter this, because some vehicles, such as bicycles, are not as fast and, if not powered, rely on the couriers pedaling which might result in delays or rejections of orders by both customers and couriers. Even though long ranges can mean a slower delivery by bicycles, for short distances the opposite can happen. This is due to the higher versatility of two wheelers, that can easily bypass car traffic and even use cycle tracks that, in some cases, are open for scooters as well. Moreover, in historic cities, the roads were not built for traffic and can be hard to access with large vehicles, giving an advantage to bicycles that must be contemplated when choosing the courier. The same can be said for parking in highly dense areas, since the time to immobilize the vehicle is viewed as a delay by the customer and can mean an extra cost that the courier is not willing to incorporate.

Despite most orders being small, the dimension of a product or the size of an order must be taken in consideration. Bicycles or scooters have a small capacity, mostly correspondent to the couriers' backpacks, and are more prone to having the individual items of an order clashing, which can damage the product. For this reason, cargo bikes, car or even vans are more appropriate for large items that exceed the maximum volume of the backpack and for large orders with many items.

External factors imposed by different cities also play a role. This includes limitations on the circulation of gasoline-powered or diesel-powered vehicles in some areas or even noise restrictions after dusk that favor the use of bicycles and electric vehicles.

Every method of transportation has its own pros and cons, with no vehicle being superior in all aspects. Figure 12 represents the performance of different vehicles in relation to five metrics relevant to the delivery problem. Bicycles and vans end up being the antithesis to each other, with all other options having some sort of compromise between flexibility and range/capacity.

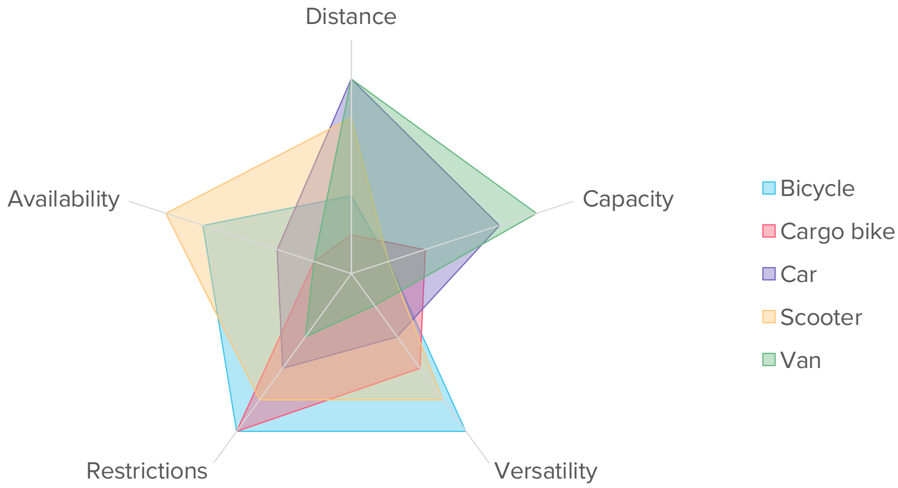

Figure 12*. Performance comparison by vehicle type.*

Crowdsourced couriers constitute a factor of uncertainty since they have the ability to choose their shifts, vehicles and can reject orders. This means that the number of vehicles available at any given time is variable and not fully controllable by the company. Furthermore, to guarantee a good service level, a rating system is employed that also plays a role in the assignment of a courier. According to this system, couriers with a higher rating have priority when choosing their shift to work. Apart from this, the total pool of couriers that can be called into action includes not only idle couriers, but also those that are completing an order and, in some cases, it can be better to assign the closer courier even if not currently available. All mentioned factors play a role on allocating couriers, but it should be noted that matching the supply of couriers to the demand of requests is not always ideal, meaning, for example, that in some cases couriers on bicycles can be tasked with delivering over long distances and some orders can even be unfulfilled, which has a negative impact on the customer.

2.5 Chapter Conclusions

The advent of the internet age revolutionized all aspects of life, to which logistics is not an exception. Consumer behavior fundamentally shifted and constantly demanded faster deliveries. New and unique models, though often misnamed, emerged to try and respond to a dilemma that traditional logistics had no answer to. One such solution are IDs that grew to prominence in the last decade, especially during the Covid-19 pandemic. This type of delivery, as it is today, is better suited for the distribution of small and perishable goods in urban settings, where the constant flow of orders allows for a large and decentralized fleet of vehicles with small capacity each. An enlargement of the geographical area covered is expected, with more people in rural areas demanding the service, which puts extra stress on the supply chain. Although the incorporation of automated robots might alleviate some of the issues, improving the operations and designing superior alternative delivery models is crucial for the long-term success of a business in such a competitive environment.

Despite delivery tasks being straightforward – moving goods from point A to point B –, the reality is that there are complex processes behind the operations of decentralized platforms. These platforms have all sorts of variability when it comes to couriers, vehicles, load capacity, time-windows, pooling, priorities and so on. This means that various, often contradicting aspects factor in on the assignment decision and the crafting of the best route. In summary, the goal of the present dissertation is to develop a method to assign couriers to orders based on proximity to the pickup, distance to the delivery location, dimensions of the order, type of vehicle, parking and road limitations and courier availability. The developed method with be applied to a realistic case study from a company operating under the operation model mentioned.

3 Literature Review

This chapter presents an overview of past scientific contributions relevant to address the problem at hand. Subchapter 3.1 examines the *plethora* of on-demand problems and identifies the most significant for the context of delivery. Subchapter 3.2 analyses the main features of ID and SDD problems as well as the objectives and solution methodologies used. Subchapter 3.3 is dedicated to the classic AP and the solution approaches commonly applied. Subchapter 3.4 summarizes the main insights of the chapter.

3.1 On-Demand Delivery Problems

IDs are remarkably distinct from conventional distribution, meaning that traditional delivery problems cannot be applied. On the other hand, being a new field of study means that there is a lack of uniformity, evidenced by the absence of a clearly defined general purpose ID problem.

(Cordeau et al. 2007) addressed the dispatching of urban couriers, emergency vehicles, aircrafts and dial-a-ride vehicles for elderly people and labeled them Transportation On-Demand (TOD) problems, but the nomenclature was not widely adopted. Instead, researchers have focused on niches of on-demand such as ride-hailing (Bertsimas et al. 2019, Masoud and Jayakrishnan 2017), crowdsourced delivery using excess capacity (Arslan et al. 2019, Dayarian and Savelsbergh 2020), e-commerce (Ulmer and Streng 2019), meal delivery (Reyes et al. 2018, Steever et al. 2019), grocery delivery (Fikar et al. 2018), personal shoppers (Arslan et al. 2021) or SDD (Zhou and Lin 2019) problems.

For the most part, IDs have been studied in the form of meal delivery problems. These problems capture many real-world aspects, but usually let other factors such as order and fleet heterogeneity unexplored. For that reason, both ID mostly applied to the context of meal delivery and SDD problems are considered.

3.2 Instant and Same-day Delivery Problems

In this subchapter, both ID and SDD problems are described according to the special features each model presents, the objectives chosen and the methods employed for reaching a solution. The categories selected are loosely based on existent taxonomies for dynamic problems, but also include other aspects innate to on-demand problems (Ojeda Rios et al. 2021, Psaraftis et al. 2016).

3.2.1 Problem Features

The features used in each model vary significantly and are formulated not only as constraints, but also as criteria that must be satisfied in the objective function. The various features are divided according to the three categories – physical, scenario and information – proposed by Eksioglu et al. (2009).

Physical Characteristics

Theoretically, IDs do not necessarily require multiple pickup points but, in practice, almost all problems consider more than one depot, while for SDDs the opposite happens.

Platform based courier deliveries have only recently started to be studied, with the publishing of the Meal Delivery Routing Problem (MDRP) (Reyes et al. 2018), meaning that most models overlook many of the peculiarities of these services. This is evident when it comes to orders, where only three authors have explored models that incorporate order heterogeneity. Liao et al. (2020) considers that single orders vary between one to five boxes, meaning that the size of the order plays a role in the assignment process, especially because not all vehicles have the same capacity and batching is allowed. Liu (2019) goes one step further and incorporates both weight and the type of meal – hot or cold – that cannot be transported together. This study is based on drone delivery and weight is preferred to volume because the payload significantly affects battery consumption.

There is considerable literature regarding assignment and routing of heterogenous fleets. Most of these studies consider vehicles with different capacities and costs (Gendreau et al. 1999). In the case of IDs, limits on capacity are usually expressed in number of orders that a vehicle can transport since most researchers do not take into consideration the size of orders, however, if the order composition is considered, they can also be expressed in volume or weight. Liu (2019) incorporates not only the maximum carrying weight of various drone models but also their velocities. Ulmer and Thomas (2018) also considers speed by having drones that travel in a straight line at high speed and road-based vehicles that move at slow speed. This is similar to X. Chen et al. (2022) approach, that multiplies the Euclidian distance by a constant to account for street networks and traffic. Fikar et al. (2018), despite not incorporating speed into the model, explores the integration of vans and cargo bikes considering that the two differ in capacity and range, as well as even considers areas restricted to vans.

Time windows mean that a service must be performed within a given interval and are categorized as hard – if impossible to violate – or soft – if non-binding and penalized through the objective function (Kallehauge et al. 2005). In the case of on-demand deliveries, it can be associated with pickup, delivery or with couriers in the form of shifts. In this last case, hard constraints are used (Tu et al. 2020). For customer orders, time windows are mostly soft, with penalties for lateness and bonuses for early arrivals (Li et al. 2022, Liao et al. 2020), but can also take the form of a hard deadline (Ulmer and Thomas 2018) attributed to each request that ought not be infringed. One article compared a fixed delivery interval with a tailored window based on the maximum expected preparation time and concluded that having realistic estimates does not compromise system efficiency (Steever et al. 2019).

Most of the selected articles attempt to mimic the effect of crowdsourced drivers, most commonly in the form of shifts or more rarely by allowing couriers to enter or leave the system at any moment (Fikar et al. 2018, Snoeck and Winkenbach 2022). Furthermore, a significant portion of studies account for limits on the amount of work a courier can perform. Zhou and Lin (2019) use a hard limit on the individual courier working time. Auad et al. (2022) considers a hard limit that blocks couriers from changing between areas as a proxy for distance. Other authors provide soft measures to ensure that workload is balanced (Yildiz and Savelsbergh 2019). Wang et al. (2022) defines an index of unbalance based on the standard deviation of the number of assigned orders to couriers, however this classification is not universal.

Travel time is affected by external conditions and some studies have captured this effect by dynamically update travel time to simulate traffic and weather conditions (Steever et al. 2019, Tu et al. 2020).

Scenario Characteristics

IDs demand that orders start being processed as soon they as placed. Most platforms allow not only this modality, but also scheduled deliveries which feed information to the platform that can be used for managing capacity. Nevertheless, this policy has not been extensively covered, especially in conjunction with courier management. Liu (2019) and Voccia et al. (2019) consider the ability of the customer to provide a delivery window to receive the order; however, both problems have fixed fleets, which means that the platform cannot provide incentives for more couriers to join the service and, as a result, the only effect in study is the batching, assignment and routing and potential changes. Tu et al. (2020), despite having a variable crowdsourced fleet and scheduled deliveries, does not change the entry rate of new workers based on demand. However, this study applies pre-dispatching constraints that prevent couriers from being concentrated in the same region by instead serving requests from more distant outlets.

For the majority of ODD problems new orders can arrive at any moment. This means that the dispatching and subsequent routing can suffer alterations. Most studies consider a cut-off time and after the delivery of a bundle has started no changes are allowed (X. Chen et al. 2022, Voccia et al. 2019). Other authors formulate their problems in a way that allows for diversions albeit with some restrictions. This means that diversions are not allowed while the courier is driving (Ulmer et al. 2021) or when it is driving to a restaurant for picking up an order (Steever et al. 2019).

Preposition consists in the ability of couriers to travel to a pickup location upon a customer request and wait for the order to be prepared. Almost all ID articles account for this policy, as well as some SDD problems. For instance, Bozanta et al. (2022), based on a proactive model, allows idle couriers to go to a restaurant if there are no new orders entering the system, instead of waiting at their current location.

After delivering a parcel, the various models have different perspectives on what should the courier positioning be. Some authors require that the courier returns to a base, which is more common in SDD problems, while others consider that the driver remains in the same region and others still give couriers the ability to relocate. The policy to return to base is generally associated with the existence of a single depot, which is typical for SDD problems or for cases where autonomous vehicles that must be recharged at a station are used (X. Chen et al. 2022, Ulmer and Thomas 2018). Allowing the couriers to relocate is a staple of on-demand platforms with crowdsourced couriers, but it is seldom addressed in the literature, likely due to the extra complexity introduced and the fact that, often, couriers must wait for orders to be processed and the reduction in travel time to the pickup location would have a meager effect on the overall solution. Nonetheless, a few authors explore this feature from different angles, namely Ulmer et al. (2021) that directs a driver to the nearest empty restaurant after delivering all orders assigned to him. The restaurant is only viewed as empty if no other idle courier is stationed there, meaning that couriers are distributed across multiple restaurants if orders are scarce. The author suggests that other strategies for relocation should be studied in future work. Auad et al. (2022) proposes a slight variation to the previous policy by repositioning couriers to the closest pickup point in the courier's service area.

Employing crowdsourced drivers implies that couriers can reject orders. Reyes et al. (2018), despite not including this feature in the first MDRP, points to this strategy for future research. Since then, a number of authors have explored this ability. Tu et al. (2020) considers that orders are rejected automatically if a courier decides to leave the system, which is a shallow representation of the real-world ability to reject an order. Bozanta et al. (2022) proposes a more realistic depiction of the feature by enabling rejections after the assignment and using a machine learning algorithm to understand whether neglecting some orders, and accepting a penalty, can have an overwhelming benefit on the overall performance. In the conducted study, there was a slight improvement derived from including this strategy, but it should be noted that all results are reliant on the value of the penalty that can be hard to set for a practical setting. For SDD problems, rejections are also considered but in the form of centralized decisions in the depots, rather than particular to each courier (Ulmer and Thomas 2018, Voccia et al. 2019).

One of the most prevalent strategies used in both problems is pooling, also referred to as batching or bundling. Most SDD problems have a single depot and, as a result, aggregating orders from dispersed customers is a matter only of delivery. On the contrary, IDs are dependent not only on the proximity of customers, but also of the sources of goods. Arslan et al. (2019) study the use of crowdsourced workers, allowing them to specify the number of stops to be included in a single trip which is used as the limiting factor to the number of requests that can be grouped together using a constraint. Li et al. (2022) applied a Density Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm first proposed by Ram et al. (2010) in order to aggregate orders. The algorithm is fed information about the location of the pairs of customers and restaurants by the platform and connects the high-density duos in a neighborhood. Each courier is assigned to a region and is not allowed to trespass it until all requests are attended. Liao et al. (2020) uses a k-means clustering learning algorithm that evaluates the distance between two data points. A small distance implies a high similarity, resulting in the two objects being more likely placed in the same cluster. Reyes et al. (2018) implements a strategy where an ideal bundle size is pursued. This target can be set beforehand or can be adjusted dynamically based on the number of couriers and orders. The orders are then pooled together aiming at the target. The algorithm allows for more orders to be inserted above the maximum threshold of a bundle only if it improves the overall route efficiency. Despite expanding on the previous work, Yildiz and Savelsbergh (2019), formulate the bundles served as an element of a work-package. In the first phase, only individual orders are considered, while in a second phase attempts are made to aggregate batches whose efficiency is then compared with the initial solutions obtained for the single order scenario.

While pooling strategies attribute many orders to a single courier, the reverse can also occur. Having multiple couriers per order can happen either with a split strategy – in which an order with multiple items from the same customer is broken down and each driver is tasked with delivering a piece of the pie – or with a relay strategy sometimes called transshipment – in which the courier that picked up the parcel transports it only for part of the journey, handing it to another courier tasked with delivering it to the next stop or to the customer door-step. Zhou and Lin (2019) considers fixed relay stations and a two-level model. The upper level is tasked with finding the candidate paths for inter-zonal delivery. With less than two relays between source and destination all paths are enumerated, but for trips with more than two relays to be computationally efficient, the formulation relies on finding the shortest paths. The lower-level model is fed the list of drivers, their respective sequence of visiting points and the relay points for each pair of neighborhoods and uses this information to determine the final intra-zonal assignment and routing. Similarly, Fikar et al. (2018) uses hubs to transfer payloads. Vans are tasked with bringing large quantities of goods from the sources to the hubs and cargo bikes, then with distributing the orders through the customer nodes. Li et al. (2022) proposes the use of transfer stations for meal delivery to expand the range of restaurants. These stations are placed based on the time-weighted distance between customer and restaurant nodes to convert long-range tours into multiple smaller trips that are assigned to regional couriers that can pre-position themselves on the transfer stations for minimal delay. First, regional orders are generated. The long-distance orders are added to build new test instances. Then, long-distance orders are divided and the routing is re-optimized for each region, including normal and split paths.

Steever et al. (2019) address the MDRP and study policies to handle orders that include food from multiple restaurants. Two strategies are considered – split, meaning that more than one courier is involved in pickup and delivery, and – non-split, where one courier must visit all the locations and deliver the consolidated order. The non-split option is modeled as a constraint that can be relaxed for the split scenario. The article concluded that a split strategy is effective in ensuring freshness but also increases the operational cost when compared to the non-split policy. Wang et al. (2022) also consider splitting but not as a policy, instead it is the customer that chooses to have the order divided, which is modeled by a random variable. When the customer chooses non-split or all orders are from the same place, the system handles it as one order, while if the customer chooses to split it, these are viewed as multiple orders. Contrary to the previous article, the study concluded that the non-split strategy carries higher costs.

Information Characteristics

According to Pillac et al. (2013), delivery problems can be boxed into one of four categories depending on the evolution, *i.e.* if the input is known beforehand or changes with time, and quality of information, *i.e.*, deterministic or stochastic input. Static and deterministic problems mean that the input is known *a priori*, and this does not change during the route's execution. While in static and stochastic problems, the route is designed in advance with the known input. Stochastic variables are only revealed with the execution of the route thus allowing for minor changes. Dynamic and deterministic problems assume that part or all information is unavailable, being only revealed through the execution of the route. Likewise, dynamic and stochastic problems have the input revealed with the route's execution but use probabilistic information to make decisions. Of the selected articles, none are static and deterministic, which is expected given the uncertainty related to aspects intrinsic to on-demand services. Most are dynamic and deterministic or dynamic and stochastic, even though there are some static and stochastic problems. Since dynamism and stochasticity add complexity to a model, it is important to focus on those elements.

The major source of dynamism in ID and SDD problems are the requests that are revealed over time. In some cases, requests include additional information that other authors model independently, but for being incorporated in the request can be considered dynamic in nature. This is the case of preparation time (Liu 2019) or the splitting option (Wang et al. 2022). Besides orders, Fikar et al. (2018), Tu et al. (2020) and Arslan et al. (2019) considered that couriers arrive and leave the system in a dynamic manner to simulate the real-world scenario of crowdsourced workers, for ID and SDD respectively. Apart from orders and couriers, Tu et al. (2020) also included dynamic travel times to mimic the effect of traffic.

Stochasticity is less present when compared to dynamism. Yet, about half of the authors consider it, and it is usually associated with requests. This means not only that requests arrive with different periodicities, but also that the size of the order and the pickup and delivery locations can be random. Liu (2019) builds a model with stochastic orders that includes the regular aspects of an order plus the type of meal and a random preparation time. Ulmer et al. (2021) and Ulmer and Streng (2019) both include additional uncertainty associated with the ready or preparation times.

The main features found in the literature of meal and SDD problems are summarized in Table 2. The table includes features relevant to the context of ID assignment and routing problems, while also indicating the percentage of selected articles that incorporate such features.

