LUISS

Degree Program in Data Science and Management

Course of International Operations and Global Supply Chain

From Data to Delivery: Data-Driven Engineering of Last-Mile Optimization for Quick Commerce in Rome and Montreal

Prof. Lorenza Morandini

SUPERVISOR

Dr. Giovanni Valentini

CO-SUPERVISOR

ID 789331

CANDIDATE

Academic Year 2024/2025

Abstract

This thesis explores the operational optimization of last-mile deliveries for quick commerce (Q-commerce) platforms, focusing on courier assignment strategies in the distinct urban environments of Montreal and Rome. Q-commerce has reshaped consumer expectations through the rapid delivery of essential goods, bringing both new logistical opportunities and acute challenges, particularly in the last-mile segment of the last mile. Recognizing the impact of assignment logic on delivery efficiency, this work develops a modular simulation framework that enables rigorous, scenario-based benchmarking of both classical optimization and interpretable machine learning approaches.

Four assignment strategies are evaluated: a naive greedy baseline, the Hungarian algorithm for minimum-cost matching, a standalone Adaptive Neuro-Fuzzy Inference System (ANFIS), and a novel hybrid pairwise method that integrates ANFIS predictions within combinatorial optimization. Each method is tested under both realistic and geometric routing assumptions, with further analysis of batching-enabled dispatch using context-aware synthetic data tailored to each city.

Empirical results show that classical combinatorial optimization, exemplified by the Hungarian algorithm, consistently achieves the lowest delivery latency and operational cost under high-fidelity routing. Machine learning and batching-enabled strategies, especially the hybrid pairwise approach, offer incremental gains in efficiency and workload fairness in geometric and batching scenarios, though these benefits are context-dependent and sensitive to input features. The cross-city analysis reveals that infrastructure, regulatory context, and compensation models significantly shape both absolute and relative assignment performance.

Overall, the findings affirm the reliability of classical optimization for last-mile logistics, while highlighting the evolving potential, and current constraints, of data-driven and neuro-fuzzy assignment strategies. This thesis provides an empirical foundation for developing adaptive, context-sensitive assignment systems in urban logistics, emphasizing the need to align technical innovation with real-world operational realities.

Acknowledgements

I would like to express my sincere gratitude to Professor Lorenza Morandini for her invaluable guidance, thoughtful advice, and constant encouragement throughout the development of this thesis. Her time, expertise, and insightful inputs have been instrumental in shaping both the direction and the quality of this work. It has been a genuine pleasure and privilege to work under her supervision.

Contents

1 Introduction

1.1 Context & Motivation

The history of commerce is inherently intertwined with that of humanity. The earliest form of commerce was built on the barter system, originating from the basic need to exchange for goods and services. This system was eventually revolutionized through the invention of money, whereby said goods and services were exchanged for currency. This system continues to be practiced today, even though the currency itself has radically changed over the following millennia, i.e., from gold and silver coins to fiat currency and credit cards.

The creation and widespread dispersion of the internet further revolutionized commerce, as goods could now be purchased online. Online shopping first became accessible to the public in 1991 [UT Permian Basin, 2022], with companies such as Amazon and E-Bay coming to fruition. This type of shopping done via the internet was later dubbed E-commerce, and as of today, is one of the largest markets, boasting revenues of US $4.32tn in 2025, as well as one of the fastest growing with CAGR of 8.02% [Statista, 2025b].

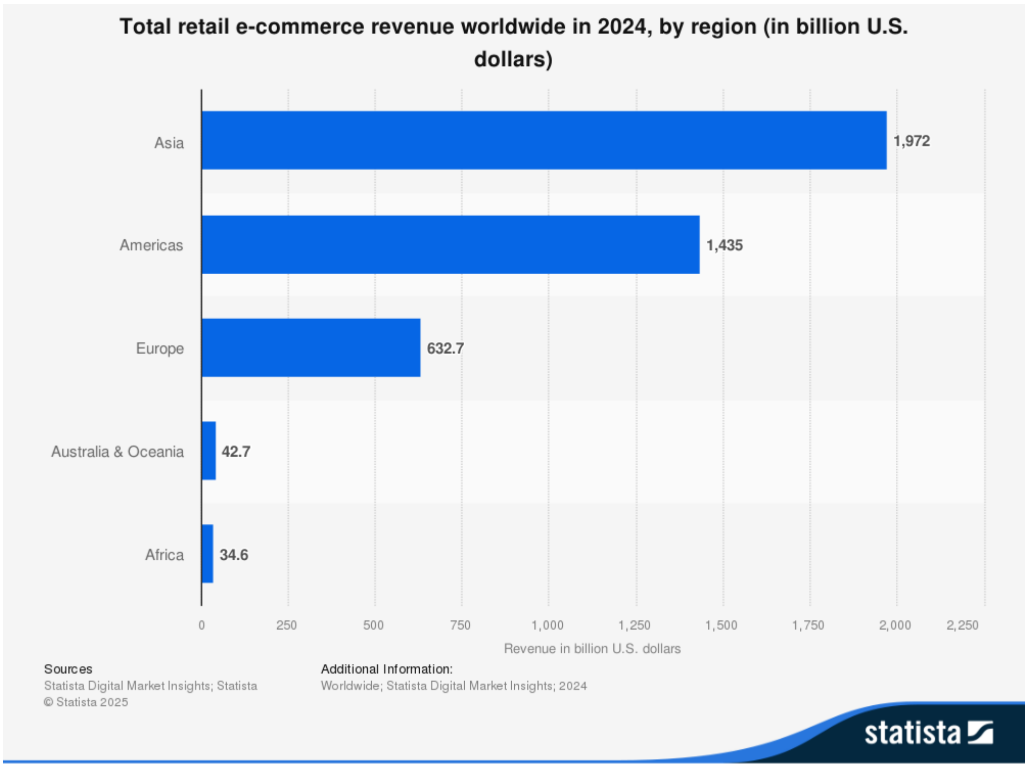


Figure 1: Total retail e-commerce revenue worldwide in 2024, by region (in billion U.S. dollars). Source: Statista Digital Market Insights, 2025.

However, in recent years, a specific type of e-commerce has rapidly grown in popularity. As consumers became accustomed to the convenience and speed of online shopping, the expectations of said consumers consequently evolved rapidly in tandem with the drastic technological advancements that have occurred since e-commerce's inception. This type of e-commerce, called Quick Commerce, as well as q-commerce or q-com for short, can be defined as online shopping characterized by the extremely fast delivery of goods, normally in under 30 minutes [Berger, 2022].