Table 2. *Physical, scenario and information features of ID and SDD problems.*

	Reference	**Physical**						**Scenario**					**Information**	
		HO	**HV**	**TW**	**VC**	**WL**	**T**	**RD**	**CR**	**AR**	**PP**	**OP**	**DE**	**SE**
	Percentage of papers %	*14%*	*33%*	*38%*	*43%*	*38%*	*10%*	*24%*	*14%*	*19%*	*62%*	*24%*	*86%*	*52%*
ID	Auad et al. (2022)				✓	✓			✓				✓	
	Bozanta et al. (2022)			✓						✓			✓	
	J. fang Chen et al. (2022)										✓		✓	
	Fikar et al. (2018)	✓	✓		✓						✓	✓	✓	
	Li et al. (2022)			✓							✓	✓	✓	
	Liao et al. (2020)	✓	✓	✓		✓					✓			✓
	Liu (2019)	✓	✓			✓							✓	✓
	Reyes et al. (2018)			✓	✓								✓	
	Snoeck and Winkenbach (2022)				✓								✓	✓
	Steever et al. (2019)		✓	✓	✓		✓	✓			✓	✓	✓	✓
	Tu et al. (2020)		✓	✓	✓		✓		✓	✓			✓	
	Ulmer et al. (2021)							✓	✓		✓		✓	✓
	Wang et al. (2022)					✓		✓				✓	✓	✓
	Yildiz and Savelsbergh (2019)				✓	✓					✓		✓	
SDD	Arslan et al. (2019)				✓	✓					✓		✓	
	X. Chen et al. (2022)		✓			✓		✓					✓	✓
	Ulmer and Streng (2019)										✓		✓	✓
	Ulmer and Thomas (2018)		✓	✓	✓					✓	✓		✓	✓
	Voccia et al. (2019)			✓				✓		✓	✓		✓	✓
	Zhou and Lin (2019)					✓						✓		✓
	This work		✓		✓	✓	✓	✓				✓	✓	

Heterogeneous orders (HO); Heterogeneous vehicles (HV); Customer time windows (TW); Variable couriers (VC); Working limits (WL); Traffic (T); Route diversion (RD); Courier relocation (CR); Ability to reject (AR); Pooling policies (PP); Other policy (OP): relaying or slitting; Dynamic elements (DE); Stochastic elements (SE).

3.2.2 Objective Function

When it comes to the objective function, all authors, with the exception of Liao et al. (2020), consider single objective functions. In the abovementioned case, three objectives – average customer satisfaction, total carbon footprint and the scheduling equalization utilization rate of couriers – are optimized. Some of the most popular objectives of traditional deliveries are shared with on-demand problems and include cost, travel and waiting time minimization and maximization of quality of service or the number of served customers (Ojeda Rios et al. 2021), while lateness or earliness are very unique to meal delivery problems.

J. fang Chen et al. (2022) and Tu et al. (2020) calculate cost based on travelled distance and attribute a penalty cost for tardiness that is influenced by the waiting time at the restaurant. Snoeck and Winkenbach (2022) contemplate not only operational, but also the strategic decision of locating a facility. The objective is to minimize the whole network cost including setup, drivers, distance and lost sales. Other studies focus solely on courier costs as a function of travel and waiting time (Arslan et al. 2019, Yildiz and Savelsbergh 2019, Zhou and Lin 2019). Lateness or earliness are rarely used as a sole objective and are instead combined with other metrics. Liu (2019) defines minimizing lateness as the main objective, but also includes travel time minimization to evade trivial battery swaps and to avoid needless drone wandering. Reyes et al. (2018) maximizes the thruput, dividing the number of orders in a bundle by the total delivery time, as well as penalizing lateness. Similarly, Fikar et al. (2018) and Li et al. (2022) also penalize lateness but, instead of maximizing thruput, travel time is minimized. On the other hand, Wang et al. (2022) maximizes earliness while penalizing lateness. Steever et al. (2019), apart from considering that same function, also proposes two other alternatives which include minimizing the time since an order is ready to pick up until it is delivered or minimizing the total travel time.

Auad et al. (2022) have the objective of minimizing the freshness loss by minimizing the time between an order becoming available for delivery and the time it is picked up by a courier. Bozanta et al. (2022) consider an objective function that maximizes the revenue obtained from served requests. By contrast, instead of focusing on the orders served, the objective function formulated by Voccia et al. (2019) relies on minimizing the number of requests that violate time constraints and total route duration. Other objectives less frequently found in literature include minimizing waiting time (Ulmer et al. 2021), minimizing the total delivery times (Ulmer and Streng 2019) and maximizing the number of customers served (X. Chen et al. 2022, Ulmer and Thomas 2018).

Table 3 summarizes the insights of this subsection. The first two columns present the single objective or multiple objective functions, while the other columns emphasize the most common objectives targeted in each article, as well as their prevalence in the selected articles in percentage terms.

Table 3. *Objective functions of ID and SDD problems.*

	Reference	Objective		Minimize						Maximize			
		SO	**MO**	**C**	**L**	**TT**	**WT**	**DT**	**CF**	**CS**	**E**	**CU**	**O**
	Percentage of papers %	*95%*	*5%*	*29%*	*38%*	*10%*	*14%*	*10%*	*5%*	*24%*	*14%*	*5%*	*14%*
ID	Auad et al. (2022)	✓					✓			✓			
	Bozanta et al. (2022)	✓								✓			
	J. fang Chen et al. (2022)	✓		✓									
	Fikar et al. (2018)	✓				✓				✓			
	Li et al. (2022)	✓			✓								
	Liao et al. (2020)		✓		✓				✓		✓	✓	
	Liu (2019)	✓			✓								✓
	Reyes et al. (2018)	✓			✓								✓
	Snoeck and Winkenbach (2022)	✓		✓									
	Steever et al. (2019)	✓			✓	✓					✓		
	Tu et al. (2020)	✓		✓	✓								
	Ulmer et al. (2021)	✓					✓						
	Wang et al. (2022)	✓			✓						✓		
	Yildiz and Savelsbergh (2019)	✓		✓	✓		✓	✓					
SDD	Arslan et al. (2019)	✓		✓									
	X. Chen et al. (2022)	✓								✓			
	Ulmer and Streng (2019)	✓						✓					
	Ulmer and Thomas (2018))	✓								✓			
	Voccia et al. (2019)	✓											✓
	Zhou and Lin (2019)	✓		✓									
	This work	✓		✓									

Single objective (SO); Multiple objective (MO); Cost (C); Lateness (L); Travel time (TT); Waiting time (WT); Delivery time (DT); Carbon footprint (CF); Customers served (CS); Earliness (E); Courier utilization (CU); Other (O).

3.2.3 Solution Methods

The characteristics of the APs of ID and SDD make it unfeasible to apply exact methods to find optimal solutions for large problem instances. Consequently, it is more common for approximate or hybrid approaches to be applied and, in most cases, exact methods are used to produce a benchmark to which to compare the approximate solutions. Yildiz and Savelsbergh (2019) propose an exact solution approach for the MDRP based on a Branch-and-Price (BP) algorithm. For this method to be feasible, the pooling of orders is limited to two, a feasibility constraint to serve all requests is enforced and the model is static and deterministic. Reyes et al. (2018) picks up on this exact solution approach and compares it to a rolling-horizon approach for repeated matching using an Adaptive Large Neighborhood Search (ALNS) heuristic. Zhou and Lin (2019) applies a Branch-and-Bound (BB) algorithm in order to have a reference to measure the solutions obtained by an Adaptive Boundary Relaxation (ABR) heuristic. Arslan et al. (2019) considers a rolling horizon framework that solves the matching problem with the information available at a specific time. The formulation of the sub-problems considers that each driver is assigned to, at most, one request, making this formulation closer to the classic AP, which can be solved in polynomial time by the recursive exact algorithm proposed.

Steever et al. (2019) present an auction-based heuristic to cope with dynamic orders and the necessity of re-solving the problem with every new entry. Whenever a request arrives, a sub-problem for each active courier, considering the already attributed orders plus the new one, is solved and the objective evaluated. Then, in the myopic approach, the courier with the maximum bid is assigned to the customer. A proactive variant is also proposed that not only considers the bid value, but also future looking metrics of equity – distance between the nearest courier and all restaurants – and dispersion – scattering of couriers over the grid. The same authors also use an exact approach to contrast the performance of the heuristic. The exact method does not consider the future looking measures and even for small sets takes a prohibitive time to complete. Wang et al. (2022) build upon the previous work and propose a simplified auction-based heuristic that balances only one metric of responsiveness to future orders. Other heuristics are explored such as ALNS (Li et al. 2022) or progressive dispatch algorithms (Liu 2019).

Hybrid approaches combine two or more algorithms being able to exploit the advantages of different methods. Auad et al. (2022) use a two-phase heuristic that dynamically reconfigures the sets of regions first and afterwards applies the matching algorithm to a bipartite graph formulation. Fikar et al. (2018) rely on Agent-Based Modeling (ABM) and simulation that periodically invokes a two-stage procedure that runs an insertion heuristic to generate the assignment, routing and scheduling, followed by the optimization of the routes with Local Search (LS). Tu et al. (2020) based a hybrid metaheuristic on ALNS, to ensure diversity in the solutions, and balanced it with Tabu Search (TS) to intensify and improve assignments and routes. Apart from the mentioned exact method, Yildiz and Savelsbergh (2019) developed a Column Generation (CG) heuristic. The hybrid method is based on an algorithm of simultaneous column and row generation and the initial set of columns is given by a Greedy Heuristic (GH).

A considerable number of studies have applied Machine Learning (ML) methods to APs. ML is a branch of Artificial Intelligence (AI) that enables intelligent agents to improve a goal based on the experience gathered by interacting with the environment without being explicitly programmed. These methods can be divided into supervised, unsupervised and reinforcement learning (RL). Supervised means that the input data is labeled with the desired output so that the model can learn and predict outcomes. On the contrary, unsupervised means that the inputs are not labeled and the model explores the underlying structure of the data by identifying patterns. In RL the interaction between agents and environment takes the form of sequences of actions, observations and rewards that train and shape how the model acts (Soni and Kumar 2022). RL methods are by far the most common ML approaches present in the selected articles. Liao et al. (2020) use an unsupervised learning technique – k-means – paired with Principal Component Analysis (PCA) to cluster similar orders in the same trip. In a subsequent phase, the initial assignment and routing obtained via a Nondominated Sorting Genetic Algorithm II (NSGA-II) is optimized using ALNS. J. fang Chen et al. (2022) use a best matching heuristic to assign new requests to drivers. Whenever there is a tie, a tie-breaking strategy is employed based on ML. The authors compare two supervised techniques, Problem Transformation Method (PTM) and Algorithm Adaption Method (AAM). The RL methods found in the relevant literature include Q-Learning (Bozanta et al. 2022), Deep Q-Networks (Bozanta et al. 2022, X. Chen et al. 2022) and Markov Decision Process (MDP) (Ulmer et al. 2021, Ulmer and Streng 2019, Ulmer and Thomas 2018, Voccia et al. 2019)

Table 4 summarizes the insights of this subsection. The first set of columns comprises the type of methods used, followed by the algorithms. The 'Online' column registers if the solution methods apply calculations on-the-fly as new information becomes known. The last column indicates if the article tested the algorithm using data from a real-world scenario.

Table 4. *Solution methods and other related characteristics for ID and SDD problems.*

	Reference	**Approach**			**ML**	**Online**	**RW**
		E	**A**	**H**			
	Percentage of papers %	*24%*	*59%*	*33%*	*43%*	*62%*	*43%*
ID	Auad et al. (2022)			✓			✓
	Bozanta et al. (2022)		✓		✓	✓	
	J. fang Chen et al. (2022)		✓		✓	✓	✓
	Fikar et al. (2018)			✓		✓	
	Li et al. (2022)		✓			✓	✓
	Liao et al. (2020)			✓	✓	✓	
	Liu (2019)		✓			✓	
	Reyes et al. (2018)	✓		✓		✓	✓
	Snoeck and Winkenbach (2022)		✓		✓		✓
	Steever et al. (2019)	✓	✓	✓		✓	
	Tu et al. (2020)			✓		✓	✓
	Ulmer et al. (2021)		✓		✓		✓
	Wang et al. (2022)		✓			✓	
	Yildiz and Savelsbergh (2019)	✓		✓		✓	✓
SDD	Arslan et al. (2019)	✓					
	X. Chen et al. (2022)		✓		✓	✓	
	Ulmer and Streng (2019)		✓		✓		
	Ulmer and Thomas (2018)		✓		✓		✓
	Voccia et al. (2019)			✓	✓	✓	
	Zhou and Lin (2019)	✓	✓				
	This work	✓				✓	✓

Exact approach (E); Approximate approach (A); Hybrid approach (H); Machine Learning (ML); Real-world data (RW).

3.3 Classical Assignment Problems

From an operational level, IDs require the assignment of orders and the routing of couriers. Even though formulations vary on a case-by-case basis, delivery problems can be sorted depending on whether these two aspects are considered in a holistic manner or not. A two-stage approach means that every sub-problem is addressed individually, with the first phase's output being the second's input. If the assignment is made beforehand, the sequencing of orders or routing is reduced to solving multiple instances of a Traveling Salesman Problem (TSP) (Mor and Speranza 2022). Conversely, in the one-stage method, the two sub-problems are coalesced into a single formulation (Montoya-Torres et al. 2015). While one-stage problems can be seen as a variation of the classical AP, the two-stage formulation is usually referred to as the Vehicle Routing Problem (VRP).

Introduced by Kuhn (1955), the AP is one of the oldest combinatorial optimization problems that continues to be studied until this day (Schrijver 2005). In its most simple form, it can be summarized as matching n

tasks and n agents, one-to-one, as to minimize the total cost of the assignment. Given the index sets of agents $I = \{1, \dots, n\}$ and tasks $J = \{1, \dots, n\}$, the cost of assigning task j to agent i (c_{ij}) and the binary variable that specify if task j is assigned to agent i (x_{ij}), the mathematical model for the AP is defined as:

$$\min \quad \sum_{i \in I} \sum_{j \in J} c_{ij}\, x_{ij} \tag{1.1}$$

$$s.t. \quad \sum_{i \in I} x_{ij} = 1 \qquad j \in J \tag{1.2}$$

$$\sum_{j \in J} x_{ij} = 1 \qquad i \in I \tag{1.3}$$

$$x_{ij} \in \{0,1\} \qquad i \in I, j \in J \tag{1.4}$$

Despite the objective function (1.1) being a minimization of the assignment cost, the problem can also be formulated as a maximization of profit. In the case of a continuous problem, constraints 1.4 can be substituted with bounding constraints $0 \leq x_{ij} \leq 1$, $i \in I, j \in J$ (Maniezzo et al. 2021). Constraints 1.2 and 1.3 are known as assignment or semi-assignment constraints and ensure that each task is assigned to one agent only and that each agent is assigned to one task only respectively. Lastly, constraints 1.4 ensure that the decision variables respect the integrality condition.

Over the past fifty years, numerous variations have been proposed which are captured and reliably cataloged in the surveys of Öncan (2007) and Pentico (2007). The Generalized AP (GAP) is one of the most studied variants and particular relevant to the context of IDs. First proposed by Ross and Soland (1975), the GAP not only considers an unequal number of tasks and agents, but also allows for the assignment of multiple tasks to the same agent while considering the capacity limitations. The mathematical model of the GAP is identical to the classic AP, with the exceptions of the index set of agents $I = \{1, \dots, m\}$, the amount requested by task j to agent i (q_{ij}), the capacity of agent i (Q_i) and constraints 1.3 that is replaced by 2.3 to ensure that the available capacity of each agent is not exceeded, resulting in the following formulation:

$$\min \quad \sum_{i \in I} \sum_{j \in J} c_{ij}\, x_{ij} \tag{2.1}$$

$$s.t. \quad \sum_{i \in I} x_{ij} = 1 \qquad j \in J \tag{2.2}$$

$$\sum_{j \in J} q_{ij} x_{ij} \leq Q_i \qquad i \in I \tag{2.3}$$

$$x_{ij} \in \{0,1\} \qquad i \in I, j \in J \tag{2.4}$$

There are several algorithms to solve the simpler classic AP in polynomial time. However, the GAP is at least as hard as any nondeterministic polynomial time (NP) problem, hence being considered NP-hard

(Martello and Toth 1992). Therefore, for large instances, it is computationally intractable to use exact approaches favoring the use of approximate heuristic approaches.

The Hungarian or Kuhn-Munkres algorithm, proposed by Kuhn (1955) and refined by Munkres (1957), is a classic approach to solve the AP. The original method proposes a $n \times n$ qualification matrix, populated by the assignment costs (c_{ij}), where the columns represent tasks and the rows represent agents. The problem is said to be unbalanced if the number of tasks and agents is not equal, however, it can be solved by adding columns or rows with zeros to create fictitious tasks or agents that are not assigned but allow the algorithm to find a solution. The procedure is comprised of five steps: (1) Subtracting the smallest entry in each row from all other entries in that row, resulting in all rows having at least one zero; (2) Subtracting the smallest entry in each column from all other entries in that column, resulting in all columns having at least one zero; (3) Testing if an optimum assignment can be made by determining the minimum number of lines needed to cover all zeros is equal to the number of rows or columns. If this is the case, skip to step 5, otherwise go to step 4; (4) Subtracting the smallest entry not yet covered from all other uncovered entries in the matrix and adding it to each entry at an intersection of covering lines and repeating step 3; (5) Assigning the tasks to agents beginning with rows or columns with only one zero, followed by the other zero, ensuring that each row and column has one match only.

The Kuhn-Munkres algorithm cannot be applied to the GAP, as it is only applicable to square matrixes and cannot handle the added constraints 2.3, requiring other exact or approximate methods to solve the problem. An alternative to the Kuhn-Munkres algorithm is the Jonker-Volgenant (JV) algorithm that is computationally less complex and handles rectangular cost matrixes without the need to convert them into square matrixes (Jonker and Volgenant 1987). Common exact methods include BB (Cattrysse and Van Wassenhove 1992), BP (Savelsbergh 1997) and the relaxation of the semi-assignment (Martello and Toth 1992) or the capacity constraints (Ross and Soland 1975). Heuristics are used in GAP in one of two ways – either to generate solutions or to obtain an upper bound to be used in BB procedures (Nauss 2003). Common heuristics and meta-heuristics used alone or in hybrid formulations include Lagrangean relaxation and decomposition, Greedy heuristics, Adaptive Search Procedures, Simulating Annealing, Tabu Search, Genetic Algorithm, and Variable Depth Search (Morales and Romeijn 2005, Osman 1995).

3.4 Chapter Conclusions

Significant work has been conducted on the assignment and routing of ID operations, mostly applied to meal delivery. Some features have been studied extensively, namely multiple pickup points, pooling policies, the ability to pre-position, dynamic requests and, to a lesser degree, customer time-windows, diversions and crowdsourced couriers. Other traits intrinsic to platform-based delivery were studied in a limited manner, such as order and vehicle heterogeneity mostly accounting for size or weight and capacity respectively, while order priority and traffic limitations were virtually unexplored.

Almost all research relies on single objective functions with some measure of time or cost and frequently conjugating more than one metric. The models are solved largely with approximate or hybrid methods, and exact approaches are almost exclusively relegated for small scale problems or used as a standard to compare other algorithms. In the last two years, a clear move towards machine learning can be noticed, which seems promising given the necessity of fast online algorithms to use in real-world applications.

The Hungarian Algorithm (HA) is the preferred choice to solve classic APs, however, to study the effects of capacity and pooling, either approximate or hybrid methods must be pursued. Foundational works on the MDRP (Reyes et al. 2018), as well as interesting research of various types of heterogeneity (Steever et al. 2019, Tu et al. 2020, Ulmer and Thomas 2018), are likely to provide a solid backbone to an ID model.

4 Methodology

This chapter describes the model and solution approaches to solve the ID problem. Subchapter 4.1 characterizes the real-world problem. Subchapter 4.2 presents the dynamic approach that manages requests and couriers. Subchapter 4.3 details how the problem is modelled. Subchapter 4.4 lays out the solution methods used to solve the model. Subchapter 4.5 summarizes the main insights of this chapter.

4.1 Problem Description

The ID problem consists in finding the optimal assignment of orders to couriers, while respecting a set of constraints. Uncertainty is associated to both couriers that can log in and out of the platform and orders that are instantaneously put forward by customers and none of this information is known beforehand. Couriers start their day at one location and have an associated mode of transport, which for the case study is limited to bicycle, car and scooter, and cannot be changed. Each type of vehicle has a different speed. Naturally motorized vehicles are quicker, however can be slowed down by traffic that varies regionally and dynamically throughout the day. The platform automatically limits the type of vehicle that can be assigned to an order based on the size of the order, the availability of parking spaces, limitations to the circulation among other factors. Besides, each order has a pickup and delivery point that must be visited by the same courier. Assigned couriers must travel to a given location to pick the items. In some cases, it might be necessary to wait at the picking station for the order to be processed while in other cases, the courier must pick the items from an order and purchase them, *e.g.,* picking and buying all items in a grocery list. After delivering an order the courier can either stay at the delivery location waiting for a new assignment or proactively move to an area with higher demand. Figure 13 illustrates an example with four couriers (a car, two scooters and a bicycle) that are represented by their current locations and three orders, each represented by a pickup and a delivery node. The three squares above or below locations indicate the type of vehicle (for couriers) and the vehicles that can perform the order (for pickup locations). Order one can only be performed by bicycles or scooters, order two by car and order three by any vehicle. These restrictions limit the possible connections that are represented with dashed lines and make up the possible assignments. The solid lines represent the links between an order's pickup and delivery and are always connected if that order is selected. When each courier can only be assigned to at most one order at each time, the model can be simplified and viewed as an AP and in that case, if the order's pickup is visited, the client's or delivery location is necessarily next.

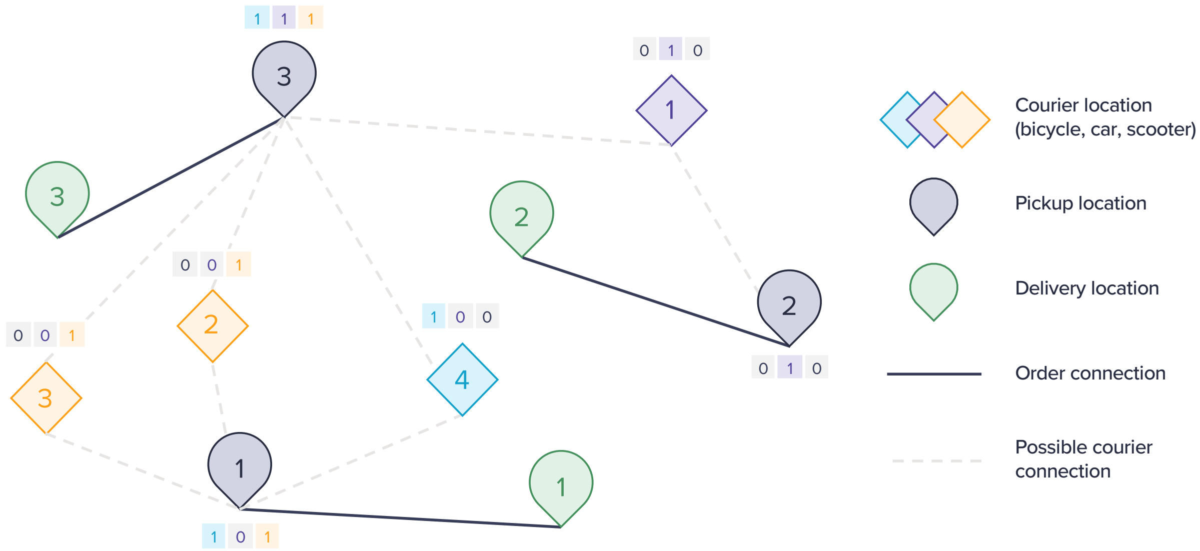

Figure 13. *Problem representation with four couriers and three orders.*

4.2 Dynamic Framework

Modelling IDs requires the incorporation of dynamic elements, namely the arrival of orders, changes in the fleet of couriers and the city traffic throughout the day so as to achieve a sincere representation of the real-world. To capture the dynamic aspects, Pillac et al. (2013) outlines various strategies, among them periodic reoptimization, which serves as the backbone for the presented framework.

Figure 14 illustrates a simplified flowchart where some of actions were condensed and represented as processes. The first step is to import to the program data containing information about couriers, orders and parameters as well as a timeline that initially lists the entries and exits of couriers and the arrival of orders. Courier data includes a unique identifier for each courier, the type of vehicle and the location at which the courier starts the shift. Likewise, order data include an identifier, information of the types of vehicle that can serve the request and the associated pickup and delivery locations, and also the submission time. The parameters include the speed of each type of vehicle by region, the traffic coefficients, the regional circuity factors (ratios of network and Euclidean distance that are incorporated to increase the precision of distance estimate) and the service times at the pickup and delivery locations as well as the time waiting for couriers to accept an order. Having imported the data, three additional empty databases are built to hold the available couriers at the current moment, the orders that have been submitted and have not yet been picked and the current assignments. Two other databases are created to store the final records of assignments and failed assignments to aid with the result collection. Afterwards, the model is initialized by setting the starting and end time and step.

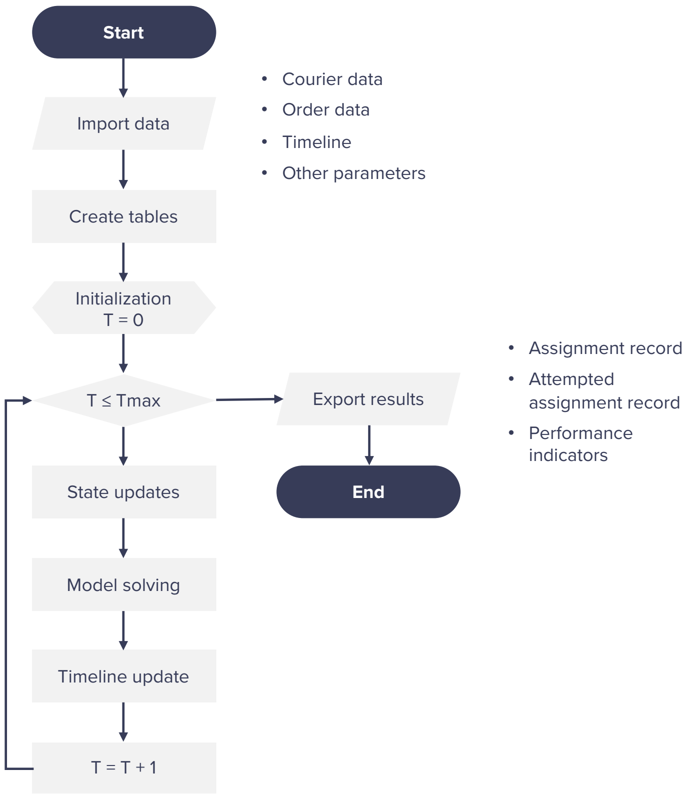

Figure 14. *Dynamic ID strategy or problem workflow.*

Represented as a single process in the flowchart, the state update phase condenses a series of updates that must take place before solving the model, including updating: the lists of active couriers and unserved requests, the position of couriers in-transit and the arrival times of couriers. For the first step, the timeline is filtered for current time, then a series of routines are triggered depending on the events that take place during that instant. Table 5 summarizes the types of possible events and the correspondent changes. While the arrivals of courier and orders are straightforward, the exiting of the system by couriers is dependent upon the courier not carrying a package. If that is the case, the system waits for the courier to finish by postponing the exit. Since it is assumed that couriers and orders can be reassigned up until the picking, both remain in the available databases. This strategy aims at preventing premature assignments by increasing the window of opportunity, allowing for new couriers or orders to arrive and provide a better overall outcome, while ensuring that the courier is on the way. The strategy has a downside, the increase in the running time of the model, however since the variation is small, it is a trade-off worthy of making. In Chapter 5 a thorough comparison between the assignment strategy and the faster baseline strategy is made. When the courier arrives at the picking location, both the order and the courier are removed from the available databases and the match is made permanent by removing it from the assignment database and placing it in the permanent record. After the order has been carried out, the courier re-enters the available courier database. Having completed all the updates, the timeline is cleaned of all events reported in the temporary assignment table (pickup start order, pickup start courier and delivery end) that were not made permanent before re-solving the problem, ensuring that only the best matches remain in the timeline. After updating the lists of available couriers and orders, the second step is to update the courier position. Despite the assignments only becoming permanent after picking, couriers start to drive towards the location. Since the courier are kept in the available database, if a new order drops closer to the courier or a new courier enter the system, this results in having to find a new optimum which requires knowing the current position of the courier. This is done in the second step by calculating the coordinates of an intermediate point in the line between the courier initial location and the pickup location that varies linearly as a function of time, *e.g.,* if a trip takes 10 minutes, at 5 minutes time, the courier will be exactly halfway between the two points. With the updated list of couriers and orders and locations, the distances between points and the consequent times are computed.

Table 5. *Updates based on the type of timeline event.*

Event	Changes
Courier enter	Add courier to available couriers list
Courier leave	If: courier delivering
	Postpone for the next period
	Else:
	Remove courier from available couriers list
Order arrives	Add order to unserved orders list
Picking start (order)	Remove order from unserved orders list
	Add assignment to record
	Remove assignment
Picking start (courier)	Remove courier from available couriers list
Delivery end	Add courier to available couriers list

After state updates the AP can be solved and the Mixed-Integer Linear Programming (MILP) formulations are detailed in the following subchapters with the AP being solved using two solution methods. Solving the iteration of the model results in the creation of a new temporary assignment database. For every pair of courier and order, three events are created and added to the timeline (pickup start order, pickup start courier and delivery end). Following this step, to the current time is added the step and the algorithm will have another iteration if the end time has not been reached. When the end of the running horizon is reached, a table with the final assignments for the running horizon is printed as well as a report with key performance indicators (KPIs).

4.3 Mathematical Model Formulation

An AP requires two sets to be matched, in this case the set of couriers $K = \{1, \dots, m\}$ and the set of orders $O = \{1, \dots, n\}$. The existence of constraints regarding vehicle types and travel times implies having additional sets of vehicle types V and of regions R that affect travel time. The notation used is summarized below and the model presented afterwards.

Sets

K	Set of available couriers	V	Set of vehicle types
O	Set of unserved orders	R	Set of regions

Parameters

l_k^K Current location of courier k

l_o^P Pickup location of order o

l_o^D Drop-off location of order o

ω_{vk}^K 1 if courier's k vehicle is of type v, and 0 otherwise

ω_{vo}^O 1 if order o can be assigned to a courier with a vehicle of type v, and 0 otherwise

T_o^S Submission time of order o

T_{ko}^A Assignment time of order o to courier k

T_{ko}^P Picking start time of order o by courier k

T_{ko}^D Delivery end time of order o by courier k

d_{ko}^P Distance between courier k and the pickup location of order o

d_o^D Distance between the pickup and drop-off locations of order o

c_r Circuity factor of region r

$s_{v,r}$ Speed of vehicle of type v for region r without traffic

μ_v Congestion factor of vehicle of type v

t_{ko}^P Travel time for courier k to get to the pickup point of order o from its current location

t_{ko}^D Travel time for courier k to get to the drop-off point of order o from the pickup point or order o

τ^W Waiting time for the acceptance of an order

τ^P Pickup time

τ^D Drop-off time

Variables

x_{ko} Binary variable equal to 1 if courier k is assigned to order o, and 0 otherwise.

Considering the notation, the AP is formulated as a minimization problem. Objective function 3.1, instead of cost, minimizes the sum of total delivery times, from moment of placement to delivery. Constraints 3.2 ensure that all requests are served. Constraints 3.3 restrict the maximum number of tasks a courier can perform at each time to one. Constraints 3.4 enforce the vehicle restrictions, ensuring that only couriers driving admissible vehicles to each order can be assigned to it. Constraints 3.5 ensure the domain of the decision variable is respected.

$$\min \quad \sum_{k\in K}\sum_{o\in O}(T^D_{ko} - T^S_o)\cdot x_{ko} \qquad (3.1)$$

$$s.t. \quad \sum_{k\in K} x_{ko} = 1 \qquad o \in O \qquad (3.2)$$

$$\sum_{o\in O} x_{ko} \leq 1 \qquad k \in K \qquad (3.3)$$

$$\sum_{v\in V} \omega^K_{vk}\omega^O_{vo} \geq x_{ko} \qquad k \in K, o \in O \qquad (3.4)$$

$$x_{ko} \in \{0,1\} \qquad k \in K, o \in O \qquad (3.5)$$

For clarity, the relationships between parameters are described below. Equation 3.6 gives the distance between two points in a sphere, where E is Earth's radius and all latitudes (φ) and longitudes (λ) are in radians. The formula is used to determine the distance between couriers and pickups $d^P_{ko} = d(l^K_k, l^P_o)$ and the distance between each order's pickup and delivery locations $d^D_o = d(l^P_o, l^D_o)$. Every location parameter corresponds to a pair of latitude and longitude $l = (\varphi, \lambda)$.

$$d = E \cdot 2 \cdot \arcsin\left(\sqrt{\sin^2\left(\frac{\varphi_2 - \varphi_1}{2}\right) + \cos(\varphi_1)\cos(\varphi_2)\sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right) \qquad (3.6)$$

Equation 3.6 returns the shortest distance between two points which in urban environments is seldom the case due to infrastructure or natural features constraining the direction of movement. This is handled by multiplying a regional circuity factor to the distance. Durations are given by equation 3.7 by dividing the real distance, that is equal to the linear distance multiplied by a circuity factor c_r, by the speed of the courier with a vehicle of type v in region r, while incorporating the slowing effect of traffic for the current time on motorized vehicles.

$$t_{ko} = \frac{d_{ko} \cdot c_r}{s_{v,r}}(1 + \mu_v) \qquad (3.7)$$

Equations 3.8 and 3.9, compute the time at which the courier starts the pickup and the time at which the order is delivered to the customer, respectively.

$$T^P_{ko} = T^A_{ko} + t^P_{ko} \qquad (3.8)$$

$$T^D_{ko} = T^P_{ko} + \tau^P + t^D_{ko} + \tau^D \qquad (3.9)$$

4.4 Solution Approach

The AP is solved by means of two methods – the JV algorithm and an exact approach based on Branch-and-Cut (BC).

4.4.1 Jonker-Volgenant Algorithm

The JV algorithm is a variation of the HA which is computationally less complex and handles rectangular cost matrixes without the need to convert them into square matrixes (Jonker and Volgenant 1987). This approach follows a two-phase structure with an initialization akin to that of the HA with a similar purpose of generate a set of tentative assignments fast. The second phase is the augmentation that incrementally builds a bijective map of assignments.

Initialization

The initialization phase comprises three procedures: (1) column reduction; (2) reduction transfer; and (3) augmenting row reduction. Figure 15 presents an example of a cost matrix and the steps comprised in this phase. The initialization starts by subtracting from each column its minimum element. The resulting matrix is scanned right-to-left assigning the column to the unique row correspondent to the zero element. In the example, the result of this step is matrix (b) where column three is assigned to row two, column one is assigned to row two and column two and row three remain unassigned. Similarly, in row reduction a positive value is subtracted. Since an assigned row necessarily has one zero, this would imply having a negative entry which is impossible. For this reason, an intermediate procedure of reduction transfer takes place, whereby a reverse column reduction is applied to the column correspondent to the assigned row. In the example, step (c) is the result of the anti-column reduction, meaning that to the first column was added, instead of subtracted, the minimum value of the correspondent assigned row. Afterwards the row reduction is applied resulting in matrix (d). Compared to matrix (b), it is evident that row one is reduced at the expense of column one. The last stage attempts to find alternating paths starting in an unassigned row and ending in an unassigned column. For an unassigned row i, the algorithm finds a column j_1 with a minimum entry e_1 and a column j_2 that contains the least entry e_2 such that $e_2 \geq e_1$. Then row i is reduced by e_2. If $e_2 > e_1$, the reduction would result in negative entries and, similarly to step two, this requires an inverse column reduction to column j_1. Row i is assigned to columns j_1. If the column was previously assigned to a row, that row becomes unassigned and step three is applied to it. The process goes on until that row is assigned to a column or it becomes impossible to transfer the reduction.

$$\begin{bmatrix} 1 & 7 & 6 \\ 5 & 2 & 3 \\ 8 & 9 & 4 \end{bmatrix} \quad \begin{bmatrix} 0 & 5 & 3 \\ 4 & 0 & 0 \\ 7 & 7 & 1 \end{bmatrix} \quad \begin{bmatrix} 3 & 5 & 3 \\ 7 & 0 & 0 \\ 10 & 7 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 2 & 0 \\ 7 & 0 & 0 \\ 10 & 7 & 1 \end{bmatrix}$$

$$(a) \qquad (b) \qquad (c) \qquad (d)$$

Figure 15. *Example: (a) cost matrix; (b) after column reduction; (c) after anti-column reduction; (d) after row reduction.*

Augmentation

The augmentation phase consists in finding the shortest path to an unassigned column. This is achieved by adapting Dijkstra's algorithm (Dijkstra 1959) to find the smallest cost instead of distance. Starting with an unassigned row, the shortest path to a column is added to the path. If the column was already assigned to a row, that row is also added to the path. The distances between the row and all other columns are updated. If this results in another row having a shorter distance to the column, them the connection is replaced. The procedure is repeated until an unassigned column is found. Finally, after augmentation, the solution is updated ensuring that the assignments correspond to the cost matrix row's minimum entry.

4.4.2 Branch-and-Cut Algorithm

The BC is an exact method that combines a BB algorithm with the cutting planes method to tighten the linear programming (LP) relaxations. The method starts with a preprocessing step intended to reduce the size of the problem by removing unnecessary constraints and fixing variables. The problem is presolved to avoid unnecessary steps in the case of the optimal solution being integer. If this is not the case, the integrality constraints are removed and the optimal solution to the LP problem is generated. To remove fractional solutions, Gomory's cuts are applied to tighten the solution space and unlike branching, do not create additional sub-problems. If the solution is not optimal, the problem branches and the steps after preprocessing are repeated for the new set of open nodes until the optimal integer solution is found. This algorithm is employed by Gurobi, a mathematical optimization solver, used to solve the AP.

4.5 Chapter Conclusions

The problem described was not addressed in the literature and as a result demanded a new formulation that addresses the vehicle restrictions that prohibits vehicles of delivering some orders. The combined effects of dynamic congestion and regional speed limits and road networks are not covered in previous works on the topic which required adding those steps to the formulation under the form of additional computations. A dynamic framework is presented capable of handling the arrival of requests and couriers and updating their positions thorough periodic reoptimization. The formulations and respective notations are presented and the respective solutions approaches presented.

5 Experiments and Results

This chapter presents the results of the experiments and respective analysis. Subchapter 5.1 explains the pre-processing applied to raw data to enable it to feed the model. Subchapter 5.2 analyses the baseline model, compares the results with the real assignment, and performs sensitivity analysis on key parameters. Subchapter 5.3 compares different policies with the baseline model. Subchapter 5.4 summarizes the main insights of this chapter.

5.1 Data Treatment

The model presented in the previous chapter is tested using a set of data obtained from the records of an on-demand delivery company. The data provided by this company was collected during one day of activity in the London metropolitan area in April 2018. Three databases were provided – the assignment database, that keeps record of the assignments and is used to compare the solutions provided with the one from the model; and the courier and jobs databases, that register continuous updates of location and state. These last two databases provide most the model's inputs but must be treated before feeding it.

Courier input is obtained by filtering the data by the driver's identifier, its vehicle and the initial location. Since the provided data concerns jobs, which might contain more than one customer, obtaining the order input requires decoupling the jobs with many orders. Afterwards the data is filtered by the identifier of the request, vehicle types that can serve it, as well as the pickup and delivery locations. The initial timeline is obtained by registering the first-time couriers and orders are recorded and the last-time couriers are recorded. These data include 203 couriers and 1062 requests. Figure 16 represents the number of couriers in the system by type of vehicle and Figure 17 shows the number of orders that arrive each hour that can be delivered by every vehicle or that are restricted to one or more types of vehicle. The curves mimic each other with more couriers than orders for the entire day except for 9 am, which also registered the highest value of pooled orders. Despite the fleet being relatively stable for most of the day, its composition varies with more cars working in the morning period and bicycles in the evening.

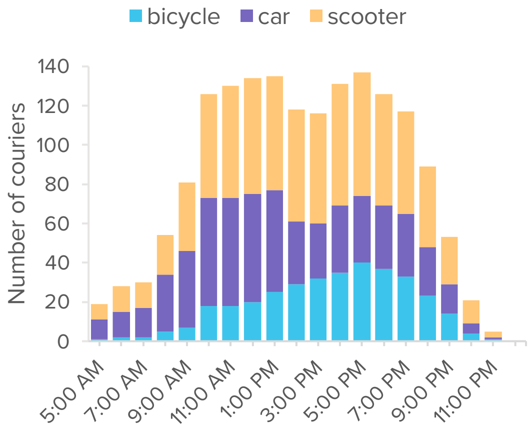

Figure 16. *Couriers in the system per hour.*

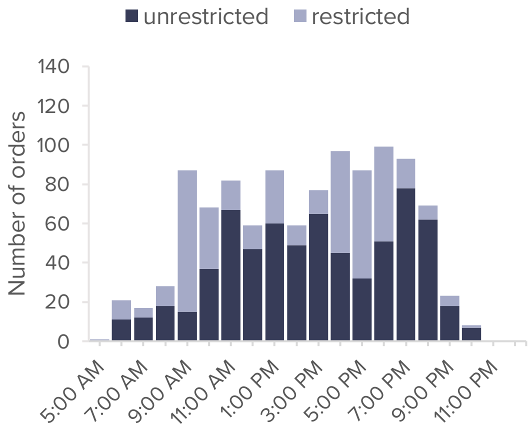

Figure 17. *Order arrivals per hour.*

Parameters can be manually tunned to fit different settings, however, to accurately represent the real scenario the service time parameters were derived from the courier data, by analyzing the duration of the changes of state. Figure 18 juxtaposes a timeline with the sequential stages that a courier delivering an order without pooling goes through and a map illustrative of the trip. The colors used in the timeline are analogous to the location in the map to indicate that the courier remains at the location during those stages and the variables used to describe the service time durations and the travelling times are the same adopted in the formulations presented in the previous chapter. The dots in the timeline indicate whether the courier already carries or not the order item(s). The stage “items purchased” is ignored because it is assumed that the picking ends only when the courier starts to travel to the delivery location.