Order fulfillment is handled by a network of small warehouses called dark stores, i.e., private micro-fulfillment centers optimized for the picking, packaging, and dispatching of items, or by traditional stores, dependent on the business model utilized by the q-commerce company in question. These aforementioned micro-fulfillment centers (MFCs) are compact, often automated storage facilities located in urban areas to enable rapid picking and dispatch of goods, usually within 30 minutes or less of order placement. Furthermore, they usually have a limited product range, focusing on frequently purchased items. The delivery of said items is then handled by the q-commerce company's fleet of drivers/riders. The fleet is typically composed of a variety of delivery vehicles, ranging from regular bikes to motorbikes and cars, and the driver is normally in proximity to both the pick-up point as well as the client in order to minimize delivery time.

Given that these companies' main activity revolves around the rapid delivery of goods, their operational success is largely determined by how efficiently they manage last-mile logistics, i.e., the final, and most expensive, segment of the supply chain. Last-mile delivery alone can account for over half of total shipping costs, driven by the challenges of fragmented demand, urban congestion, and the need for speed and reliability. Within this context, the assignment of orders to couriers is a critical decision point. Even small improvements in assignment algorithms can translate directly into lower costs, faster deliveries, and a more resilient logistics network.

Therefore, the primary goal of this thesis is to advance the state of courier assignment in quick commerce platforms by developing, benchmarking, and critically evaluating a range of assignment algorithms, from classical combinatorial methods to modern interpretable machine learning approaches. Through rigorous simulation in urban environments, the thesis not only seeks to assess the generalizability and robustness of these assignment strategies, but also to examine how the interplay between algorithm and local context, such as regulatory environments and compensation structures, shapes operational efficiency and cost. By applying and comparing these strategies in two contrasting urban contexts, Montreal and Rome, the work highlights both the flexibility of the proposed framework and the specific conditions under which different assignment models prove most effective.

Characteristic	Quick Commerce	Restaurant Delivery	Traditional Grocery Delivery
Delivery Speed	10–30 mins	20–45 mins	Same day to 2+ days
Typical Basket Size	Small	Single meal	Large (weekly shop)
Product Range	Essential goods (non-prepared)	Prepared food (cooked meals)	Full grocery selection
Fulfillment Source	Dark stores / Partner stores	Restaurants	Supermarkets / Aggregators
Platform Type	Dedicated q-com apps (e.g., Gopuff, Glovo)	Food delivery apps (e.g., Uber Eats, DoorDash)	Retail or third-party (e.g., Instacart, Carrefour)
Primary Use Case	Urgent / spontaneous needs	Meals / convenience dining	Planned grocery shopping
Delivery Model	Gig couriers (bike/scooter/car)	Gig couriers (bike/scooter/car)	Store staff or delivery partners
Order Planning	Impulsive	Impulsive	Planned

Table 1: Characteristics of Quick Commerce vs. Restaurant Delivery & Traditional Grocery Delivery

1.2 Defining Quick Commerce

1.2.1 History of Quick Commerce

Whilst quick commerce's rapid ascension in popularity may be rather recent, it is by no means a new concept. For instance, Kozmo.com was a venture capital funded, online company, founded in 1998 during the dot-com boom, which promised the delivery of DVDs, video games, and even Starbucks coffee in under an hour in cities across America; however, the company's dedication to making such rapid deliveries led them to spending as much as US $30 million in operation a month, thus leading them to cease operations in 2001 [Blair, 2001].

Since then, many companies have learnt from Kozmo.com's mistakes and emerged in this niche space, notably in the early 2010s, many of which have gone on to become prominent players in today's market. These players include Uber Eats, Glovo, DoorDash, Deliveroo, Foodora, Delivery Hero, and Gopuff amongst many others. Furthermore, the covid pandemic caused the popularity of such apps to spike drastically, causing a large influx of investments into these companies, as well as the creation of many others. In fact, funding in online food delivery companies totaled over US $19.1 billion worldwide in 2021 [Statista, 2023c]; however, post-pandemic economic slowdown affected the survival of many of these newly founded players, as well as already established players. These economic woes, coupled with the fact that venture capital investment had drastically diminished, led to the defunctness of many of these newly founded companies. Conversely, some of these companies were acquired by larger players, exemplified by Gorillas, a German ultrafast delivery company founded in 2020 using a large network of dark stores, being acquired by Turkish giant Getir for US $1.2 billion in 2022 [Tuncay and Ersen, 2022]. That being said, even large players suffered losses, with companies such as Gopuff having to reduce their workforce by more than 10% in the same year [NielsenIQ, 2024].

Figure 2: Logos of prominent last-mile delivery and quick commerce platforms worldwide. These companies, ranging from established giants to regional specialists, have shaped the evolution of urban delivery ecosystems.

It is important to note that while companies like Uber Eats, DoorDash, and Glovo are often associated with restaurant food delivery, they have increasingly expanded into the quick commerce domain by offering groceries, convenience goods, and other non-prepared items for rapid delivery. Therefore, while the origins of these companies lie in the food delivery sector, their evolution and current operations make them central actors in the q-commerce space. For the purposes of this thesis, the term ”quick commerce” refers primarily to the rapid delivery of everyday goods, although overlaps with restaurant delivery platforms will occasionally be referenced due to shared infrastructure and market players. While restaurant delivery shares many logistical similarities with q-commerce, it is typically excluded from its definition because it involves the preparation of freshly cooked meals, whereas q-commerce is centered on the rapid delivery of pre-packaged or ready-to-use consumer goods.

Moreover, despite sharing some surface similarities with traditional delivery models for groceries, quick commerce distinguishes itself through its operational intensity, customer expectations, and logistical structure. Unlike traditional delivery services, which often revolve around scheduled fulfillment, larger order volumes, and longer lead times, q-commerce is designed for ultra-rapid, small-basket fulfillment, often within 30 minutes of order placement. This time-sensitive nature requires a fundamentally different logistics network, relying on dense urban micro-fulfillment infrastructure, real-time courier allocation, and predictive demand forecasting. The impulsive, convenience-driven nature of q-commerce demands extreme agility and precision, thus making last-mile optimization in this space a uniquely complex and pressing challenge.

1.2.2 Current Market Trends

Today’s market is dominated by five main companies. The two largest are Uber Eats and Delivery Hero, both boasting worldwide revenues of US $12.2 billion and US $10.97 billion respectively in 2023 [Statista, 2023c]. These two are then followed by DoorDash, Just Eat, and Deliveroo. When analyzing the bigger picture, the quick commerce market is projected to reach $195.01 billion in 2025, with a CAGR of 8.05%, thus suggesting that the projected market volume will attain that of US $265.84 billion by 2029. The largest market as of right now is China, which accounts for nearly half of all revenues generated worldwide, i.e., US $92.68 billion, followed by the United States of America, with revenues of US $62.63 billion [Statista, 2025c].

These trends are consistent with the market penetration rate, where quick-commerce companies have penetrated approximately 23.9% of the market in China, followed closely by Singapore and USA, with a market penetration rate of 17.7% and 17.5% respectively. Worldwide, the user penetration rate is 8.6%, meaning that the services of these companies is used by 675.6 million people, and is expected to grow to 11.0% in 2029, suggesting that total number of users will approximately be 886.8 million people [Statista, 2025c]).

The total annual revenue per user has also increased with respect to the number of users, with the average annual revenue per user (AARPU) increasing from US $148.09 in 2017, to US $288.63 [Statista, 2025c]). This increase in *AARPU* is not only indicative of the growth in popularity of quick commerce, but also of the growth of the average basket. In Europe, the average order basket was approximated to be around 20 euros (Roland Berger), but when asked to comment on this during an episode of The Retail Reality Show podcast in 2024, Senior Vice-President of Quick Commerce at Delivery Hero Milena Lazarevska stated that the average for their company had risen to a range of 27-30 euros per basket [Show, 2024].

Lazarevska also went on to discuss to discuss the target clientele. She stated that most users were young adults, with the upper age limit being 55, and that there was an even mix of both men and women. Furthermore, Lazarevska

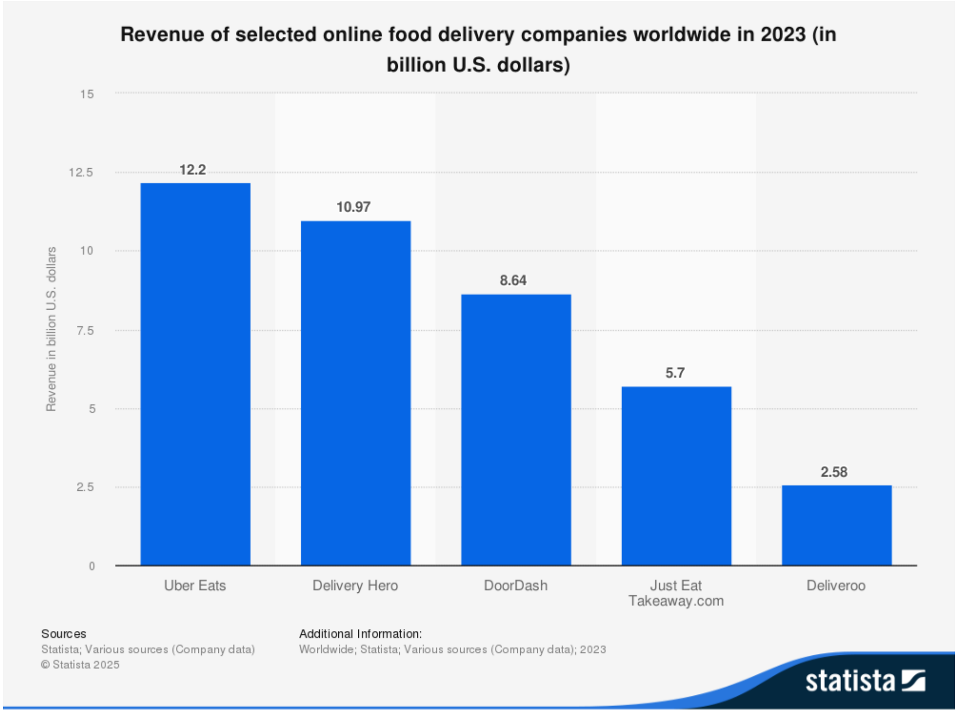


Figure 3: Revenue of selected online food delivery companies worldwide in 2023 (in billion U.S. dollars). Source: Statista, Company data, 2025.

also commented that the most valuable type of customer was not one who infrequently bought large baskets, but a repeat customer who would do smaller baskets, as well as those who purchased fresh produce.

When analyzing the underlying macroeconomic drivers behind the growth of quick commerce, a number of factors come into play, including the rise in popularity of smartphones, the shift in consumer expectations and demands for ultra-fast service and convenient deliveries, as well as the general growth in e-commerce. Furthermore, the covid pandemic also massively contributed to its growth, as companies increasingly went digital to help mitigate the economic downturn caused by the closure of brick-and-mortar stores [UNCTAD, 2021]; however, not only did covid help grow the quick commerce market, but it also caused a significant shift in the market with respect to the operations of companies. As more businesses partnered with quick commerce platforms, they became increasingly omnichannel and expanded across more verticals. These companies began delivering a wide range of products, from cosmetics and alcohol to other daily necessities typically found in pharmacies or convenience stores.

1.2.3 Quick Commerce Business Models

There are two types of business models that a quick commerce company may use, both having different profit models. The first is through partnerships with local retailers in a marketplace model, where the companies earn a commission on each order placed. This specific type of business model allows for faster entry into the market, as there are less sunk costs given that the inventory and warehousing is handled by third parties. The second model relies on dark stores, which, as previously mentioned, are small, strategically located fulfillment centers owned by the company itself. In this case, companies function like retailers, purchasing goods wholesale and profiting from the margins between the buying and selling price. The key advantage of this model lies in greater control over inventory, pricing, and service speed, though it comes with higher operational costs.

Figure 4: Interior view of a dark store used for quick commerce operations. Dark stores are specialized micro-fulfillment centers optimized for rapid picking, packing, and dispatching of goods, enabling ultrafast delivery in urban areas.

In both cases, companies typically charge their clients delivery fees, which can vary based on order size, delivery distance, and service level (e.g., express delivery). Additionally, quick commerce companies have also recently begun offering subscription models, thus negating any delivery fees, and adding another revenue stream for said platforms. Lastly, companies in this industry also profit from advertisements, as brands pay for premium placements on the platform, targeted promotions, or featured listings to boost visibility and drive sales.