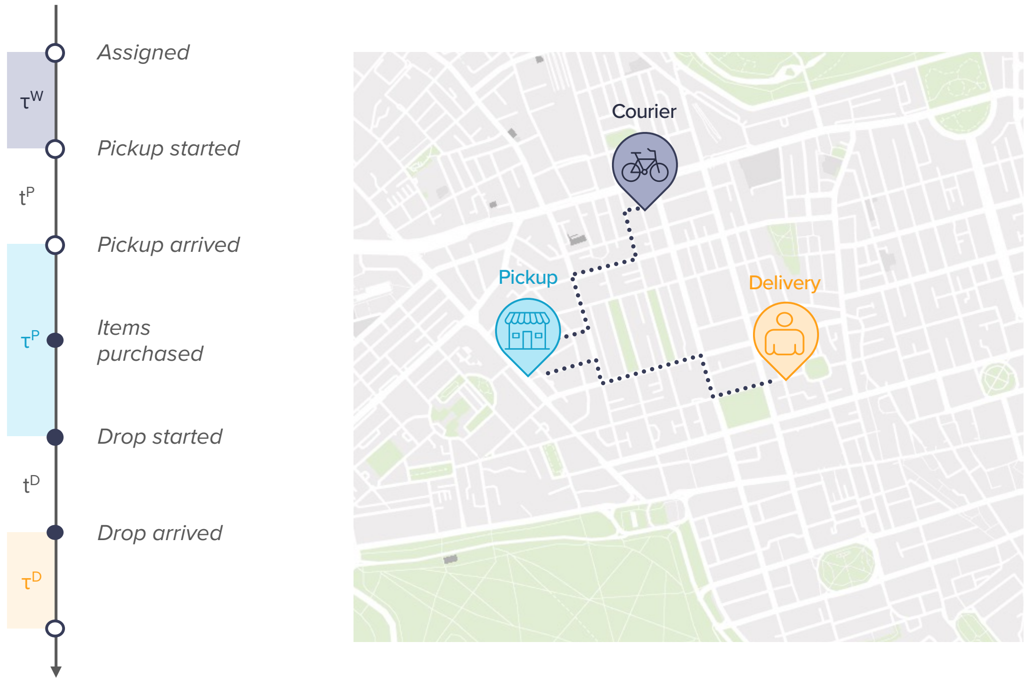

Figure 18. *Stages and derived parameters (left); Example of map with locations for each stage (right).*

Couriers start with no state and remain idle until an order is assigned. Upon being assigned, couriers must decide if they want to accept the request and if so the state changes to “pickup started”. Afterwards the courier leaves its current position and travels to the pickup point. When the destination is reached, the state changes to “pickup arrived”, and the courier remains at the location. The order can be ready by the time the courier arrives or require preparation by the store or even require that the courier picks and purchases different items. After having the order’s items, the state changes to “items purchased”, and them to “drop started”. The pickup service time comprises the interval between the states of “drop started” and “pickup arrived”. Next the courier moves towards the customer location changing the state to “drop arrived” when arrives. For a job with one order only this is the last state of the courier, and after its completion its state goes back to idle. The difference in time between the first drop arrived and the next idle state is defined as the delivery or drop service time. The waiting time and the pickup and delivery durations were computed for every order. Figure 19 shows a box plot without outliers for the three parameters. Waiting time has the smallest variation with most requests taking one minute or less to be accepted. By contrast the time for pickup has a greater variation which is expected given that this stage is often dependent on third-party restaurants or stores to be prepared. All service times follow right skewed distributions, however since the problem is deterministic, these values were set to the averages of 2, 12 and 6 minutes respectively for the waiting, pickup and delivery times.

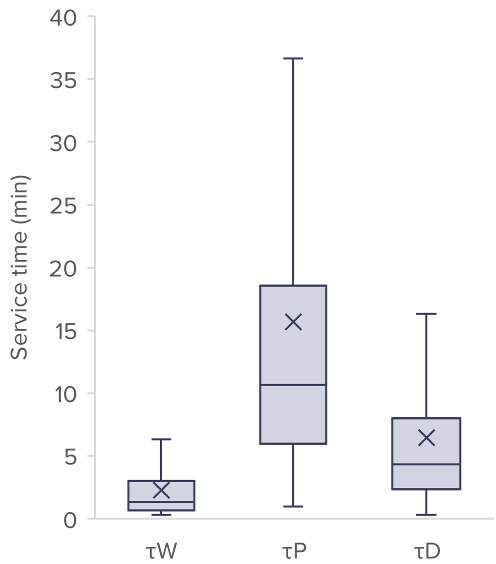

Figure 19. *Distribution of service time parameters, in minutes.*

Deducing the distance travelled by couriers from the data is not possible. Location updates are usually only recorded when there is a change in state and not for every period. Without additional software, and only the coordinates of start and end points, getting the real distance in a city network entails a degree of error due to the approximation. In literature, there are two functions usually employed to curb this problem – Euclidean and Manhattan. Euclidean distance is the length of a line segment connecting two points. Manhattan distance is the sum of vertical and horizontal segments joining two points in a grid. The Euclidean approach represents the shortest path between the two points, which makes it unrealistic to represent the distance in a city network. The Manhattan approach is more precise for a perfect grid; however, most cities do not resemble such layout. Furthermore, some roads only allow for traffic to flow in one direction or have lower speed limits, which makes the actual route follow neither path. Figure 20 shows the difference between the distance according to the Euclidean and Manhattan formulas and the real distance retrieved by Open Source Routing Machine (OSRM) engine (Project OSRM 2022).

For this reason, it is common to use a circuity or directness factor to correct a distance function that is nothing more than a ratio between the real distance and the Euclidean distance (Cardillo et al. 2006). Since the data comprises both inner and outer London, and road networks vary wildly by region, instead of a single circuity factor for the entire metropolitan area, a circuity factor is computed for every region. To determine these factors, the pickup, delivery and initial courier position nodes are grouped and filtered by region. For each borough, the distance between all nodes is computed using Euclidean and Manhattan formulas, as well as the distance of the fastest route according to OSRM engine. Figure 21 represents a box chart with the relative errors of the approximations and the real distance. When compared to the Manhattan, the Euclidean approach produces a greater error, however, when pondered with regional circuity factors, the estimates result in less error which indicate that of the three approximations, regional circuity estimates are closer to the actual distance values.

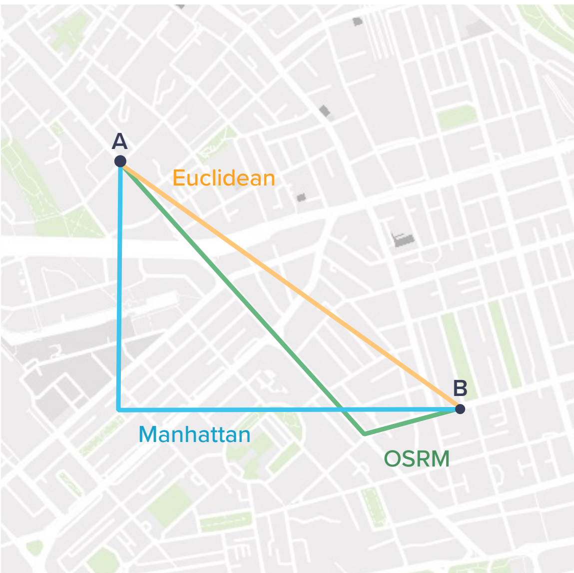

Figure 20. *Representation of Euclidean, Manhattan and real distance between two points.*

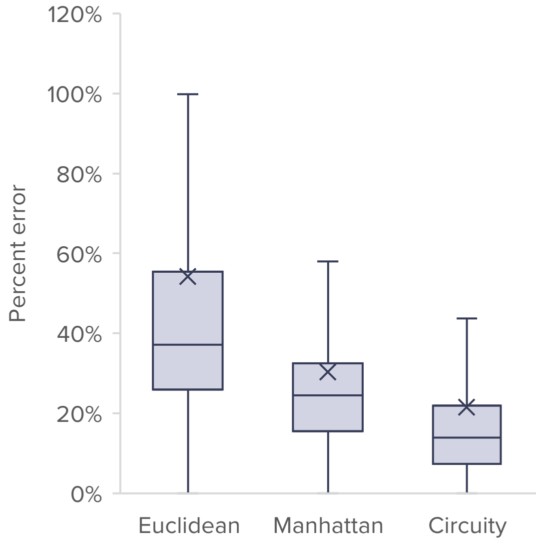

Figure 21. *Error produced by distance approximations.*

Apart from the distance between the nodes, OSRM engine provides the travel time without congestion for each route which allows to compute the average baseline speed for each borough. Figure 22 shows the average speed provided by OSRM in yellow and the average speeds for three times of the day according to the Department for Transport for the year 2018 (London Assembly 2020). The AM, inner and PM peaks, respectively comprise the periods of 7am to 10am, 10am to 4pm and 4pm to 19pm. The OSRM speeds are naturally higher since do not reflect the effect of congestion, but the curves follow a similar pattern, which suggest a high accuracy of OSRM engine. These regional speeds are used in the model for both cars and scooters. Because bicycles can use dedicated paths and due to a lack of precise bicycle speed data for every region, it is assumed that bicycles travel at 22,5 km/h independently of the region (Strava 2020). Both estimates for motorized vehicles and bicycles are tested in the following subchapters.

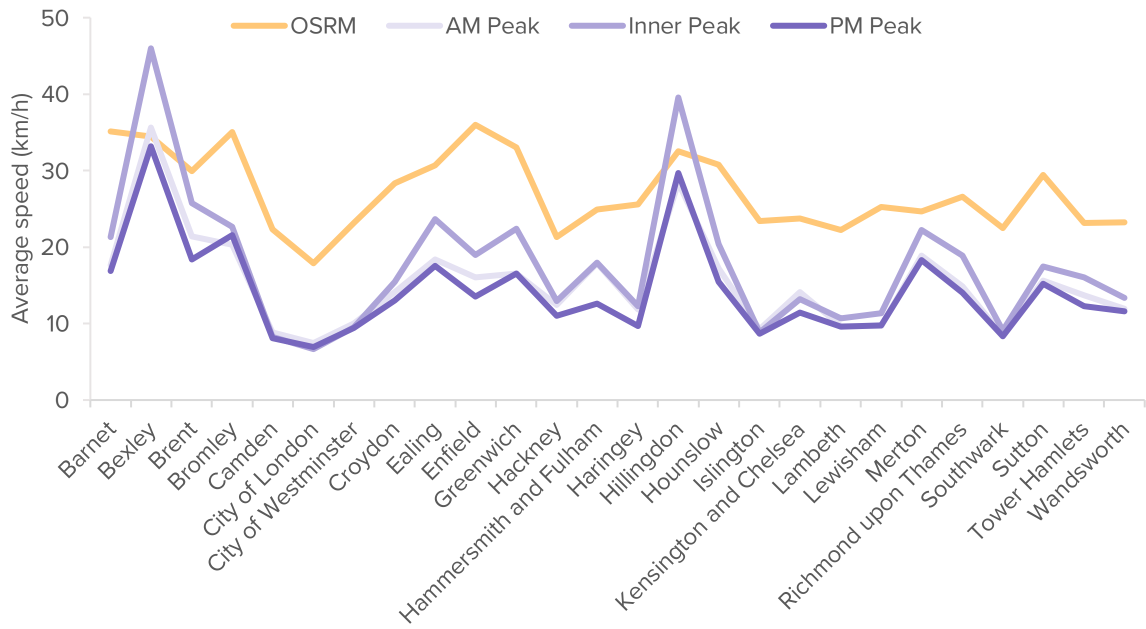

Figure 22. *Average car speed for borough with and without congestion, in kilometers per hour.*

The traffic input was derived from TomTom's traffic index (TomTom 2022), which provides an hourly congestion level. According to the index, a congestion level of 20% means that a trip takes 20% longer when compared to a non-congested period. Since the congestion index accounts for hourly data only, a piecewise linear function was used to describe the congestion level between two hours. Figure 23 depicts the congestion function for the day in question, starting with low congestion at 5am and peaking at 8am and 5pm which corresponds to the regular commuting to work and back home. The congestion input is equal to zero for bicycles and takes the value from the function if the vehicle is motorized.

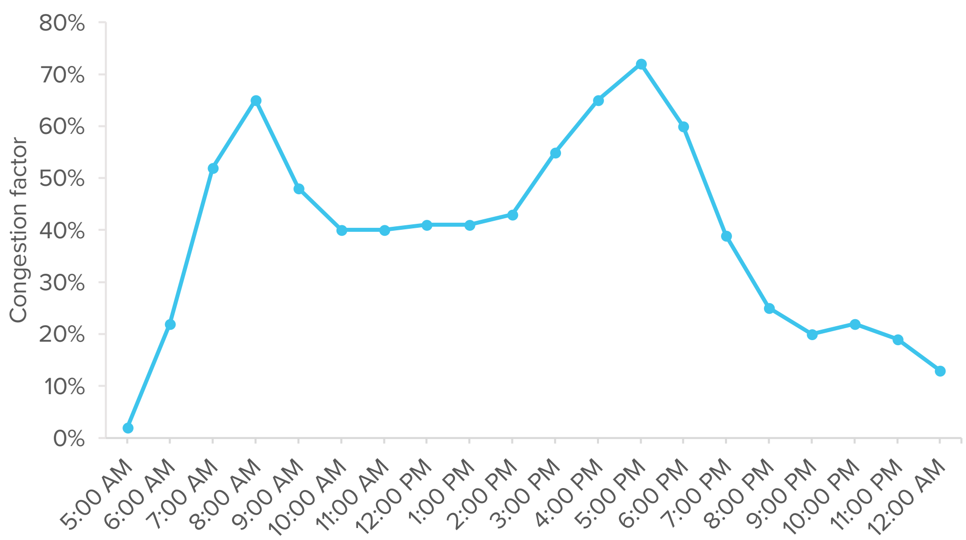

Figure 23*. Congestion factor during the day.*

5.2 Baseline Results

The assignment model runs within a dynamic framework for every instant. Depending on the updates that take place before solving the assignment, the inputs are modified, resulting in different outcomes. The simplest way of matching couriers and orders is to do so as soon as new orders arrive and not changing the assignment even if in a later instant a different match is better. This myopic approach is labeled as baseline to distinguish it from the alternative models that incorporate specific policies that are discussed in greater detail in Subchapter 5.3. The baseline framework is useful to serve as a benchmark against which other policies can be compared to. The dynamic baseline model is solved separately using the JV and BC algorithms and has as input the instance derived from the company's data and the parameters displayed in Table 6, Table 7, Table 8 and Table 9.

Table 6. *Service time and bicycle parameters.*

Parameter	Value
Waiting time (τ^W)	2 min
Pickup time (τ^P)	16 min
Delivery time (τ^D)	6 min
Bicycle speed (s_1)	22.5 km/h
Bicycle congestion factor (μ_1)	0

***Table 7**. Motorized vehicle speed by region in kilometers per hour.*

Region	0	1	2	3	4	5	6	7	8	9	10	11	12
Speed	35.1	34.5	29.9	35.0	22.3	17.9	23.2	28.3	30.7	36.0	33.0	21.3	24.9
Region	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**
Speed	25.6	32.5	30.8	23.4	23.7	22.3	25.2	24.7	26.6	22.5	29.4	23.1	23.3

Table 8. *Circuity factor by region.*

Region	0	1	2	3	4	5	6	7	8	9	10	11	12
Factor	1.50	1.93	1.39	1.48	1.50	1.72	1.53	1.71	1.6	1.53	1.64	1.55	1.45
Region	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**
Factor	1.41	2.03	1.46	1.53	1.43	1.74	1.97	1.72	1.77	1.5	1.78	1.48	1.65

***Table 9**. Congestion factor applied to motorized vehicles.*

Hour	5	6	7	8	9	10	11	12	13	14
Factor	0.01	0.16	0.37	0.47	0.37	0.34	0.37	0.41	0.41	0.41
Hour	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**
Factor	0.48	0.53	0.58	0.50	0.34	0.23	0.18	0.18	0.15	0.10

The assignment algorithm (JV or BC) is invoked every minute of a day, starting at 5am (minute 0) and ending at 12am (minute 1140), and solved to minimize the total delivery time at every instant. Table 10 resumes the sum and average objective value for both algorithms, as well as the average time it takes to process an entire workday. The JV algorithm is faster, which is expected since it is optimized for assignment purposes, while the BC is a general-purpose tool. There is a difference, albeit small, between the sum of objective values during the day of operations. This might seem paradoxical since both approaches are exact, however, the discrepancy is due to the way both algorithms assign orders to a fleet, which is heterogeneous.

Table 10. *Running time and objective values for the two algorithms.*

Algorithm	Running Time	Total Objective	Average Objective
JV	9.7 s	47 599 min	44 min 49 s
BC	17.3 s	47 844 min	45 min 3 s

The JV algorithm starts assigning from right to left side of the cost matrix, meaning that if two options have an equal value and only one can be selected, the JV approach selects the one closer to the right side of the matrix. By contrast, the BC algorithm selects the first optimal solution found, which might not coincide with the exact pair selected by the JV approach. This does not present any difference at the instant it happens but may have consequences in later instants' assignments. Supposing that an order that can be served by any vehicle arrives and there are only two couriers in the system, one with a bicycle and another with a car. If the assignment results in a tie where the JV algorithm assigns the car and the BC algorithm the bicycle, if a new order arrives afterwards that must be attributed to a bicycle, the JV algorithm can immediately assign a courier, while the BC algorithm must either wait for a new bicycle courier to arrive or wait for the first delivery to finish to assign the same courier. This effect starts a chain reaction that changes the couriers and orders available in subsequent instants, resulting in variations that, for the instance set used, average 14 seconds per order. Figure 24 illustrates the above-mentioned chain reaction by contrasting the number of orders both approaches assign to the same courier and the number of orders that are assigned to a different courier depending on the approach. In the morning, the percentage of different assignments is small, however throughout the day small changes accumulate and the disparities between the algorithms increase.

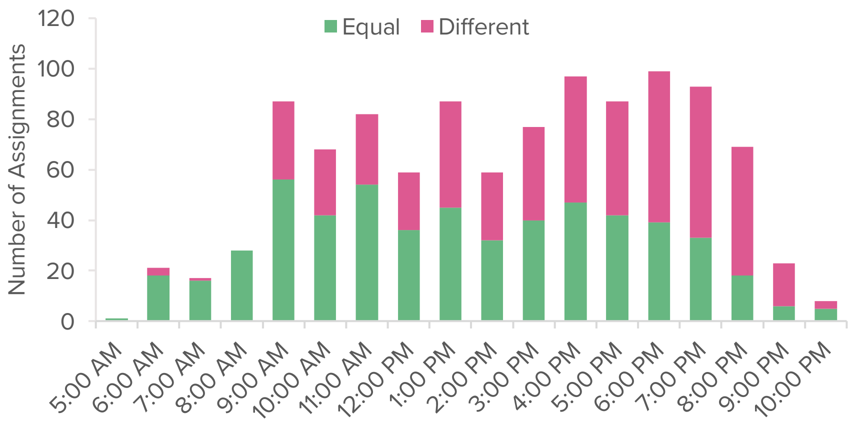

Figure 24*. Number of equal and different assignments between the JV and BC approach, for every hour.*

Despite the assignment differences, the two approaches lead to similar results not only in the sum of the objective values, but also in other KPIs. Table 11 compiles the results, grouped by type of vehicle, of some KPIs for the two algorithms. Both approaches score similarly in all metrics, with a negligible difference in the use of bicycles and scooters with the JV approach using more scooters and the BC more bicycles.

Table 11. KPIs for the baseline model, by bicycle (B), car (C) and scooter (S).

Algorithm	Requests Served			Delivery Time			Assignment Time			Courier Utilization		
	B	C	S	B	C	S	B	C	S	B	C	S
JV	31%	31%	38%	38	43	50	0	0	1	7.5	4.9	4.8
BC	32%	31%	37%	38	44	51	0	0	1	7.7	4.9	4.7

5.2.1 Comparison with Real Assignment

The data provided by the company included a list of courier and order pairs that indicated which courier ended up performing each order. The list does not necessarily correspond to the optimal assignment obtained by the company's HA because couriers can reject a job and the final assignment might not be the ideal one. However, for the purpose of comparing the developed approach with the real results, the assignments presented by the company in this list are used. For each courier and order pair, delivery time is computed using the same parameters of speed, service time, congestion and circuity factors used in the developed approach. The distance between locations is calculated using the same formula. Each courier has an initial location when entering the system and it is assumed that the first order of that courier starts from that point. If a courier is assigned to more than one order, the starting location is not its initial point, but rather the delivery location of the previous order. The sum of all delivery times using the real assignments results in 48 002 minutes in the same period, which is equivalent to an average delivery time of 45 minutes and 12 seconds. The objective for the real assignment is close to the values obtained with both JV and BC algorithms, although on average 23 and 9 seconds slower, respectively. This difference is negligible and can be attributed in part to the fact that these are not the optimal assignments, but instead the ones that were accepted, and to the fact that the real matches of couriers and orders did not incorporate congestion and the regional variability of roads. Focusing only on the overall objective does not tell the full story. To get a better grasp, the differences one must focus on are other KPIs, namely the orders delivered by each type of vehicle, courier utilization, and how the delivery time varies during the day and by region. Figure 25 shows percentagewise the requests served by each type of vehicle for the real assignment and the JV and BC assignments.

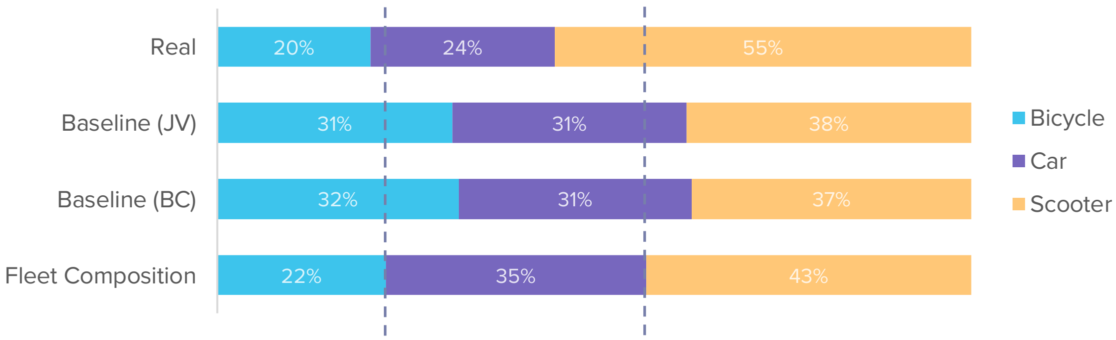

***Figure 25.** Requests served by vehicle type and fleet composition.*

The last bar in the chart represents the average fleet composition for the whole day. Both approaches that solve the dynamic model result in a similar composition, with around a third of the requests assigned for each type of vehicle. By contrast, the real assignment produced by the company attributes more than half of the matches to scooters, a fourth to cars and a fifth to bicycles. The results of the mathematical model, with either approach, might seem more even but, when juxtaposed to the average fleet composition, it becomes clear that both approaches result in more bicycles being matched at the expense of cars and especially scooters. The real model, on the other hand, favors scooters at the expense mostly of cars. Despite having more orders assigned to couriers with bicycles, the proportion of cars to scooters is the same as the fleet composition. This suggests that either congestion or regional speeds or both are the cause behind the ratios of modes of transport. Bicycles have a fixed speed and are immune to congestion, while cars and scooters have the same regional speeds and consider the congestion factor.

A challenge that ODD platforms with crowdsourced workers face concerns the difficulty in hiring and retaining couriers, in part due to the existence of competing firms. If the workload of a courier is low, the courier might leave the platform or use it in parallel with other platforms. On the other hand, if the workload is too high, the courier can be overworked and start to reject orders. For these reasons and from a long-term perspective, it is preferable to have a balanced use of the courier fleet, which is measured by the courier utilization indicator. Table 12 registers the values of the minimum, maximum and average values of orders per courier for each vehicle type, as well as the standard deviation for the developed approaches and for the real assignment. The results confirm the conclusions of the previous analysis with scooters in the case of the real model and bicycles for the developed approaches having higher utilizations. The standard deviation and maximum utilization are within the same range, however slightly higher for the dynamic approaches. Nevertheless, it should also be noted that the real assignment leaves seven couriers without any job, all driving cars, while the approach with the JV algorithm leaves three couriers (two cars and a scooter), and the approach with the BC algorithm leaves two couriers (a car and a scooter). The differences between the minimum and maximum number of orders assigned are high and could be improved by implementing strategies incentives to relocate.

Table 12. *Courier utilization by vehicle type.*

Model	Vehicle	Min	Mean	Max	Std. Dev
Real	Bicycle	1	4.9	11	2.5
	Car	0	4.2	13	2.5
	Scooter	1	6.9	19	3.8
Model (JV)	Bicycle	1	7.5	17	4.1
	Car	0	4.9	11	2.4
	Scooter	0	4.8	15	2.9
Model (BC)	Bicycle	1	7.7	18	4.4
	Car	0	4.9	13	2.6
	Scooter	0	4.7	16	2.8

The delivery time can be analyzed not only in total as the objective, but also grouped by vehicle type, period of the day and region. Figure 26 represents the average delivery time of each vehicle for the real assignment, as well as in the dynamic approaches solved with the two algorithms. Bicycles are the fastest mode of transport, which is partially due to being able to avoid congestion. However, if traffic was the sole reason at play, there should not be a discrepancy between the delivery time of motorized vehicles, since the same speed was considered for scooters and cars. Cars being on average faster than scooters can be attributed to the composition of the fleet during the day. Recalling Figure 16, the period when the active fleet has more cars is from 9am to 1pm, while the number of scooters is approximately the same from 10am to 7pm. This means that more cars are available in the system when congestion is lower, while scooters remain in a similar proportion throughout the day, being more penalized for the congestion factor. Another cause might be the starting location of couriers with cars that are placed in peripheric boroughs that have lower circuity factors and higher average speed limits which enables faster deliveries.

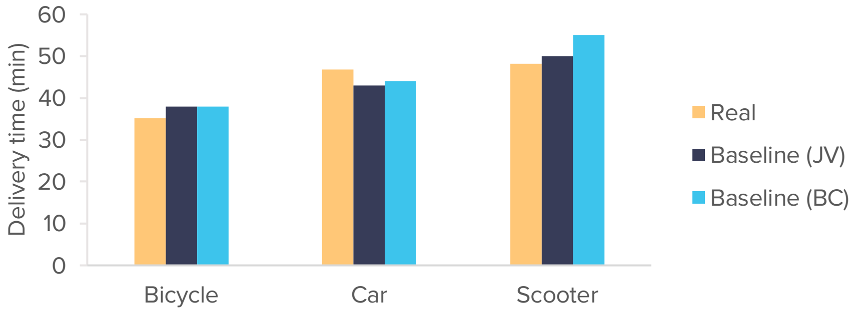

***Figure 26**. Delivery time by vehicle type, in minutes.*

Figure 27 looks at the average delivery time from a periodic perspective. The morning period goes from 5am to 9am, the day period from 9am to 2pm, evening from 2pm to 7pm and night from 7pm to midnight. The morning period represents the lowest delivery time because, during this period, congestion is lower and the ratio of couriers to orders is high. Traffic also explains the values for evening and night periods, but fails at justifying the spike, for both real and proposed assignments, during the day period. Instead, this increase is attributed to the ratio of couriers and orders. The period from 9am to 10am is the only time when more orders arrive than couriers are available in the system, furthermore, and recalling Figure 17, an overwhelming majority of orders have vehicle restrictions during this period. Of these, 49 orders must be served by scooters between 9am and 10pm when there are only 35 couriers with scooters in the system which also helps explain the higher delivery time of scooters represented in Figure 26. The real assignment enables pooling, which means that orders can be combined to be delivered by the same courier. This implies that when there is a lack of couriers with scooters, instead of waiting for more to arrive or finish the deliveries, the real model can attribute many orders to a single courier, which helps to explain the lower delivery time for the day period.

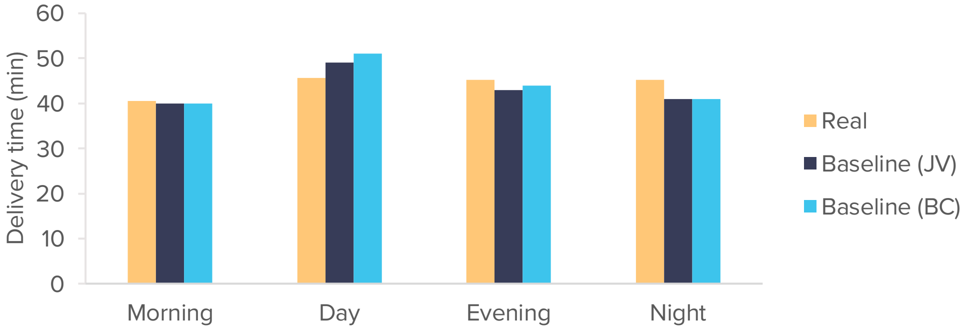

***Figure 27**. Delivery time by period, in minutes.*

Regionally, there are disparities in the average delivery time, which can range from 35 minutes to more than 80 minutes. For most regions, delivery time tends to be closer to the average and the more atypical values are registered in regions where there are fewer requests. Figure 28 represents the map of London, where the colors represent, for each region, the ratio between delivery time of the approach solved by the JV algorithm and the real assignment. Results obtained for the proposed approach using the BC algorithm are not displayed since they are analogous to the ones obtained for the proposed approach using the JV algorithm. If the ratio is higher, it means that the real assignment results in a shorter delivery time and is represented in red. A lower ratio means that the JV algorithm has the shortest delivery time and the correspondent borough is represented in blue. Light grey boroughs have no orders associated, while boroughs in white have an identical delivery time.

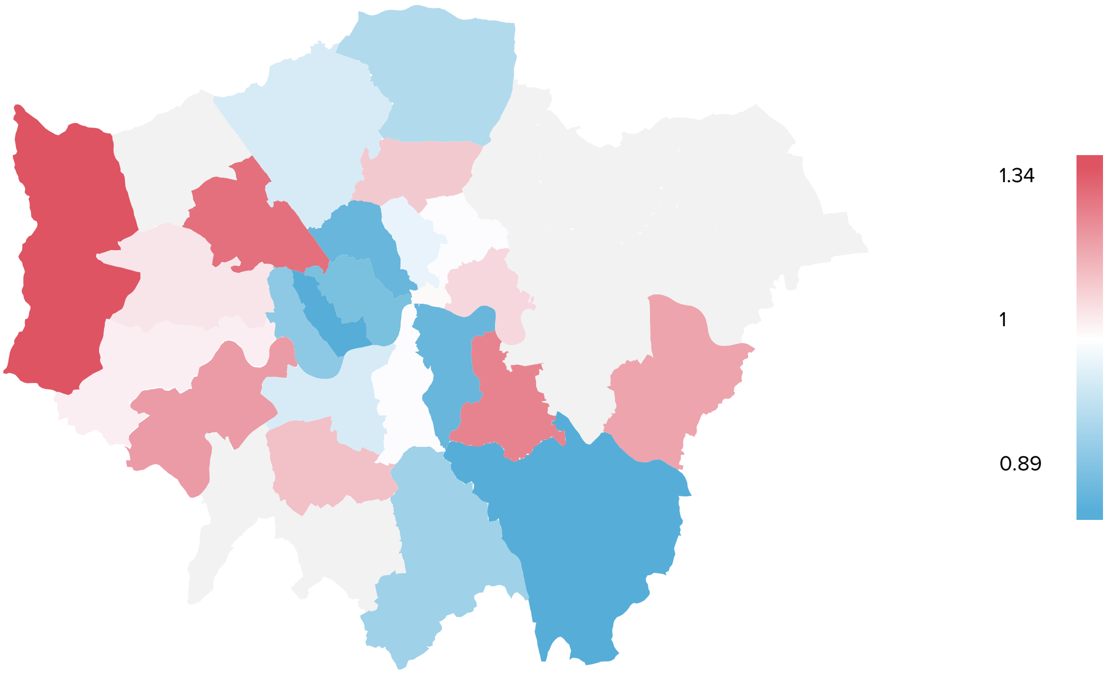

Figure 28. *Ratio of delivery time between the proposed approach (JV) and the real assignment.*

In the inner part of the city, the JV algorithm approach results in faster assignments, which is the region where most requests originate from. In the outer part of the city, results are mixed and, in the west, the real assignment produces faster deliveries. The cause for these results is not clear and might be attributed to variability that becomes more apparent in the outer part of the city, that contain only 20% of orders, while the inner part of the city corresponds to 80%. Other influences might play a role, such as the previously mentioned excess of orders for scooters that take place especially in peripheric regions or the starting location of couriers.

5.2.2 Sensitivity Analysis

Modelling a system implies making assumptions and approximations with regards to parameters. The sensitivity analysis aims at studying how the outputs of the model are affected by variations on the inputs. These experiments are conducted for the same set of data for the entire day.

Service Time

The service times of pickup and delivery are derived from the data provided by the company and the method used consists in spotting changes in state that indicate whether the pickup started or finished. Because these states are dependent on courier confirmation, it is not assured that those are the actual service times since the courier could have confirmed the update before or after it took place. For this reason, it is important to study the impact of the pickup and delivery time on the KPIs.

Table 13 registers the changes in the average delivery time, the percentage of requests served and the average courier utilization for each type of vehicle. Changing the pickup time has a proportional effect on the objective value, which is expected since the time at pickup is directly added in the delivery time calculation. As the pickup time decreases, more bicycle couriers are matched, while when it increases more motorized vehicles are chosen. When the average delivery is shorter, it means that more couriers finish their tasks faster and return to the list of agents waiting for orders. Also, the model tends to choose bicycles as they are not slowed down by congestion and, when the process is repeated with more bicycles, it becomes more likely for a bicycle to be assigned, which is confirmed by the higher utilization.

Table 13. *Average objective, requests served and courier utilization by bicycle (B), car (C) and scooter (S) for variations in pickup time.*

τ^P	Obj	Requests Served			Courier Utilization		
		B	C	S	B	C	S
4	32	35%	29%	35%	8.5	4.7	4.4
8	36	33%	30%	36%	8.0	4.8	4.7
12	40	32%	30%	38%	7.8	4.8	4.8
16	45	31%	31%	38%	7.5	4.9	4.8
20	50	31%	31%	39%	7.4	4.8	5.0
24	55	30%	30%	40%	7.3	4.8	5.0
28	60	28%	32%	40%	6.8	5.1	5.0

Time spent at the delivery location influences the objective value in a linear fashion, as evidenced by Table 14. Similarly to the pickup time, a reduction in this parameter results in shorter tasks and, as a consequence, more couriers with bicycles being utilized.

Table 14. *Average objective, requests served and courier utilization by bicycle (B), car (C) and scooter (S) for variations in drop time.*

T^D	Obj	Requests Served			Courier Utilization		
		B	C	S	B	C	S
2	40	32%	30%	37%	7.8	4.8	4.8
4	43	32%	30%	38%	7.8	4.8	4.8
6	45	31%	31%	38%	7.5	4.9	4.8
8	47	32%	31%	38%	7.7	4.9	4.7
10	50	31%	31%	39%	7.4	4.8	5.0

Speed

Parameters used for the speed of couriers are derived from distinct sources. In the case of bicycles, it is assumed that those couriers have a fixed speed that is not conditioned by the region or by congestion. Table 15 shows the results when only the speed of bicycles is changed.

Table 15. *Average objective, requests served and courier utilization by bicycle (B), car (C) and scooter (S) for variations in the speed of bicycles.*

s_1	Obj	Requests Served			Courier Utilization		
		B	C	S	B	C	S
9	49	10%	38%	52%	2.5	6.1	6.5
13.5	47	15%	36%	49%	4.1	5.7	6.1
18	46	25%	33%	42%	6.1	5.3	5.2
22.5	45	31%	31%	38%	7.5	4.9	4.8
27	44	36%	31%	33%	8.8	4.9	4.3

Figure 29 plots the average delivery time in minutes on the left axis and the average number of trips per courier on the right side for each vehicle type. When the speed of bicycles is decreased, utilization of couriers with bicycles drops faster than the increase in utilization of cars or scooters. This is due to both cars and scooters becoming more utilized instead of only one vehicle type. At the same time, average delivery time increases non-linearly. This can be attributed to the fact that, despite the average delivery becoming slower, this effect is mitigated by the substitution of bicycle couriers by cars and scooters.

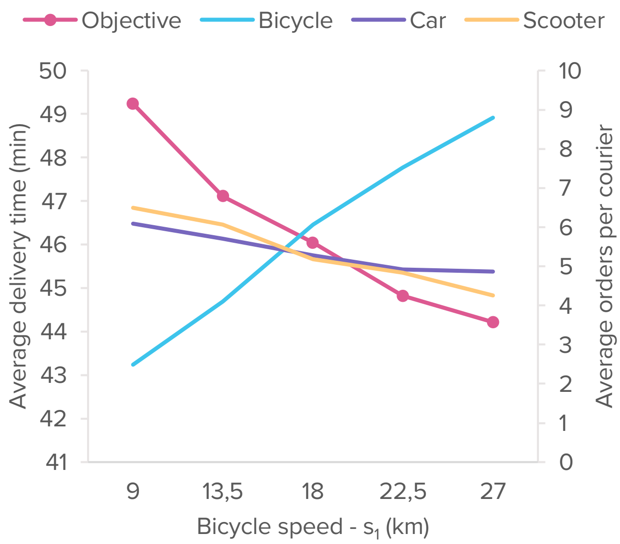

Figure 29. *Delivery time and courier utilization by variation in the speed of bicycles.*

The speed without congestion of motorized vehicles for each region is approximated using distance and the duration of a trip between two nodes using a third-party engine. Speeds for each region are varied with percentage intervals because the speed in every region is unique and decreasing the same amount in all would result in some regions' speed being set to unrealistic values. Table 16 registers the KPIs for intervals of 10% variation in the speed of motorized vehicles.

__Table 16.__ Average objective, requests served and courier utilization by bicycle (B), car (C) and scooter (S) for variations in the speed of motorized vehicles.

$s_{2,3}$	Obj	Requests Served			Courier Utilization		
		B	**C**	**S**	**B**	**C**	**S**
-20%	51	35%	31%	34%	8.4	4.9	4.4
-10%	48	33%	31%	35%	8.0	5.0	4.5
0%	45	31%	31%	38%	7.5	4.9	4.8
10%	43	30%	31%	39%	7.2	4.9	4.9
20%	42	27%	31%	42%	6.6	5.0	5.4
30%	40	23%	33%	44%	5.6	5.2	5.5
40%	39	20%	34%	46%	4.9	5.3	5.8

An increase in the speed of motorized vehicles makes couriers using these vehicles faster and more likely to be matched which, in turn, results in a decreasing delivery time. Figure 30 plots the average delivery time along with the utilization of bicycles, cars and scooters. Similarly to the sensitivity analysis performed on the speed of bicycles, the variation in the objective value is non-linear. It is also true that the utilization of couriers with bicycles drops more precipitously than the increase of motorized transports, but again this is due to the orders served by a single vehicle type being spread to both cars and scooters.

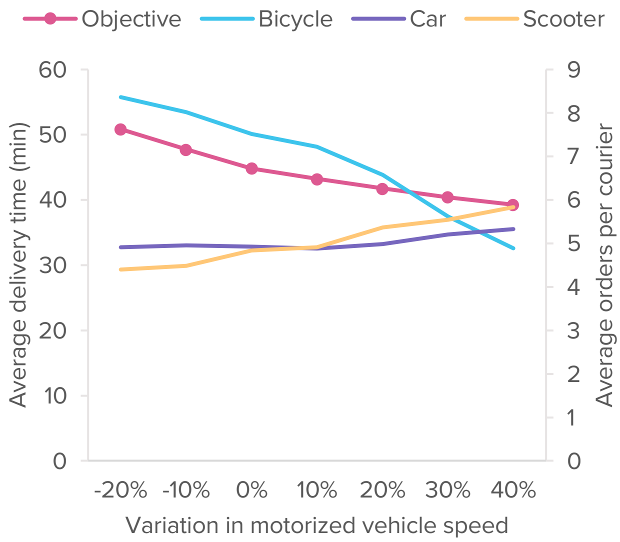

__Figure 30.__ Delivery time and courier utilization by variation in the speed of motorized vehicles.

Congestion

In the model, congestion is applied only to motorized vehicles. In reality, this might not be exactly the case. On the one hand, scooters usually can maneuver in an effective way and avoid traffic to some extent. On the other hand, even if bicycles have exclusive lanes or go through sidewalks, some roads end up being used and may face some congestion. For this reason, a sensitivity analysis is performed on the congestion factor of bicycles and the results are presented in Table 17. It is assumed that 100% is the scenario where the congestion factor is equal to that of motorized vehicles and 0% the scenario with no congestion. An increase in the congestion factor of bicycles increases the average delivery time in a small amount. The most drastic change comes at the level of the composition of assignments, that becomes proportional to the number of couriers available with each vehicle in the fleet. At the same time, the utilization of cars and scooters increases at the expense of bicycles.