Certain companies have recently begun adopting a hybrid approach however, meaning that they combine both the dark store model with the marketplace one with the goal of maximizing geographical reach and operational efficiency. During the podcast, Lazarevska said that this was the most successful model, as they were platforms which combined multiple different verticals, thus best satisfying the demands of their clientele. She also went on to say that the companies that didn't survive post-pandemic were the stand-alone dark stores, as they had the highest capital expenditures.

The logistics and requirements for the two base models differ slightly. As aforementioned, the marketplace models require significantly less capital investment as the inventory and warehousing is handled by third parties, while the investment level is much higher for the dark store model due to the costs associated with location (e.g., rent, electricity, water, staff, etc.) in addition to the costs of being a retailer. Furthermore, while both models thrive in large cities and metropolises, i.e., densely populated areas, the population requirement for a marketplace model to be successful is significantly smaller than that of the dark store model. Lazarevska stated that the minimum population size for a dark store model would be approximately 300,000, while the marketplace model would succeed "as long as there is a supermarket" [Show, 2024].

1.3 Comparative Market Analysis

To effectively optimize last-mile deliveries for quick commerce in North America and Europe, specifically within the context of Montreal, Canada and Rome, Italy respectively, it is essential to first examine the defining characteristics of each market and city, thereby identifying their differences and commonalities. Additionally, although these two cities may not be the quick commerce capitals of their respective continents, they do offer contrasting urban environments and market structures, and ultimately were chosen due to the personal connection that I have with them, as I grew up in Montreal and decided to continue my academic career in Rome.

1.3.1 Market Maturity and Key Players

Although quick commerce has gained significant traction globally, the development of its markets and the penetration of the players within said markets varies heavily by region based on socioeconomic factors, as well as consumer behaviors, and regulation. In this context, both Rome and Montreal represent urban environments where the q-com market is present but still evolving, with each city occupying a distinct position within its regional landscape.

In Italy, the quick commerce market has seen some steady growth in recent years, though it remains relatively within the early stages of when comparing it to the rest of Europe. In particular, the quick commerce market lags far behind the ones found in Western Europe, with the likes of England, France, and Germany emerging as somewhat of a hotspot for q-com companies during the covid-19 pandemic. In fact, a 2021 Euromonitor report stated that over 30 companies were actively competing within the market, "most of which [were] established within the past [year]" [Bogdanova, 2021].

Although a promising environment for q-com companies, mainly due to the growth of the sharing economy and the increasing digitization of payments, the Italian market has proved rather tricky for these companies with giants such as Uber Eats divesting operations in June of 2023 [Mattia, 2023], and start-ups such as Gorillas opting to leave Italy in 2022 [Ardoni, 2022], citing lack of scalability and the dominance of other platforms as their reasons for exiting. Today, the Italian market is dominated by three players, with the most popular platform being Glovo, followed by Deliveroo and Just Eat. These platforms better adapted to Italian preferences than the previous two, offering a broader mix of services that were better suited to the local customer base. Today, the Italian quick commerce market is estimated to generate over US $862 million in 2025, and is expected to continue to grow to over US $1 billion by 2028 [Statista, 2025c].

The Italian market shares striking similarities to the Canadian one. Similarly to comparing the Italian market to markets in other European countries, the Canadian market is also lagging behind its continental neighbors, i.e., the USA; however, with the rise in mobile technology, and with the changes in demands and expectations from consumers, the Canadian market itself has also experienced some significant growth over the past few years. In fact, a study conducted by Dalhousie University's Agri-Food Analytics lab found that "the number of Canadians who ordered groceries online at least once a month doubled during the pandemic" [Research, 2024], and that this behaviour had continued post-pandemic, suggesting a promising future for q-commerce within the Canadian market.

In 2024, the most popular grocery delivery applications was Instacart with 775,000 downloads, followed by Flashfood with 383,000 downloads, nearly half of what Instacart's downloads were [Statista, 2024c]. Consumer reliance on food delivery apps was also high, seeing that online surveys from 2024 found that nearly 54% of Canadians reported having used Uber Eats within the last 12 months, followed by 49% for DoorDash, and 41% for SkipTheDishes [Statista, 2024b]. While originally specialized in food delivery from restaurants, many of these platforms have expanded to now also specialize in groceries, ready-to-eat meals, and convenience store items. Furthermore, to simulate their respective growth and to broaden their reach, many of these platforms formed strategic collaborations with local grocers. For instance, Instacart partnered up with numerous local retailers, as well as the top 5 national chains, including Costco, Loblaws, and Walmart [Instacart, 2022]. Moreover, both Instacart and Uber Eats announced a partnership with Canadian food conglomerate Empire, thus adding nearly 250 grocery stores to their respective platforms, including the likes of Sobeys, FreshCo, and Farm Boy [MacConchie, 2024].

Today, the Canadian market is estimated to generate US $1.3 billion in revenue in 2025, and is poised for continued growth and development, with estimates suggesting that it will reach nearly US $1.9 billion in revenue by 2029 [Statista, 2025c].

While both countries exhibit similarities within their respective quick commerce markets, most notably with both of them being still in their early stages and with a vast potential for growth, where these markets exhibit quite polarizing differences is with respect to the consumer behaviour within each market, as well as within each cities' infrastructure and regulation, subjects which will be further discussed within the following sections.

1.3.2 Consumer Behaviour and Cultural Dynamics

Consumer behaviours and cultural dynamics play a key role in the success of any company who attempts to do business in any region of the world. They are often the reason that certain companies are very successful in their home nation but fail to replicate that success when expanding to foreign nations, no matter how close the countries may be, or how similar their cultures are perceived to be. For example, these two factors are some of the main reasons that Walmart, the world's largest retailer, failed in Germany, as they failed to understand the nuances of the German market, as well as the preferences of the local customer base [Saini, 2019].

In Italy, food consumption habits are deeply tied to local traditions, with a culture that particularly emphasizes fresh, and seasonal ingredients. The Italian consumers have strong preferences for premium, local products, making it difficult for international competitors. Furthermore, consumers prefer taking frequent trips to the grocery store, purchasing only small baskets at a time, with many people even preferring to go to their local food markets to obtain fresh produce daily. While this somewhat goes against the values of quick commerce, given that larger baskets are more profitable for companies working in this space, and that the Italian public tend to prioritize the freshness of produce over the convenience of obtaining, it can also align with these values as quick commerce is all about maximizing convenience and minimizing friction. In fact, a survey produced by Statista in 2024 showed that 28% of Italian consumers would use quick delivery services for fresh fruits and vegetables, and that 16% of them would use it for non-perishable groceries [YouGov, 2024], indicating that there is a growing demand for immediacy of food, particularly if the quality of the produce remains the same.