Table 17. *Average objective, requests served and courier utilization by bicycle (B), car (C) and scooter (S) for variations in the congestion factor of bicycles.*

μ_1	Obj	Requests Served			Courier Utilization		
		B	C	S	B	C	S
0%	45	31%	31%	38%	7.5	4.9	4.8
20%	45	31%	31%	38%	7.5	4.9	4.8
40%	46	28%	32%	40%	6.8	5.1	5.0
60%	46	26%	33%	41%	6.3	5.2	5.1
80%	46	23%	33%	44%	5.6	5.2	5.6
100%	47	21%	33%	45%	5.1	5.3	5.7

Table 18 presents the results of the sensitivity analysis for motorized vehicles' congestion. Delivery time varies directly with the congestion factor variation, however, the changes in request by vehicle type and utilization are less substantial. Changes in congestion have a directly proportional effect on the KPIs of scooters, and inversely on bicycles. Cars, despite being motorized vehicles, are virtually unaffected by these changes. The reason for this might be due to the variation in congestion being small and the total number of scooters available in the system being greater than cars, especially at times with high traffic.

Table 18. *Average objective, requests served and courier utilization by bicycle (B), car (C) and scooter (S) for variations in the congestion factor of motorized vehicles.*

$\mu_{2,3}$	Obj	Requests Served			Courier Utilization		
		B	C	S	B	C	S
-30%	43	30%	31%	39%	7.3	4.9	4.9
-20%	44	30%	31%	39%	7.2	5.0	4.9
-10%	45	31%	32%	38%	7.4	5.0	4.8
0%	45	31%	31%	38%	7.5	4.9	4.8
10%	46	33%	31%	36%	8.0	4.8	4.6
20%	47	33%	31%	36%	8.0	4.9	4.5
30%	47	33%	31%	36%	8.0	4.9	4.5

In practice, scooters are less susceptible to congestion than cars, due to higher maneuverability, ability to navigate in between lanes in heavy traffic jams and higher acceleration that constitutes an advantage in stoplights. For this reason, a sensibility analysis is performed on the congestion factor of scooters alone and is recorded in Table 19. The results show that the effect on the objective value is greater than that of bicycle congestion, which is expected since the number of couriers with scooters is higher than that of couriers with bicycles. A reduction in the congestion of scooters leads to more orders being assigned to theses couriers at the expense of bicycles and cars and the same can be said about courier utilization.

The sensitivity analysis evidence the impact of certain parameters in the performance of the model. For this reason, it is important to deepen the knowledge regarding the service times and the distinction in congestion between the various modes of transport.

***Table 19**. Average objective, requests served and courier utilization by bicycle (B), car (C) and scooter (S) for variations in the congestion factor of scooters.*

μ_3	Obj	Requests Served			Courier Utilization		
		B	C	S	B	C	S
0%	41	24%	25%	52%	5.7	3.9	6.5
20%	42	25%	26%	50%	5.9	4.1	6.3
40%	43	27%	28%	45%	6.6	4.5	5.7
60%	44	30%	29%	41%	7.3	4.5	5.2
80%	44	30%	30%	40%	7.2	4.7	5.1
100%	45	31%	31%	38%	7.5	4.9	4.8

5.3 Policy Analysis

The baseline model does not necessarily provide the best delivery time or the best results for other KPIs. Analyzing the results obtained for different policies provides valuable insights about the trade-offs and suggests strategies to follow in order to achieve desired goals.

5.3.1 Available until Pickup

The baseline policy considers that immediately after an assignment, the assigned order is permanently removed from the list of orders waiting for assignment and the assigned courier becomes unavailable until finishing this job. This means that, if minutes later a new courier arrives that could improve the overall objective, such option is never contemplated. To prevent this issue, the assignment until pickup policy is incorporated that allows couriers and orders to remain available for assignment up until the courier reaches the pickup point. This implies that if in the future a courier logs in or finishes a delivery task and is faster to reach the pickup point, the assignment can be changed. The policy also requires the current location of the pre-assigned courier to be updated until the assignment is made permanent or nullified. Recalling Figure 14, all changes concerned in this policy take place at the state and timeline update stages. Figure 31 shows a timeline with all the steps of a delivery in the model. When an order is assigned, there is a waiting time accounting for the time couriers take to accept the job (τ^W) and, afterwards, the courier starts to travel towards the pickup location, reaching it t^P minutes after the confirmation.

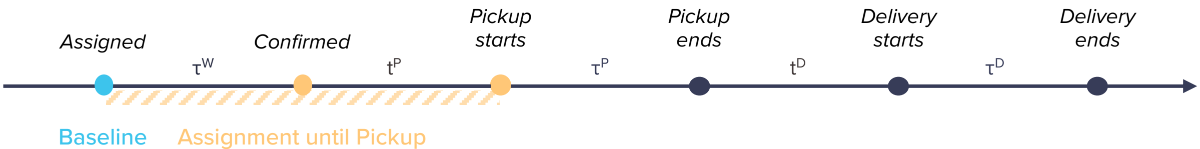

***Figure 31.** Delivery timeline with assignment windows for different policies.*

The framework with the available until pickup policy is run using the JV and BC algorithms, considering as input the same instance used for the baseline. Table 20 compares the baseline and available until pickup policies in terms of running time and objective value. The average time it took to run the model increased six and eight times for the JV and BC approaches, respectively, when compared to the baseline. This is due to the excess of state updates that this policy adds and the fact that each assignment becomes larger with more couriers and orders. Even the case in which the BC approach takes more than two minutes to simulate the whole day is fast enough to handle real-time results. In terms of the total objective, there is a reduction of 1717 minutes for the JV algorithm and 2032 minutes for the BC algorithm when compared to the baseline, or on average a reduction of 1 minute and 37 seconds and 1 minute and 49 seconds, respectively. The policy results in the objectives for the two approaches becoming more similar since, with more couriers and orders in an assignment decision, the probability of a tie is smaller.

Table 20. *Running time and objective values for the baseline and assignment until pickup policies.*

Policy	Algorithm	Running Time	Total Objective	Average Objective
Baseline	JV	9.7 s	47 599 min	44 min 49 s
	BC	17.3 s	47 944 min	45 min 3 s
Available until Pickup	JV	61.3 s	45 882 min	43 min 12 s
	BC	137.1 s	45 912 min	43 min 14 s

Table 21 compiles the percentage of requests served by each vehicle type, as well as the relevant metrics of courier utilization for the baseline and the new policy for both algorithms. The addition of this policy has little to no effect on the percentage of orders attributed to each type of vehicle, yet still affects courier utilization. When compared with the baseline, the assignment until pickup policy results in changes in the couriers with no orders, the couriers with more orders and the variation in the number or order per courier. While the baseline policy results in three and two couriers not being assigned to any order for the JV and BC algorithms respectively, for the assignment until pickup both algorithms result in only one courier with no assignments. Moreover, the maximum number of orders a scooter courier performs is also reduced. The average courier utilization remains practically the same, with modest increases for scooters and decreases for cars. There is also a reduction in the standard deviation, which is consistent with the increase in the orders performed by the courier with less assignments and the one with more assignments. Although the major objective of the policy is to reduce the delivery time, it produced an improvement on the courier utilization metrics with less couriers having no jobs or excessive workloads.

Table 21. *Requests served and courier utilization by vehicle type.*

Policy	Vehicle	Requests	Min	Mean	Max	Std. Dev
Baseline (JV)	Bicycle	31%	1	7.5	17	4.1
	Car	31%	0	4.9	11	2.4
	Scooter	38%	0	4.8	15	2.9
Baseline (BC)	Bicycle	32%	1	7.7	18	4.4
	Car	31%	0	4.9	13	2.6
	Scooter	37%	0	4.7	16	2.8
Assignment until Pickup (JV)	Bicycle	32%	2	7.6	17	4.0
	Car	30%	1	4.7	11	2.3
	Scooter	39%	0	4.9	13	2.6
Assignment until Pickup (BC)	Bicycle	31%	1	7.6	18	3.9
	Car	31%	1	4.8	11	2.3
	Scooter	38%	0	4.8	12	2.6

In terms of delivery time by vehicle and when compared with the baseline policy (BP), this policy (AuP) registered an overall reduction and greater similarity between the JV and BC algorithms. The same can be said for the delivery time for each period. Figure 32 compares the results of delivery time per vehicle type and Figure 33 by period.

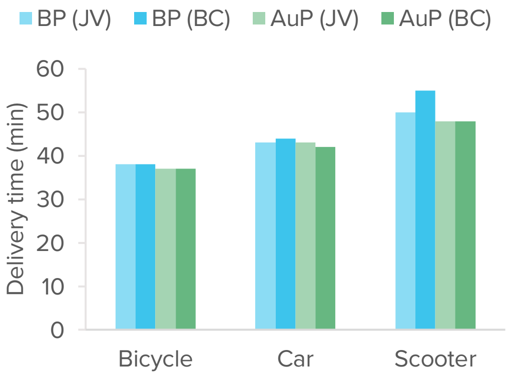

Figure 32. *Delivery time by vehicle type, in minutes.*

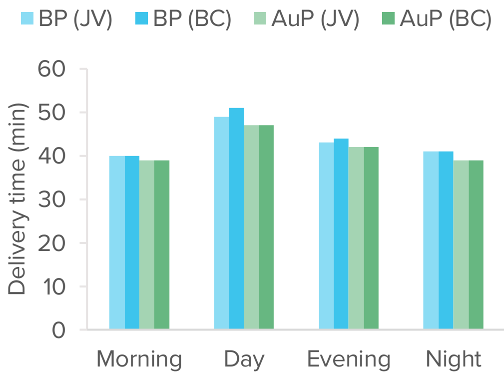

Figure 33. *Delivery time by vehicle period, in minutes.*

Regionally, the delivery time is, by the most part, improved. Figure 34 represents a map of the city where each borough is colored according to the value of the ratio between the delivery time for the assignment until pickup policy and the baseline policy. The boroughs of Barnet and Hackney register, on average, longer delivery times with the new policy, although only worse by approximately one minute. The rest of the regions either reduce the delivery time or maintain it. The boroughs of Hillingdon and Richmond upon Thames register the biggest improvements, of approximately 20%. Recalling Figure 28, those were two regions that performed the worst against the real assignments. The drastic improvement might be linked to these regions having less couriers that are moved in the early periods to areas with more requests, which would have left the region without the means to respond promptly to new orders. With long assignment windows, if a request arrives in an area without couriers, a distant courier is mobilized and starts driving in that direction. However, if a courier starts or finishes delivering an order and is closer to the pickup, a new match is established and the courier that was coming to aid can remain in the same region and serve local requests.

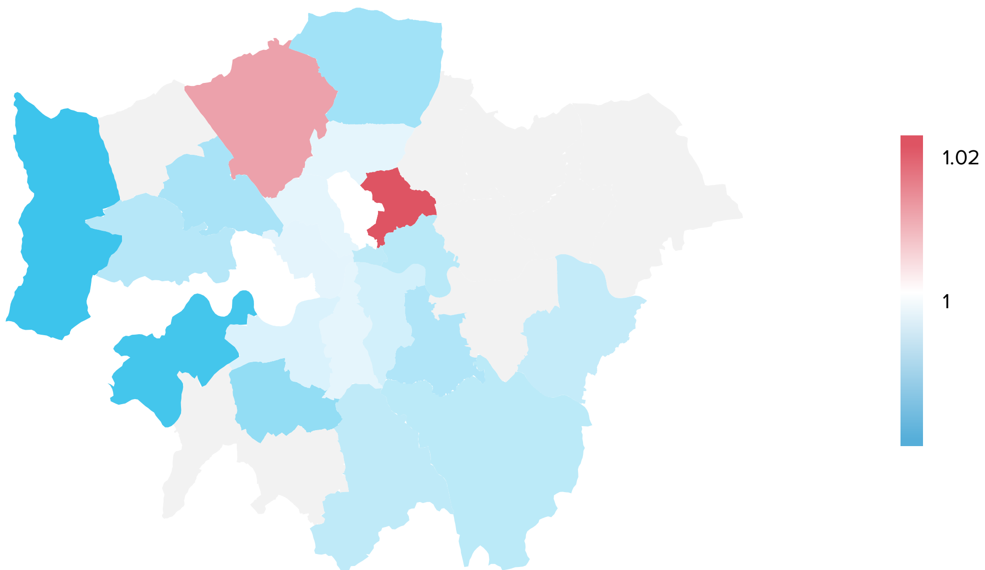

Figure 34. *Ratio of delivery time between the assignment policy and the baseline for the JV algorithm.*

The assignment until pickup policy also raises concerns about the time couriers spend on assignments that fail to materialize. This requires specific metrics to quantify and evaluate the policy and, to achieve this, the KPIs of number of failed assignments, average time between request submission and definitive assignment, as well as the average time couriers spend on unrealized assignments are added. For the entire running period, there is a total of 156 unrealized assignments. Assuming that each one is equivalent to an order, which might not be the case since one order can be re-assigned many times, this implies that 15% of orders or less are re-assigned at some point. Scooters are the most common transport among temporary assignments, making up 48%, followed up by cars with 33% and bicycles with 19%. Congestion makes bicycles a more appealing vehicle for delivery, which explains why they are unassigned less, while scooters make up the bulk of the fleet, making them more likely to get unassigned. Figure 35 represents the number of failed assignments by period and differentiated by type of vehicle for the JV algorithm. The morning registers only seven failed assignments due to the ratio of couriers to orders being high. During the day this number increases drastically, peaking in the evening and then decreasing at night. The proportion of vehicles assigned to orders is consistent to the fleet composition, with more cars in the morning and bicycles in the evening. Figure 36 presents the same results using the BC algorithm.

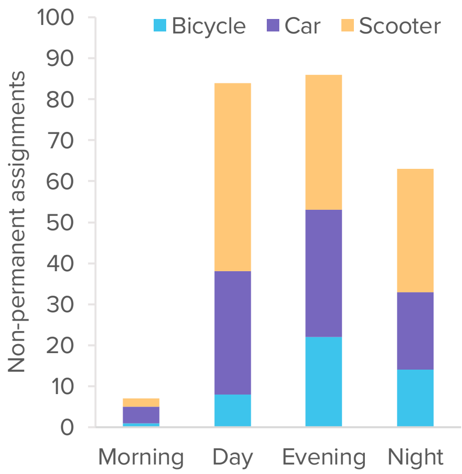

Figure 35. *Number of non-permanent assignments per vehicle type, by period, for the JV algorithm.*

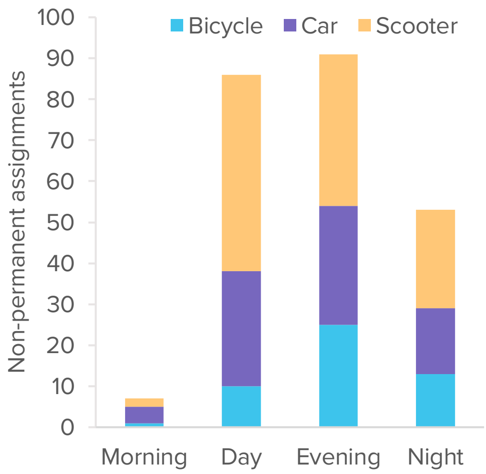

Figure 36. *Number of non-permanent assignments per vehicle type, by period, for the BC algorithm.*

Regionally, there is no clear trend that could be linked to an excessive number of re-assignments. Figure 37 shows the failed assignments per region for the two approaches. The borough of Brent is by far the one with more non-permanent assignments. This is caused by the abnormal number of orders that must be performed by scooters that arrive in the day period and is further discussed in the baseline subchapter.

The assignment time comprises the period between an order being submitted and the time at which the order is assigned permanently to a courier. For the baseline model, this period is zero if there are enough couriers that respect the vehicle restrictions, or longer otherwise. In practice, due to the shortage of couriers during some periods, the average assignment time is 35 seconds for the JV and 37 seconds for the BC algorithms, respectively. For the available until pickup policy, this value averages 3 minutes and 3 seconds for both algorithms. This means that, by taking on average two and half minutes longer on the assignment, the policy is able to reduce the total delivery time, which already includes the assignment time, by 1 minute and 37 seconds for the JV algorithm and by 1 minute and 49 seconds for the BC algorithm, *i.e.*, equivalent to a 3,6% and 4,2% reduction respectively.

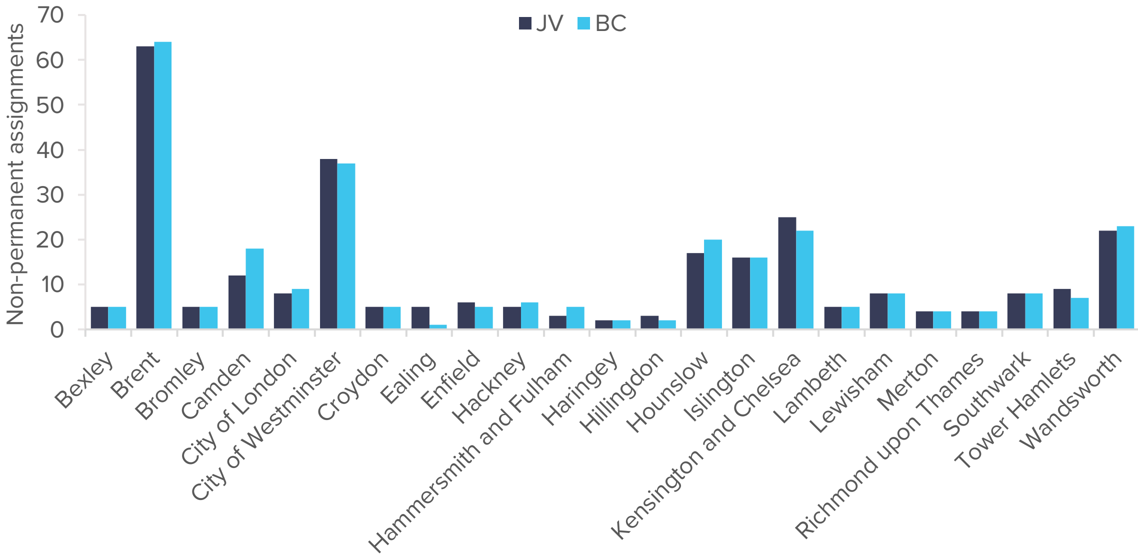

Figure 37*. Number of non-permanent assignments per region.*

Figure 38 and Figure 39 represent the average assignment time by period for the JV and BC algorithm. There is a clear peak in the morning period for scooters, that coincides with the appearance of scooter-restricted orders in the region of Brent mentioned before. Cars are the transport that takes the longest to assign during the day, evening and night periods, which might be partially influenced by congestion. However, since scooters take less time to be matched, other factors such as the initial location of both transports might account for the difference.

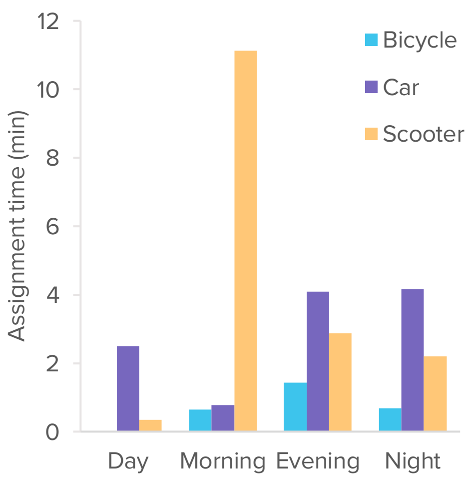

Figure 38. *Average assignment time by period per vehicle type for the JV algorithm, in minutes.*

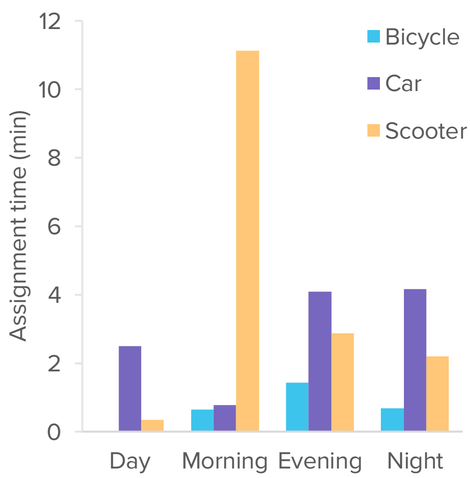

Figure 39. *Average assignment time by period per vehicle type for the BC algorithm, in minutes.*

Figure 40 displays the values for average assignment time of orders by borough. The results show that, regionally, the assignment time does not correlate with the sheer number of non-permanent assignments in Figure 37, meaning that some regions can have many re-assignments that result in orders having a small assignment time or vice-versa, *e.g.*, if an order is re-assigned three times in three minutes, the total assignment time is just three minutes. By contrast, an order can be re-assigned just one time after five minutes of the first assignment and this would result in an assignment time of 5 minutes. The region of Brent is one of a few cases where there are many re-assignments that take on average 20 minutes.