Additionally, even though the number "of [Italian] consumers who say they do all or most of their grocery shopping online" has grown exponentially since the pandemic [Mintel, 2023], Italians are still rather price sensitive, in large part due to the growing inflation and to the decrease in the population belonging to the middle class [Research, 2021], thus prompting them to find more ways to save more money. These trends also reduce the convenience proposition of quick commerce, as prices tend to be slightly higher due to the delivery fee; however, there are still positive signs within the market as consumer confidence rose by 4.6 points in 2023, and food-to-go spending (e.g. prepackaged meals, ready-to-heat convenience foods, takeaways), a highly correlated food vertical to quick commerce, grew 14.6% year-over-year between 2022 and 2023, therefore signaling a growing desire for speed and convenience in meal-related purchases [Company, 2025].

Lastly, beyond price and purchasing habits, payment preferences and demographic factors also shape the viability of quick commerce in Italy. Despite growing digitalization, Italy continues to be a heavily cash-based society, with nearly a quarter of all purchases done via cash [Stogdon, 2024]. This, coupled with the fact that Italy is the oldest country in the European Union, having a median age of above 48 and nearly 24% of the population being above 65 [Carbonaro, 2024], present unique challenges for quick commerce. Older consumers, who account for a significant portion of national spending, are often less inclined to adopt mobile-first platforms or trust digital payment systems. As such, it is in the best interest of platforms to also offer a cash-on-delivery (COD) payment option, as can already be seen by quick commerce companies such as Deliveroo, in order to gain better access to a broader clientele.

Canadian consumer behaviour, on the other hand, is heavily influenced by digital convenience, multicultural diversity, and broader acceptance of technology-based services and cashless payments. When analyzing their food shopping habits, Canadians prefer going to supermarkets where they can do one-stop shops, with over 44% doing their groceries only once a week [Statista, 2023a]. Only 3% of Canadians do their shopping daily [Statista, 2023a], which aligns better with the profitability model of quick commerce; however, despite growth in the online food shopping segment, most of which done through Instacart as previously mentioned, only 14% of the population appear to regularly purchase food and products from online channels [Statista, 2024a]. Sylvain Charlebois, professor and director of the Agri-Food Analytics Lab at Dalhousie University in Halifax, echoed this statement by saying that Canada still lags far behind the United States with respect to online food shopping, where nearly 50% of consumers engage in online shopping [Consumers Council of Canada, 2024].

Similarly to their Italian counterparts, Canadian spending behaviour has been influenced by increasing prices caused by inflation. Conversely, while Italian consumers value freshness and localness, Canadians tend to factor pricing as the number one reason for ranking grocery stores [Statista, 2023b]. In spite of that, there has been an increase in environmental consciousness amongst shoppers, particularly with the younger generation, with close to 38% saying that they would pay more for a product with lower climate impact, despite the raising prices[Statista, 2024a].

Furthermore, Toronto retail analyst Bruce Winder stated that quick commerce platforms are most notably used by Generation Z and Millennials, rather than by Boomers. He also stated that online shoppers tend to be the younger demographic, "particularly those who are affluent with disposable income, busy and pressed for time, tech-savvy and value convenience" [Consumers Council of Canada, 2024]. In fact, a news report in 2022 by credit union Vancity found that gen-z and millennials spent more than nine times on food delivery than boomers on food delivery.

Ultimately, it is clear that both Italian and Canadian consumers are interested in the development of quick commerce platforms, especially as the younger, more technically savvy generations become a larger proponent of their respective country's national spending. While Italian consumers continue to favour fresh, local produce and maintain a stronger-reliance on cash and in-person shopping experiences, Canadian consumers lean more towards convenience, both in-person and digital, through one-stop supermarkets and online deliveries. Nevertheless, both populations have experienced a behavioural shift since the pandemic, that, while slightly diminished from their peak, are indicative of long-term change. An increasing desire for time-saving solutions, responsible consumption, and positive app-based service experiences suggests that quick commerce holds long-term potential across both markets; however, the success of these platforms will depend on how said platforms adopt to the socio-economic, generational, and behavioural realties of each context. As such, the following section will take a more granular look at the how quick commerce plays out at the city level, where factors such as urban environment, infrastructure, and regulation affect the profitability of players situated in both Rome and Montreal.

1.3.3 Urban Environment, Infrastructure, and Regulation

The operational performance of quick commerce platforms is closely linked to both the physical layout and regulatory structure of the cities in which they operate. When comparing Rome and Montreal, stark differences emerge in terms of urban design, transportation infrastructure, regulatory environments, and even municipal sustainability policies, all of which can impact the efficiency and profitability of said platforms. As such, this section will focus on the local realities of both cities that create constraints and opportunities for their respective quick commerce markets.

In Rome, the realities of operating a quick commerce business are heavily influenced by the city's historical layout, and by its regulatory constraints. The urban core is composed of narrow, irregular streets as well as densely packed neighborhoods. Furthermore, the widespread use of the *Zona a Traffico Limitato* (ZTL) which restricts vehicular access to the historic center during certain times of the day, thus requiring delivery vehicles to obtain permits and navigate around time-sensitive restrictions. These urban features, designed to reduce congestion and consequently emissions, pose difficulties for route optimization and timely deliveries, both critical to quick commerce profitability models; however, Rome's mild climate enables the year-round use of scooters and bicycles to bypass traffic and service restricted areas. Although Rome's mobility infrastructure is sorely underdeveloped, most particularly denoted by the lack of protected bike lanes, as part of their operations strategy and as part of their commitment to sustainability, platforms such as Glovo continue to promote the use of e-bikes to their riders[Glovo, 2023], something only enabled by the warmer climate.

Rome's regulatory environment further complicates quick commerce operations, as not only must platforms comply with local traffic regulations, but also with national labour policies and European Union directives. In fact, Italy had begun tightening its regulation of the gig economy in recent years, with a notable example of this coming in 2021, where Milan prosecutors and the Italian Labour Inspectorate reclassified 60,000 couriers as employees rather than independent contractors, and thus ordered the four main food delivery platforms at the time, i.e., Uber Eats, Deliveroo, Glovo, and a subdivision of the Glovo group called Foodinho, to pay over €733 million (equivalent to US $889 million at the time) in fines for noncompliance with labor laws [Parodi, 2021]. Consequently, these new legal

Figure 5: Map of the Rome ZTL (Zona a Traffico Limitato) area. The green boundary highlights the central restricted traffic zone, a key operational constraint for last-mile delivery in the historical center. Base map: OpenStreetMap contributors, CC-BY-SA.

changes brought about new obligations for quick commerce platforms in terms of employment benefits, insurance, and safety protocols, all of which increase operational costs for said platforms, and further complicate the use of on-demand, flexible labor models.