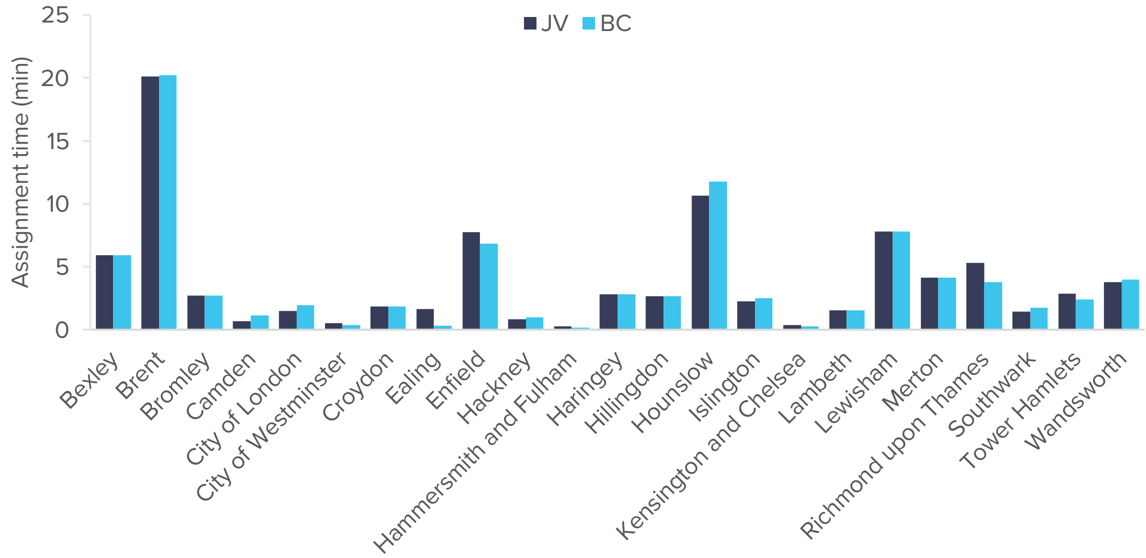

Figure 40. *Average assignment time per region, in minutes, for the JV and BC algorithms.*

From the couriers' perspective, the average agent spends 234 minutes on full deliveries and 4 minutes and 42 seconds on failed assignments, which is equivalent to less than 2% of the active time. There is variability concerning both the delivery time and time spent on failed assignments. Table 22 presents relevant statistics with regards to the time couriers spend delivering and on failed assignments. Despite differences, both approaches follow the same trends. Couriers with scooters are tangled up, on average, more time than bicycles or cars, while bicycle couriers spend the most time of the three delivering.

Table 22*. Statistical information regarding the time couriers spend on delivery and on assignments, in minutes.*

Algorithm	Vehicle	Permanent				Non-permanent			
		Min	Mean	Max	Std. Dev	Min	Mean	Max	Std. Dev
JV	Bicycle	71.6	285.1	660.0	153.4	1	3.3	13	2.5
	Car	31.9	201.1	497.9	101.8	1	3.8	10	2.4
	Scooter	0	231.3	536.5	120.5	1	5.9	16	3.4
BC	Bicycle	33.4	282.6	680.8	144.6	1	3.2	13	2.6
	Car	55.5	204.4	486.7	95.6	1	3.5	10	2.3
	Scooter	0	230.4	532.3	126.1	1	6.0	18	3.8

A bicycle courier spends eleven hours, in total, delivering. This is not exclusive to the assignment until pickup policy but working that amount with little rest might not be feasible in real life. For that reason, a policy that can effectively balance the workload is of need. This being said, it can be concluded that, while the policy improves the overall objective and alleviates the problem of the high number of orders with a restriction in the region of Brent, by no means does it completely solve the problems of surge of scooter-restricted orders or unbalanced workload.

5.3.2 Bicycle Policy

For environmental concerns or in anticipation of future legislations limiting the circulation of motorized vehicles, it is interesting to study how can more bicycles be employed and what are the effects in the total objective and other KPIs. The policy consists in giving priority to couriers with bicycles for orders whose total distance, from courier to pickup location plus pickup to delivery location, is less than a fixed value. This is done by reducing the objective value for couriers with bicycles for nearby orders before solving the model. The policy is applied over the baseline approach using the same parameters, but also applied over a set of modified parameters based on the insights gained from the sensitivity analysis.

Table 23 shows the results of the running time and objectives for the baseline policy and the bicycle policy, applied only to orders whose travelled distance is lower than or equal to three kilometers, using the two algorithms. The two policies barely differ in terms of running time, however the objective values increase which is expected since the policy introduces an inefficiency by not assigning the order to the best match and instead favoring bicycle couriers.

Table 23. *Running time and objective values for baseline and bicycle policies.*

Policy	Algorithm	Running Time	Total Objective	Average Objective
Baseline	JV	9.7 s	47 599 min	44 min 49 s
	BC	17.3 s	47 944 min	45 min 3 s
Bicycle (d^{max} = 3km)	JV	9.6 s	48 146 min	45 min 20 s
	BC	17.7 s	48 077 min	45 min 16 s

Contrary to the baseline results, where the JV algorithm performed better, for the bicycle policy it is the BC algorithm that is faster. Due the difference being less than one percent, it is not possible to discern if it is caused by the policy or to small variations in the assignments. Figure 41 represents, over the course of the day, the number of equal and different assignments between the two algorithms. For the bicycle policy, and compared to the baseline, there are thirty-two more different assignments and thirty-two less equal assignments. This excess of different assignments is attributed to the fact that with the policy, for bicycle courier under the maximum distance, bicycles are always preferred independently of the performance which results in more ties.

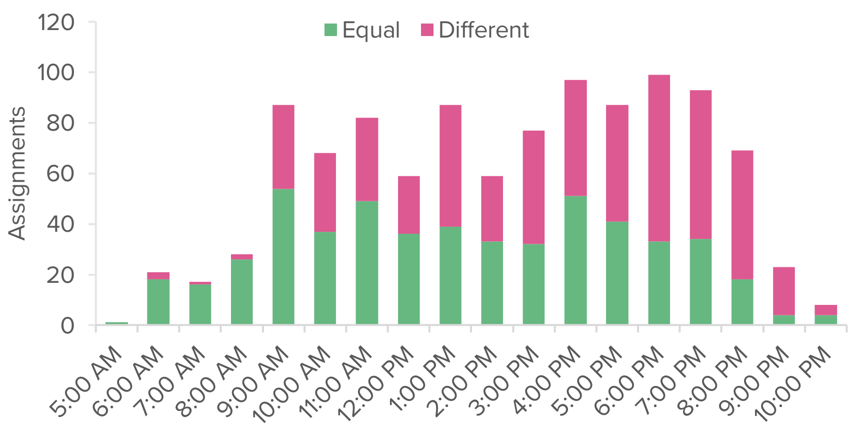

Figure 41. *Number of equal and different assignments between the JV and BC algorithms for the bicycle policy, for every hour.*

Since the policy is only applied to couriers and orders whose total travel distance does not exceed a fixed value, it is important to study how this parameter influences the KPIs. Table 24 present the objective value, the percentage of orders served by each vehicle and the courier utilization. The results indicate that with an increase in the maximum distance for the policy, there is an increase in the proportion of bicycles and in the overall utilization of bicycle couriers, at the expense of both cars and scooters, which was expected and intended by the policy. However, this effect is accompanied by slower deliveries.

Table 24. *Average objective, requests served and courier utilization by bicycle (B), car (C) and scooter (S) for variations in the maximum distance to apply the policy.*

d^{max}	Obj	Requests Served			Courier Utilization		
		B	C	S	B	C	S
0	45	32%	31%	37%	7.6	4.8	4.7
1.5	45	32%	31%	37%	7.6	4.9	4.7
3	45	34%	30%	36%	8.2	4.8	4.5
4.5	45	35%	30%	35%	8.5	4.7	4.4
6	46	38%	29%	33%	9.1	4.7	4.3
7.5	47	38%	28%	33%	9.2	4.6	4.2
9	48	39%	28%	33%	9.4	4.5	4.2

Figure 42 plots the average objective value and the proportion of requests served by each vehicle type. By looking at the graph it becomes apparent that for a distance above 4.5 kilometers, there is a steep increase in delivery time, which is due to a higher number of orders above that distance on one side, and to the fact that misallocating couriers for small distances does not change much the delivery time, while misallocating couriers that are further apart means that these couriers have to travel longer distances which lengthens the deliveries. For this reason, the policy should be set only for small distances, which means that the increments in bicycle utilization are also lower. The policy might still be worthy of being applied for small distance because it does not influence in a significant manner the overall objective and occupies couriers riding bicycles with closer orders, leaving more distant requests for motorized couriers that are not as affected by fatigue from long trips.

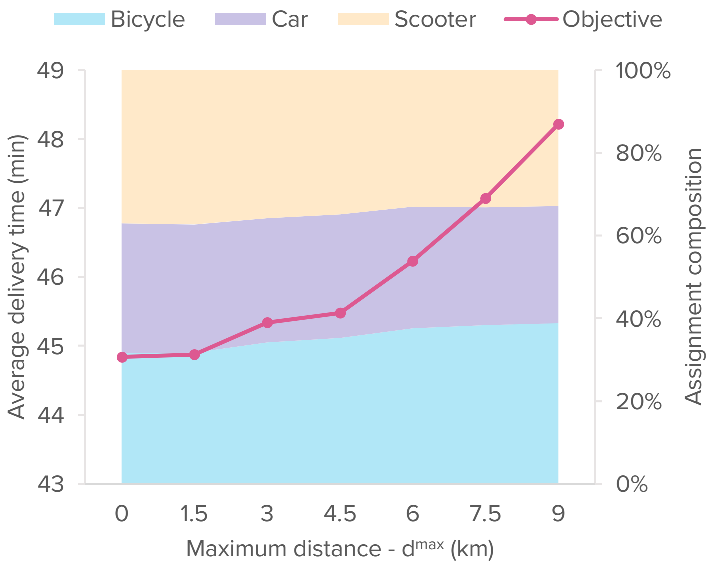

Figure 42. *Average delivery time and assignment composition for variations in the maximum distance to apply the bicycle policy.*

As evidenced by the sensitivity analysis in Subchapter 5.2, the speed and congestion parameters heavily influence the model. Furthermore, scooters are less affected by congestion and bicycles are unlikely to be totally immune. Additional experiments are conducted using the same instance applied to previous policies but changing the speed and congestion parameters. The speed of bicycles is set to the same as the motorized vehicles whenever the regional speed is less than 22.5 km/h and that value for the regions where motorized vehicle speed is higher. While congestion was kept the same for cars, reduced to 60% for scooters and increased to 40% for bicycles. The results using this set of parameters were computed and are represented in the first row of Table 25. Similarly to the previous analysis, the table compares the objective value, with the composition of the assignments and the courier utilization by vehicle type.

Table 25. *Average objective, requests served and courier utilization by bicycle (B), car (C) and scooter (S) for variations in the maximum distance to apply the policy using modified parameters.*

d^{max}	Obj	Requests Served			Courier Utilization		
		B	C	S	B	C	S
0	44	25%	30%	45%	6.0	4.7	5.8
1.5	44	26%	29%	45%	6.3	4.5	5.7
3	44	29%	29%	42%	6.9	4.6	5.3
4.5	45	32#	27%	41%	7.8	4.2	5.2
6	46	35%	27%	38%	8.5	4.3	4.8
7.5	46	37%	27%	37%	8.8	4.3	4.6
9	48	37%	26%	37%	9.0	4.2	4.6

The results are analogous to the previous analysis with slower deliveries for an increase in the maximum distance to apply the policy. At the same time the proportion of orders performed by bicycles and the utilization increases at the expense of other modes of transport. What differs relative to the previous analysis is the magnitude of the changes. Figure 43 shows the average delivery time and the proportion of orders delivered by each vehicle. Compared to Figure 42, the increase in the utilization of bicycles is proportionally higher. The difference in the number of orders served by bicycles rises twelve percentage points between the scenario without this policy (distance zero) and the scenario where the policy is applied for a maximum distance of up to nine kilometers. This happens because with the modified input motorized transports are proportionally more competitive to bicycles. Using the original input, the model heavily favored bicycles, which meant that the policy gave preference to an already favored mode of transport. The figure also evidences the fact that applied to distances higher than 4.5 kilometers, the policy leads to an increase in delivery time that can be detrimental.

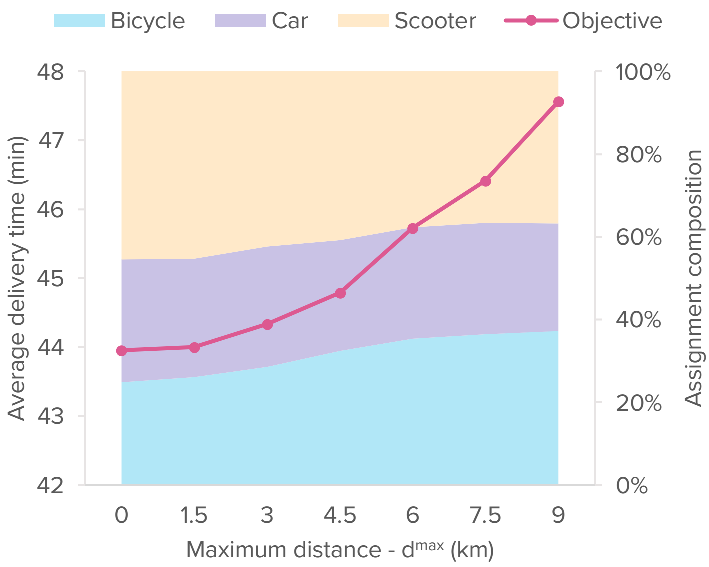

***Figure 43**. Average delivery time and assignment composition for variations in the maximum distance to apply the bicycle policy for the modified data.*

The effects of the policy are mixed. On one hand, a forced increase in bicycle courier utilization naturally leads to higher delivery times which impairs the performance of the company at the eyes of the consumer. On the other hand, the increased courier satisfaction or a decrease in direct emissions might still be worth to pursue. An intermediary policy, where bicycles are given preference for distances of three kilometers or less, can be beneficial by not only increasing the usage of bicycles, but also by ensuring that couriers with bicycles deliver closer orders and are not tired for pedaling.

5.4 Chapter Conclusions

The processes to treat raw data and produce inputs that directly feed the model are described. Additional parameters of speed, congestion and circuity are defined based on external data, which requires making assumptions with the purpose of minimizing discrepancies between the model and the real-life scenario. A series of experiments are conducted with the objective of testing the applicability of the model, improving the assignment solution and studying the trade-offs of different policies.

JV and BC approaches are used to solve the model and both produced identical results. Both algorithms can handle large instances and provide matches in real time to be used in delivery operations. Compared to the company's assignment, the JV approach results in a total reduction of 403 minutes or 0.8% and the BC in a reduction of 158 minutes or 0.3%. Both models also result in higher average courier utilization than those observed in the real-life assignment scenario. In order to test the model, several sensitivity analyses are performed on the service time, speed and congestion parameters, which evidenced that service time and congestion have a significant impact on the model and, for that reason, more attention should be dedicated in setting those parameters.

A policy that allows for assignments to be re-done until the courier arrives at the pickup point is tested using both approaches, which results in a total reduction of 2 120 minutes and 2 090 minutes for the JV and BC algorithms, respectively, when compared to the real assignment. This is equivalent to a total reduction of approximately 4.4% for both approaches, which is also accompanied by improvements on courier utilization balance, whose variation is reduced by 9.6%. The policy has the caveat of making some couriers start to travel only for the request to be reassigned to another courier, however, the average time spent with non-permanent assignments is only 2 minutes and 25 seconds. Given that the couriers take 2 minutes to accept a request, this added time is a minor obstacle that rarely affects couriers and can be considered a net positive policy.

A second policy is studied regarding the allocation of bicycle couriers. In this approach, for closer orders, priority is given to bicycles as opposed to motorized vehicles. The policy does not influence the running time of the model; however, it increases the total delivery time. If the maximum distance to apply the policy is set to 4.5 km or less, the inefficiency introduced is smaller, *e.g.*, for a maximum distance of 3 km, the increase in the total delivery time is of 0.3% and 0.2% for the JV and BC algorithms, respectively, compared to the real assignment. On the other hand, the policy increases the share of orders allocated to couriers with bicycles in 14pp in comparison with the real assignment and 3pp in comparison with the baseline. It is demonstrated that the policy can be more effective depending on the parameters used.

Both the baseline and the policies fail at dealing with periods when an influx of orders restricted to a vehicle type occurs. The real assignment copes with this and outperforms the model by using a pooling strategy that allows to assign more than one order per job to a courier. The assignment until pickup strategy helps to ameliorate this effect to a certain extent, but by no means solves it, and routing might be the only effective tactic in dealing with this issue.

6 Conclusions and Future Research

The on-demand industry is characterized by heavy competition among start-ups. IDs are no exception, where competition is not only restricted to customers, but also extends to hiring and retaining couriers that use their personal modes of transportation to perform deliveries. This results in a heterogeneous fleet that raises new challenges to the optimization of delivery operations, which are further complicated by the dynamic nature of requests and couriers' schedules, congestion and regional urban networks that affect mobility. Hence, the development of an optimization model is crucial to, in real-time, assign couriers to orders for large sets of data and subsequently analyze the results and study the implementation of policies. Such model can be used by delivery platforms and has the potential to reduce costs and improve the service level provided to customers, while ensuring that the workload is spread among couriers, that are more likely to remain motivated.

By studying the ID environment, it becomes clear that this industry has experienced enormous growth in recent years and it is not easy to hire new couriers in sufficient number, making superior optimization models a necessity. Most ID companies started in a specific niche but have since broaden the range of products that can be purchased. Also, most companies in the field have a P2C business model and employ crowdsourced workers, meaning that the fleets are heterogeneous which, together with the wider range of products, makes the optimization problems more complex.

ODD is broadly defined to describe fast systems that deliver orders in the same day as they were placed. Some authors apply the term ID to refer, more specifically, even faster ODDs, which usually take 45 minutes or less and are common in meal delivery. ID problems were formulated in the past under different designations, namely *assignment* or *routing problems*. Most problems considered few policies or restrictions. Only a third studied vehicle heterogeneity of some sort. Traffic congestion was only studied by two authors and, despite 86% of the selected articles considering dynamic elements, these were almost always linked to the arrival of orders and not to the arrivals and exits of couriers or congestion.

The case under study constitutes an example of a complex ID system, with crowdsourced couriers that can choose their schedules, multiple modes of transportation, multiple pickups and delivery points and many types of products with different sizes and shapes. Modelling all these aspects requires a dynamic model capable of handling the vehicle restrictions. The dynamic model can be solved using a JV or a BC algorithm, that solve the model fast enough to be used in real delivery operations.

A myopic baseline approach is used, which constitutes an improvement in the total delivery time objective of 0.8% using the JV algorithm and 0.3% using the BC algorithm when compared with the real assignment, and an increase of 4.5% and 1.4%, respectively, in balanced courier utilization. These results are further improved by applying a policy that enables re-assignments until the courier arrives at the pickup point. This policy achieves a reduction in total delivery time of 4.4% for the two algorithms and accomplishes an increase in balanced utilization of 9.6% when compared to the real assignment, while still being fast enough to be used in real applications. On the other hand, this policy has the inconvenient of sometimes making couriers start to work on an order only for the job to be assigned to another courier; however, on average, this rarely happens since, in the model, couriers take 2 minutes to accept an order and the average assignment time with the policy is only 2 minutes and 25 seconds, making the trade-off worth considering. Additionally, a policy that prioritizes bicycle utilization for short distances is also studied. Applied only to orders that require travelling for 3 minutes or less, the total delivery time increases by 0.3% and 0.2% for the JV and BC algorithms, respectively, while the courier utilization rises by 14pp compared to the real assignment. Even with this inefficiency, the policy might still be worth applying if a company has the objective of reducing direct carbon emissions. As evidenced by the sensitivity analyses, the model is heavily influenced by pickup time and the relative differences in speed and congestion between vehicle types.

The developed model presents, as limitations, the fact that the estimations of bicycle and scooter congestion parameters, as well as service time and speed parameters, could be further improved by collecting and processing more specific data, which would increase model reliability. Furthermore, the model does not include stochasticity, which can be linked, in this specific case, with preparation time or delivery time. The developed model should also be tested with other instances, however, the instances available in literature consider fewer couriers and orders, and do not contemplate different vehicle types.

It is relevant to mention that the model underperforms compared to the real assignment when localized peaks of orders restricted to a specific type of vehicle occur. The real assignment deals with this problem by pooling orders together. It was attempted to combine many orders in a single job by building a routing model and integrating it within the dynamic framework. The model followed a *VRP with Pickup and Delivery and with Time Windows* (VRPPDTW) formulation (Toth and Vigo 2002), but the integration within the dynamic framework was not achieved due to a lack of time. By studying the routing model in isolation, it became clear that, for large instances, it is unfeasible to run the model in real time. Solving such problem requires using a dynamic approach, subdividing the problem into multiple regional problems, applying it only when there is a lack of vehicles of a given type or even aggregating the orders first and then assigning the pools of orders.