In contrast, Montreal presents itself as a more modern, and flexible city in which to conduct quick commerce operations. Its grid-based layout and wider roads are more conducive to efficient route planning and vehicular access; however, while the central neighborhoods are densely populated, most of the city's population resides in the suburbs, thus making full-area coverage costlier and less efficient. Furthermore, Montreal benefits from a vast network of protected bike lanes, totalling over 1065 km [Ville de Montréal, 2024], closely aligning with the sustainability values of quick commerce platforms; however, bikes are rarely used in operations, partly due to the aforementioned suburban sprawl of the population, and in large part due to the weather. Given Montreal's long winter, marked by lots of snow and ice, delivery fleets in Montreal are largely composed of couriers' personal vehicles, thus increasing operational complexity and carbon emissions.

While labour regulations have historically been more permissive in Montreal than in Rome, particularly regarding the classification of gig workers, who were considered independent contractors, thus giving platforms more flexibility with regards to workforce management, recent undergoing legislative developments suggest that the Canadian regulatory environment is evolving. In June 2024, the Canadian federal government passed amendments to the Canada Labour Code that strengthen protections for gig workers in federally regulated industries, including courier services. The legislation introduces a presumption of employee status, meaning that workers must now be treated as employees unless the platform can prove otherwise, thus marking a significant shift in legal interpretation and enforcement [Government of Canada, 2023]. These changes currently only apply to federally regulated industries (e.g., transportation, banking, telecommunications, etc.), while gig work under Uber Eats and Skip the Dishes currently falls under provincial jurisdiction, it does signal a broader move towards increasing protection for gig workers in Canada, ultimately meaning that quick commerce platforms may soon face increased pressures to adapt to the same standards.

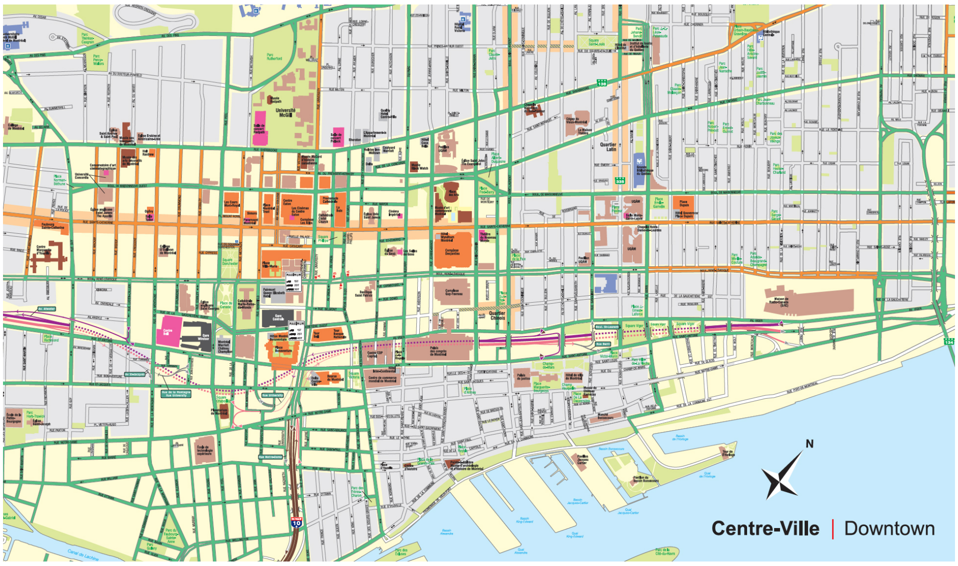


Figure 6: Downtown Montreal street layout (Centre-Ville). The grid-based design and wide avenues contrast with the more irregular street patterns of Rome, impacting last-mile delivery strategies and route optimization.

Additionally, regulator differences also extend to product availability. In Quebec, the sale and distribution of alcohol and cannabis is tightly regulated by the provincial government. The *Société des alcools du Québec* (SAQ) holds a monopoly on the distribution of alcohol, thus limiting the ability for quick commerce platforms to offer such products. On the other hand, Lazarevska had stated on the podcast that beer was one of the most commonly ordered items in Europe (The Retail Reality Show, 2024), highlighting a key distinction in how regulatory frameworks can either enable or constrain product diversification within quick commerce. Moreover, cannabis, although illegal in Italy, is regulated in at the provincial level, and is even able for delivery on platforms such as Uber Eats in other provinces such as Ontario, yet currently prohibited in Quebec.

Montreal's bilingual identity adds an additional operational cost for quick commerce platforms, as said platforms must operate in both French and English to comply with provincial regulations, i.e., Quebec's Charter of the French Language and Bill 96, and to meet customer expectations. Although necessary for full market integration, this impacts everything from app design to customer support, legal compliance, and even marketing strategy, ultimately increasing costs for the platforms.

Together, these local factors highlight the importance of geographic and regulatory context in shaping the delivery models, cost structures, and profitability of quick commerce platforms. While Rome and Montreal both offer exciting growth opportunities for players in this space, these cities also demand tailored strategies for success that respond to not only infrastructure and labor dynamics, but also to the culture and climate of the local area.

Category	**Rome**	**Montreal**
Market Maturity	Emerging, lagging behind Western Europe	Growing, lagging behind the U.S.