More attention has been devoted to the field of ID in recent years, however, there are still gaps in the existent literature and opportunities for future research. The inclusion of multiple objectives has only been studied by Liao et al. (2020) and, given that customers, couriers and partner restaurants or stores all interact with the ID company, an optimization model that reflects the objectives of each part should be studied. Other aspects of IDs should be captured in more detail, such as the inclusion of courier rejection of delivering jobs, order priority (which can be enforced by time windows) and heterogeneous requests by differentiating the service time at the pickup of restaurants and regular stores, *e.g.*, restaurants prepare orders which might make the courier wait, but if the order is already prepared, the pickup time should be nearly instantaneous, while in stores the courier has to pick and purchase the items, which requires more time. The addition of droids as a vehicle type can be interesting since there are companies already using them as alternatives or complements to human couriers. More emphasis should be placed on studying dynamic approaches by, instead of a fixed increment of time, having a variable step of time depending on the time of day or the number of couriers available and orders. The study of additional policies might provide valuable insights to be applied by delivery platforms. Reposition policies might improve the overall performance by incentivizing couriers to move to areas with higher demand of orders or lower supply of couriers. Additionally, this policy can be directed to relocate motorized vehicles to less congested areas. Finally, it might be of interest to study policies that evaluate the number of each type of vehicles in a fleet and, if the number of vehicles of a certain type is scarce, the policy holds it back and assigns the order to a vehicle of another type to prevent future shortages.

Bibliography

Ahuja K, Chandra V, Lord V, Peens C (2021) *Ordering in: The rapid evolution of food delivery* (McKinsey & Company).

Anon (2021) #42 - Bassel El Koussa - CEO @ Quiqup. *Major Ring Exp.*

Arslan AM, Agatz N, Klapp MA (2021) Operational strategies for on-demand personal shopper services. *Transp. Res. Part C Emerg. Technol.* 130:103320.

Arslan AM, Agatz N, Kroon L, Zuidwijk R (2019) Crowdsourced Delivery—A Dynamic Pickup and Delivery Problem with Ad Hoc Drivers. *Transp. Sci.* 53(1):222–235.

Auad R, Erera A, Savelsbergh M (2022) Courier Satisfaction in Rapid Delivery Systems Using Dynamic Operating Regions. *SSRN Electron. J.*

Aull B, Begley S, Chandra V, Mathur V (2021) *Making online grocery a winning proposition* (McKinsey & Company).

Barnes J (2019) Council Post: Dark Stores: The Key To Online Grocery Efficiency And Profitability. *Forbes*. Retrieved (June 3, 2022), https://www.forbes.com/sites/forbestechcouncil/2019/09/27/dark-stores-the-key-to-online-grocery-efficiency-and-profitability/.

Belk R (2014) You are what you can access: Sharing and collaborative consumption online. *J. Bus. Res.* 67(8):1595–1600.

Bertsimas D, Jaillet P, Martin S (2019) Online Vehicle Routing: The Edge of Optimization in Large-Scale Applications. *Oper. Res.* 67(1):143–162.

Blake T (2022) Best Times To Do Instacart (2022) - What Shoppers Need To Know. *This Online World*. Retrieved (June 3, 2022), https://thisonlineworld.com/instacart-best-hours-to-work/.

Bolt Food (2018) Sign-up as a Food Delivery Courier | Bolt Food. Retrieved (June 3, 2022), https://couriers.bolt.eu.

Bourke J (2021) ‘Ocado Zoom’ same-day grocery service set to expand in London. *Evening Stand.* Retrieved (June 3, 2022), https://www.standard.co.uk/business/leisure-retail/ocado-zoom-b919155.html.

Boxall A (2017) How the Ocado CargoPod Autonomous Delivery Vehicle Works. *Digit. Trends*. Retrieved (June 3, 2022), https://www.digitaltrends.com/cool-tech/ocado-cargopod-autonomous-delivery-vehicle-test/.

Bozanta A, Cevik M, Kavaklioglu C, Kavuk EM, Tosun A, Sonuc SB, Duranel A, Basar A (2022) Courier routing and assignment for food delivery service using reinforcement learning. *Comput. Ind. Eng.* 164:107871.

Cardillo A, Scellato S, Latora V, Porta S (2006) Structural properties of planar graphs of urban street patterns. *Phys. Rev. E* 73(6):066107.

Cattrysse DG, Van Wassenhove LN (1992) A survey of algorithms for the generalized assignment problem. *Eur. J. Oper. Res.* 60(3):260–272.

Chen J fang, Wang L, Wang S, Wang X, Ren H (2022) An effective matching algorithm with adaptive tie-breaking strategy for online food delivery problem. *Complex Intell. Syst.* 8(1):107–128.

Chen X, Ulmer MW, Thomas BW (2022) Deep Q-learning for same-day delivery with vehicles and drones. *Eur. J. Oper. Res.* 298(3):939–952.

Colby C (2019) On Demand Economy Research. *Rockbridge*. Retrieved (June 3, 2022), https://rockresearch.com/on-demand-economy-research/.

Convoy (2021) The Leading Digital Freight Network. *Convoy*. Retrieved (June 3, 2022), https://convoy.com/.

Cordeau JF, Laporte G, Potvin JY, Savelsbergh MWP (2007) Chapter 7 Transportation on Demand. *Handb. Oper. Res. Manag. Sci.* (Elsevier), 429–466.

Coresight Research (2021) Most important factors when choosing rapid delivery services for groceries in the United States in 2021. *Statista*. Retrieved (June 3, 2022), https://www.statista.com/statistics/1284670/main-drivers-of-rapid-delivery-groceries-united-states/.

Dablanc L, Morganti E, Arvidsson N, Woxenius J, Browne M, Saidi N (2017) The rise of on-demand 'Instant Deliveries' in European cities. *Supply Chain Forum Int. J.* 18(4):203–217.

Dayarian I, Savelsbergh M (2020) Crowdshipping and Same-day Delivery: Employing In-store Customers to Deliver Online Orders. *Prod. Oper. Manag.* 29(9):2153–2174.

Deliveroo (2022) Deliveroo. Retrieved (June 3, 2022), https://deliveroo.co.uk/.

Dijkstra EW (1959) A note on two problems in connexion with graphs. *Numer. Math.* 1(1):269–271.

Domino's Pizza (2021) Domino's Pizza Portugal, Encomenda Online. *Dominos Pizza Port. Encomenda Online*. Retrieved (June 3, 2022), https://dominospizza.pt.

DoorDash (2020) DoorDash Food Delivery & Takeout - From Restaurants Near You. Retrieved (June 3, 2022), https://www.doordash.com/.

Dudas M (2014a) Supply-Side Challenges Of The On-Demand Economy. *TechCrunch*.

Dudas M (2014b) The On-Demand Economy By The Numbers. *Just Dudas*.

Edison Trends (2021) 2021 Edison Trends U.S. Convenience Store Delivery Sales Report. Retrieved (June 3, 2022), https://trends.edison.tech/research/convenience-store-delivery-sales.html.

Eksioglu B, Vural AV, Reisman A (2009) The vehicle routing problem: A taxonomic review. *Comput. Ind. Eng.* 57(4):1472–1483.

Faoite M de (2021) Gorillas — Dark stores and e-bikes. *Medium*. Retrieved (June 3, 2022), https://blog.spaaza.com/gorillas-dark-stores-and-e-bikes-2330987015fa.

Fikar C, Hirsch P, Gronalt M (2018) A decision support system to investigate dynamic last-mile distribution facilitating cargo-bikes. *Int. J. Logist. Res. Appl.* 21(3):300–317.

Frost & Sullivan (2021) Last-mile food & grocery delivery market revenue 2020. *Statista*. Retrieved (June 3, 2022), https://www.statista.com/statistics/1275140/last-mile-food-and-grocery-delivery-market-revenue-worldwide/.

Geha J (2022) Robot grocery delivery now available in Pleasanton. *Times-Her.*

Gendreau M, Laporte G, Musaraganyi C, Taillard ÉD (1999) A tabu search heuristic for the heterogeneous fleet vehicle routing problem. *Comput. Oper. Res.* 26(12):1153–1173.

Glovo (2017) Glovo: Pede, nós entregamos! *Glovo*. Retrieved (June 3, 2022), https://glovoapp.com/.

Goodfood (2020) Get $30 OFF - Canada's #1 Meal Kit Delivery. Retrieved (June 3, 2022), https://www.makegoodfood.ca/en/home.

Gorenflo N (2017) The Present and Future of On-Demand Work. *Shareable*.

Hackett T (2014) Why On-Demand Delivery Services Failed In The 90s. *Bus. Insid.* Retrieved (June 3, 2022), https://www.businessinsider.com/on-demand-delivery-failed-in-the-90s-2014-9.

Huang E (2019) A vision for China's parcel-delivery industry.

Instacart (2020) Instacart. *Instacart*. Retrieved (June 3, 2022), https://www.instacart.com/.

International Labour Organization (2021) ILO Data Explorer. *Unempl. Rate - ILO Model. Estim. - Annu.* Retrieved (June 3, 2022), https://www.ilo.org/shinyapps/bulkexplorer7/?lang=en&segment=indicator&id=UNE_2EAP_SEX_AGE_RT_A.

Joerss M, Neuhaus F, Schröder J (2016) *How customer demands are reshaping last-mile delivery* (McKinsey & Company).

Jonker R, Volgenant A (1987) A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing* 38(4):325–340.

Kallehauge B, Larsen J, Madsen OBG, Solomon MM (2005) Vehicle Routing Problem with Time Windows. Desaulniers G, Desrosiers J, Solomon MM, eds. *Column Gener.* (Springer-Verlag, New York), 67–98.

Kammerer W, Bimschleger C, Leddy M (2020) *Digital commerce transformation* (Deloitte).

Kuhn HW (2010) The Hungarian Method for the Assignment Problem. Jünger M, Liebling TM, Naddef D, Nemhauser GL, Pulleyblank WR, Reinelt G, Rinaldi G, Wolsey LA, eds. *50 Years Integer Program. 1958-2008*. (Springer Berlin Heidelberg, Berlin, Heidelberg), 29–47.

Labuschagne H (2022) Paying to get your Uber Eats delivery first — How it works.

Li J, Yang S, Pan W, Xu Z, Wei B (2022) Meal delivery routing optimization with order allocation strategy based on transfer stations for instant logistics services. *IET Intell. Transp. Syst.*:itr2.12206.

Liao R (2021) JD.com, Meituan and Neolix to test autonomous deliveries on Beijing public roads. *TechCrunch*.

Liao W, Zhang L, Wei Z (2020) Multi-objective green meal delivery routing problem based on a two-stage solution strategy. *J. Clean. Prod.* 258:120627.

Liu Y (2019) An optimization-driven dynamic vehicle routing algorithm for on-demand meal delivery using drones. *Comput. Oper. Res.* 111:1–20.

Lomas N (2020) Uber Eats exits seven markets, transfers one as part of competitive retooling. *TechCrunch*.

Lomas N (2021) Getir buys UK's Weezy to beef up local instant grocery biz. *TechCrunch*.

London Assembly (2020) Average Road Speed in London Boroughs.

Lyons K (2021) Nuro's self-driving robot will deliver Domino's pizza orders to customers in Houston. *The Verge*. Retrieved (June 3, 2022), https://www.theverge.com/2021/4/12/22379853/nuro-self-driving-robot-delivery-dominos-pizza-houston.

Maniezzo V, Boschetti MA, Stützle T (2021) The Generalized Assignment Problem. *Matheuristics*. EURO Advanced Tutorials on Operational Research. (Springer International Publishing, Cham), 3–33.

Martello S, Toth P (1992) Generalized assignment problems. Ibaraki T, Inagaki Y, Iwama K, Nishizeki T, Yamashita M, eds. *Algorithms Comput.* Lecture Notes in Computer Science. (Springer Berlin Heidelberg, Berlin, Heidelberg), 351–369.

Masoud N, Jayakrishnan R (2017) A real-time algorithm to solve the peer-to-peer ride-matching problem in a flexible ridesharing system. *Transp. Res. Part B Methodol.* 106:218–236.

Miller M (2021) *Online Food Delivery Report 2021* (Statista Digital Market Outlook).

Montoya-Torres JR, López Franco J, Nieto Isaza S, Felizzola Jiménez H, Herazo-Padilla N (2015) A literature review on the vehicle routing problem with multiple depots. *Comput. Ind. Eng.* 79:115–129.

Mor A, Speranza MG (2022) Vehicle routing problems over time: a survey. *Ann. Oper. Res.*

Morales DR, Romeijn HE (2005) The Generalized Assignment Problem and Extensions. Du DZ, Pardalos PM, eds. *Handb. Comb. Optim.* (Kluwer Academic Publishers, Boston), 259–311.

Munkres J (1957) Algorithms for the Assignment and Transportation Problems. *J. Soc. Ind. Appl. Math.* 5(1):32–38.

Nauss RM (2003) Solving the Generalized Assignment Problem: An Optimizing and Heuristic Approach. *Inf. J. Comput.* 15(3):249–266.

Ojeda Rios BH, Xavier EC, Miyazawa FK, Amorim P, Curcio E, Santos MJ (2021) Recent dynamic vehicle routing problems: A survey. *Comput. Ind. Eng.* 160:107604.

Öncan T (2007) A Survey of the Generalized Assignment Problem and Its Applications. *INFOR Inf. Syst. Oper. Res.* 45(3):123–141.

Osman IH (1995) Heuristics for the generalised assignment problem: simulated annealing and tabu search approaches. *Oper.-Res.-Spektrum* 17(4):211–225.

Osztovits Á, Kőszegi Á, Nagy B, Damjanovics B (2015) *Sharing or paring? Growth of the sharing economy* (PricewaterhouseCoopers (PwC)).

Outbrain (2021) Influence of delivery speed and free/discounted shipping on purchase decision in selected countries/regions worldwide in 2021. *Statista*. Retrieved (June 3, 2022), https://www.statista.com/statistics/1271199/delivery-speed-shipping-cost-influence-purchase/.

Panneerselvam B (2021) Last Mile Delivery: On-demand vs Scheduled. *Oper. Blog*. Retrieved (June 3, 2022), https://operationsblog.zorp.one/2021/07/22/on-demand-delivery-vs-scheduled-delivery/.

Pentico DW (2007) Assignment problems: A golden anniversary survey. *Eur. J. Oper. Res.* 176(2):774–793.

Petriglieri G, Ashford SJ, Wrzesniewski A (2019) Agony and Ecstasy in the Gig Economy: Cultivating Holding Environments for Precarious and Personalized Work Identities. *Adm. Sci. Q.* 64(1):124–170.

Pillac V, Gendreau M, Guéret C, Medaglia AL (2013) A review of dynamic vehicle routing problems. *Eur. J. Oper. Res.* 225(1):1–11.

Pizza Hut (2021) Em Sua Casa - Pizza Hut Portugal. Retrieved (June 3, 2022), https://www.pizzahut.pt/.

Porter J (2021) Amazon is shutting down Prime Now and folding two-hour deliveries into its main app. *The Verge*. Retrieved (June 3, 2022), https://www.theverge.com/2021/5/21/22447020/amazon-shutting-down-prime-day-delivery-app-2021.

Prevost C (2018) PICKUP: last mile delivery for large-items gets early stage funding of $5 million. *FreightWaves*. Retrieved (June 3, 2022), https://www.freightwaves.com/news/startups/technology/pickup-last-mile-delivery-for-large-items.

Project OSRM (2022) Open Source Routing Machine. *Proj. OSRM*.

Psaraftis HN, Wen M, Kontovas CA (2016) Dynamic vehicle routing problems: Three decades and counting. *Networks* 67(1):3–31.

Ram A, Jalal S, Jalal AS, Kumar M (2010) A Density Based Algorithm for Discovering Density Varied Clusters in Large Spatial Databases. *Int. J. Comput. Appl.* 3(6):1–4.

Reuters (2016) How Amazon Is Making Package Delivery Even Cheaper. *Fortune*. Retrieved (June 3, 2022), https://fortune.com/2016/02/18/amazon-flex-deliveries/.

Reyes D, Erera A, Savelsbergh M, Sahasrabudhe S, O'Neil R (2018) The Meal Delivery Routing Problem. :70.

Roser M, Ritchie H, Ortiz-Ospina E (2015) Internet. *Our World Data*.

Ross GT, Soland RM (1975) A branch and bound algorithm for the generalized assignment problem. *Math. Program.* 8(1):91–103.

Salesforce Research (2021) Most popular home delivery timelines worldwide in 2021, by selected categories. *Statista*. Retrieved (June 3, 2022), https://www.statista.com/statistics/1274439/home-delivery-timeline-worldwide-by-category/.

Savelsbergh M (1997) A Branch-and-Price Algorithm for the Generalized Assignment Problem. *Oper. Res.* 45(6):831–841.

Schrijver A (2005) On the History of Combinatorial Optimization (Till 1960). *Handb. Oper. Res. Manag. Sci.* (Elsevier), 1–68.

Simonite T (2019) How Data Helps Deliver Your Dinner On Time—and Warm. *Wired* (February 29) https://www.wired.com/story/how-data-helps-deliver-your-dinner-on-time-warm/.

Snoeck A, Winkenbach M (2022) A Discrete Simulation-Based Optimization Algorithm for the Design of Highly Responsive Last-Mile Distribution Networks. *Transp. Sci.* 56(1):201–222.

Soni D, Kumar N (2022) Machine learning techniques in emerging cloud computing integrated paradigms: A survey and taxonomy. *J. Netw. Comput. Appl.* 205:103419.

Starship (2021) https://starshipdeliveries.com/campus/. Retrieved (June 3, 2022), https://starshipdeliveries.com/campus/.

StatCounter Global Stats (2022) Desktop vs Mobile vs Tablet vs Console Market Share Worldwide. *StatCounter Glob. Stats*. Retrieved (June 3, 2022), https://gs.statcounter.com/platform-market-share.

Statista (2021) Online Food Delivery - Worldwide | Statista Market Forecast. *Statista*. Retrieved (June 3, 2022), https://www.statista.com/outlook/dmo/eservices/online-food-delivery/worldwide.

Steever Z, Karwan M, Murray C (2019) Dynamic courier routing for a food delivery service. *Comput. Oper. Res.* 107:173–188.

Strava (2020) Strava releases 2020 Year In Sport Data Report. *Strava Press*.

Swamidass P (2000) Seven “rights” of logistics. *Encycl. Prod. Manuf. Manag.* (Springer US, Boston, MA).

Swyft (2021) Swyft - Delight your customers with affordable same-day delivery. Retrieved (June 3, 2022), https://www.useswyft.com/.

Taylor TA (2018) On-Demand Service Platforms. *Manuf. Serv. Oper. Manag.* 20(4):704–720.

TomTom (2022) TomTom Traffic Index.

Toth P, Vigo D eds. (2002) *The Vehicle Routing Problem* (Society for Industrial and Applied Mathematics).

Trusk (2021) Trusk, livre tout, tout de suite. *Trusk*. Retrieved (June 3, 2022), https://trusk.com/.

Tu W, Zhao T, Zhou B, Jiang J, Xia J, Li Q (2020) OCD: Online Crowdsourced Delivery for On-Demand Food. *IEEE Internet Things J.* 7(8):6842–6854.

Uber Freight (2021) Uber Freight. *Uber Freight*. Retrieved (June 3, 2022), https://www.uber.com/us/en/freight/shipper/faq/.

Ulmer MW, Streng S (2019) Same-Day delivery with pickup stations and autonomous vehicles. *Comput. Oper. Res.* 108:1–19.

Ulmer MW, Thomas BW (2018) Same-day delivery with heterogeneous fleets of drones and vehicles. *Networks* 72(4):475–505.

Ulmer MW, Thomas BW, Campbell AM, Woyak N (2021) The Restaurant Meal Delivery Problem: Dynamic Pickup and Delivery with Deadlines and Random Ready Times. *Transp. Sci.* 55(1):75–100.

Voccia SA, Campbell AM, Thomas BW (2019) The Same-Day Delivery Problem for Online Purchases. *Transp. Sci.* 53(1):167–184.

Wang K, Zhou Y, Zhang L (2022) A Workload-Balancing Order Dispatch Scheme for O2O Food Delivery with Order Splitting Choice. *J. Theor. Appl. Electron. Commer. Res.* 17(1):295–312.

Yildiz B, Savelsbergh M (2019) Provably High-Quality Solutions for the Meal Delivery Routing Problem. *Transp. Sci.* 53(5):1372–1388.

Zhang J, Hu M (2020) Meituan, Ele.me face backlash over risky conditions for food couriers. *South China Morning Post*. Retrieved (June 3, 2022), https://www.scmp.com/tech/apps-social/article/3100847/eleme-give-customers-option-wait-longer-food-deliveries-after.

Zhou W, Lin J (2019) An On-Demand Same-Day Delivery Service Using Direct Peer-to-Peer Transshipment Strategies. *Netw. Spat. Econ.* 19(2):409–443.

Zipline (2019) Zipline - Instant Logistics. Retrieved (June 3, 2022), https://flyzipline.com/.