Category	Rome	Montreal
Key Players	Glovo, Deliveroo, Just Eat	Instacart, Uber Eats, DoorDash, SkipTheDishes
Consumer Behavior	Traditional habits, strong emphasis on fresh/local products	Convenience-oriented, digital familiarity
Grocery Shopping Frequency	Frequent, small trips; daily markets	Weekly or bi-weekly bulk trips
Digital Payment Adoption	Lower adoption; cash still prevalent ($\sim$25%)	High adoption of digital and cashless payments
Price Sensitivity	High, due to inflation and shrinking middle class	Moderate, influenced by inflation
Cultural Food Preferences	Freshness prioritized over convenience	Price and convenience over localness
Urban Layout	Historic, narrow streets; ZTL zones	Grid-based; wide roads; suburban sprawl
Mobility Infrastructure	Limited bike lanes; underdeveloped	Extensive bike lanes (1,065 km)
Vehicle Usage	Scooters, bikes (weather-permitting)	Personal cars dominate in winter
Regulation (Labor)	Stricter; platforms fined, and workers reclassified	Currently flexible; federal changes incoming
Regulation (Product Availability)	Alcohol allowed; cannabis illegal	SAQ monopoly on alcohol; cannabis delivery banned in QC
Climate Impact on Ops	Mild; enables year-round bike/scooter usage	Long, snowy winters; limits bike delivery
Language Requirements	None	Must operate in French and English

Table 2: Quick Commerce Characteristics: Rome vs Montreal

1.3.4 Defining the Last-Mile in Quick-Commerce

Last-mile delivery is a logistics term which refers to the final stage of the delivery process, where a product is moved from a transportation hub to its final destination [DHL, 2023]. In the context of quick commerce, it refers to when the products are moved from the dark store, or the retail outlet, to the consumers front door. Although it represents the smallest geographical portion of the supply chain, the last mile is often the most complex, most polluting, and most cost-intensive stage of the entire delivery cycle, with estimates suggesting that last-mile deliveries account for 53% of the total cost of shipping, and 41% of the total supply chain costs (Accenture, 2021).

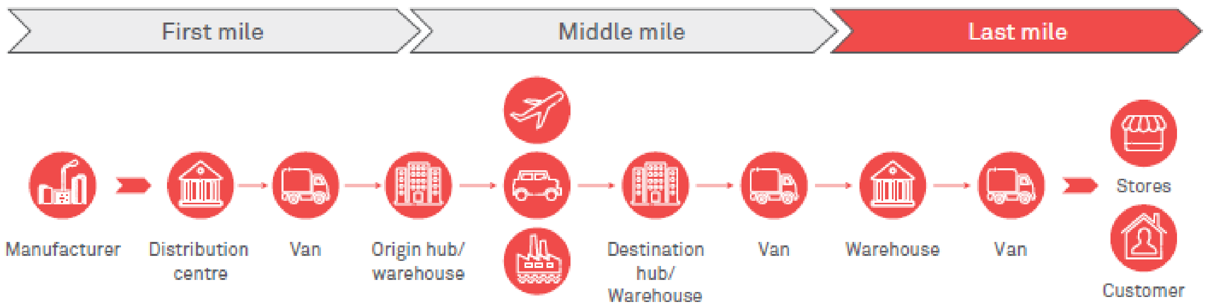


Figure 7: Illustration of the first, middle, and last mile in supply chain logistics. The last mile refers to the final delivery stage from a warehouse or distribution hub to the customer, representing the most complex and costly segment of the process.

This disproportionate cost stems in large part due to the fact that last-mile deliveries are highly fragmented due to their time sensitive nature. This prevents quick commerce platforms from fully leveraging economies of scale, given that deliveries are often made individually or in small batches. Each delivery must be managed separately, with limited opportunities to consolidate orders due to the small delivery windows, thus worsening the unit economics of each transaction. Moreover, continued usage of motorized vehicles in dense urban areas contributes significantly to congestion and therefore carbon emissions, making the last mile not only the most expensive segment of the supply chain, but also the most environmentally taxing [Accenture, 2021].

What distinguishes last-mile delivery in the quick commerce sector from traditional retail logistics is the necessity for both ultrafast fulfillment and high spatial accuracy, all within dense urban networks. Unlike conventional delivery systems that optimize for bulk delivery or route efficiency, q-commerce platforms must prioritize speed and real-time adaptability, often under unpredictable demand patterns and with limited planning time. Fulfillment must occur in minutes rather than days, and delivery must reach the exact front door of the client rather than a large warehouse. The result is a logistical environment that must continuously balance speed, cost, and service quality in a way that is both dynamic and highly context-sensitive.

These limitations are further amplified by the unique operational conditions found in individual cities. In dense and congested cities like Rome, logistical constraints are driven by historical infrastructure, narrow roadways, and regulatory vehicle access zones. In contrast, a city like Montreal faces issues tied more to weather extremes and geographic sprawl. Furthermore, demand volatility adds complexity to the operations, as order volumes fluctuate greatly based on time of day, weather, or local events, thus making it difficult to maintain optimal service levels and allocate resources efficiently. Understanding these city-specific challenges is essential for developing scalable, effective optimization solutions.

1.3.5 Strategies and Technological Innovation in the Last Mile

In response to the aforementioned logistical challenges, quick commerce platforms are increasingly turning to advanced technologies to streamline operations, reduce costs, and meet rising consumer expectations. Since speed,

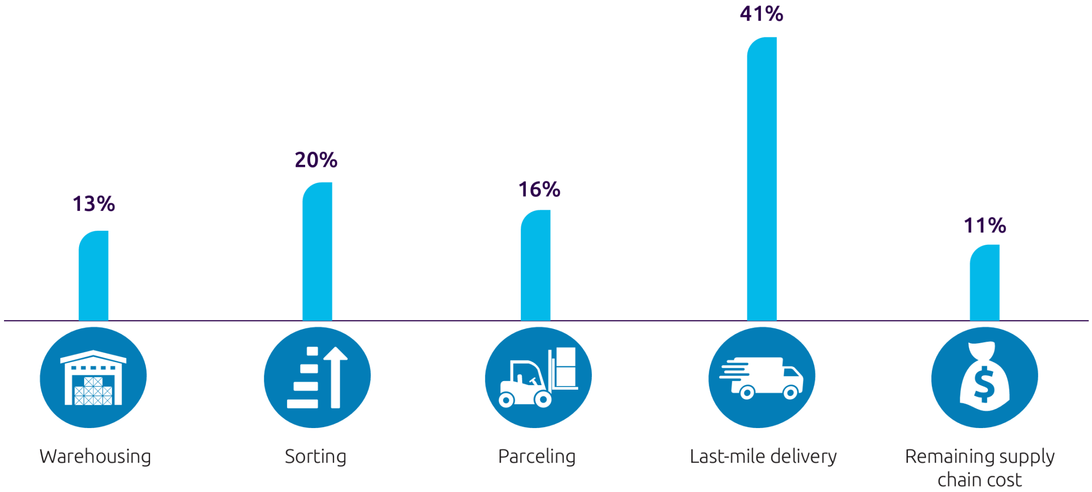


Figure 8: Breakdown of supply chain costs by activity, highlighting that last-mile delivery accounts for the largest share (41%) of total supply chain costs.

accuracy, and adaptability define the success of last-mile deliveries within this industry, innovations in routing algorithms, real-time tracking, and predictive analytics, all of which have been massively complimented and improved upon by the introduction of machine learning techniques and artificial intelligence have become essential components of the operational infrastructure.

One of the most critical innovations for all commerce platforms alike has been the real-time dynamic routing systems, which use traffic data, weather forecasts, and GPS to continually update delivery routes. Platforms such as Glovo and Uber Eats use these systems to reduce idle time, avoid congested areas, and assign deliveries to the most apt courier. For example, Glovo's algorithm, Jarvis, automatically assigns incoming orders to the most optimal courier based on a number of live factors such as courier's location, traffic conditions, vehicle type, preparation time at pick-up point, and even exceptional situations such as bad weather or local protests [Glovo, 2020]. This algorithm, based on knowledge in both artificial intelligence and operations research (i.e, mathematical problem-solving and decision-making methods), recalculates assignments for every few seconds in order to maximize efficiency for couriers, businesses, and end-users alike, thus resulting in a system that minimizes waiting times, and improving costs for the quick commerce platform.

In parallel, demand forecasting tools powered by AI have also become central to optimizing labor and inventory management. These tools can help predict hourly and weekly demand, and can even help predict fluctuations in demand based on factors such as weather, ultimately leading to better inventory management and a reduction of waste for said platforms, as well as better fleet placement during peak periods. A concrete example of this is international shipping giants UPS, who recently upgraded their proprietary algorithm named ORION (i.e., On-Road Integrated Optimization and Navigation), evolving it into dynamic Orion [Leonard, 2021]. The original model introduced in 2016 focused on optimizing routes based on static package and traffic data, helping drivers reduce approximately eight miles off their routes [Leonard, 2021], reducing annual fuel consumption and CO2 production by 10 million gallons and 100,000 metric tons respectively [INFORMS, 2016]. Meanwhile, the updated version integrates predictive analysis to forecast delivery surges and dynamically adjust courier routes in real-time. By analyzing historical trends, weather forecasts, and live operational data, dynamic ORION proactively positions couriers in high-demand zones before the volume materializes. This resulted in drivers shaving off another two to four miles per delivery day, consequently further reducing fuel consumption and carbon emissions, and also saving the company anywhere between US $300 – 400 million per year [INFORMS, 2016].

Order batching and consolidation algorithms are also gaining prominence in quick commerce logistics. Traditionally, quick commerce emphasized single-order deliveries to meet rapid fulfillment expectations. However, platforms are now exploring short batching windows that allow multiple nearby orders to be grouped and delivered in a single route, particularly during peak hours. This approach must balance the promise of fast delivery with operational efficiency. Research indicates that effective batching can significantly improve courier productivity and overall cost-efficiency. For instance, studies have shown that implementing optimized order batching strategies can reduce total order picking time and labor costs by substantial margins. Advanced algorithms, such as those based on metaheuristic approaches like the Improved NSGA-II, have been developed to optimize order batching and routing, leading to enhanced efficiency in warehouse operations [Xu et al., 2025].

Moreover, geofencing technologies, i.e., systems used to create virtual boundaries around specific geographical areas through the use of real-time information via floating data or RFID [Awati, 2022], have become instrumental in enabling hyper-local logistics, i.e., localized delivery strategies which focus on minimizing distance and time between fulfillment centers and customer drop-off points. By segregating zones based on traffic patterns and proximity to pick-up points, platforms can automate actions such as dispatching couriers, rerouting orders, and sending ETA updates or delivery confirmations. In turn, this enhances real-time coordination, thereby supporting faster courier assignments and minimizing delivery lag. In more advanced implementations, MFCs and dark stores are strategically set-up on location-based data, further reducing the time lag between order placement and delivery initiation, while also optimizing for operational costs[Yadav, 2024].

In summary, even with all these technological advancements, the last mile still remains the most complex and cost-intensive part of the quick commerce supply chain; however, advancements in real-time dynamic routing systems, AI-powered forecasting, order batching, and geofencing have greatly aided companies in this field. Nevertheless, research in last-mile optimization still continues, with companies looking to reduce their waste and increase their profitability as much as possible.

These ongoing challenges have sparked a growing body of research in operations management, urban logistics, and data science, with each offering different approaches to optimize the last-mile. From predictive modeling to heuristic route planning, a range of methods has been proposed and tested, often with varying success depending on the geographical context and delivery model in question. Yet, there remains a noticeable gap in comparative studies that examine how these strategies perform across different urban environments, particularly between European and North American cities.

1.4 Research Questions

To structure the inquiry and frame the empirical analysis, this thesis is guided by the following research questions:

- **How do classical assignment algorithms, such as the Hungarian method, compare to interpretable machine learning-based approaches (e.g., ANFIS neuro-fuzzy models) in optimizing courier assignment for last-mile delivery in Q-commerce?** What differences emerge in terms of delivery latency, operational cost, and resource utilization?
- **What are the main operational strengths and limitations of modern machine learning and neuro-fuzzy models when applied to real-time courier assignment, particularly regarding interpretability, adaptability, and scalability?**
- **How do the tested assignment algorithms perform across two contrasting urban environments, namely Montreal and Rome?** Are there any notable differences or consistent patterns in model effectiveness that can be linked to local regulatory, infrastructural, or compensation-related characteristics?
- **What practical insights and lessons can be drawn from simulation regarding the trade-offs, edge cases, and failure modes of each assignment strategy, and how might these inform the future design and deployment of last-mile delivery systems?**

By systematically addressing these questions, the thesis aims not only to benchmark assignment algorithms under controlled experimental conditions, but also to surface practical considerations that arise in deploying such systems in real urban contexts. To establish the academic foundation for this work, the following chapter reviews the existing literature on last-mile logistics, optimization techniques, and the application of data-driven methods in urban delivery systems. This review will identify key theoretical and methodological contributions to the field, while positioning the present thesis within the broader academic context by outlining the specific gaps it seeks to address.

2 Literature Review

2.1 Introduction

In recent years, city-bound transportation of goods has made a radical shift. The boom in online shopping, coupled with increasing consumer expectations of rapidity and convenience, has moved last-mile delivery, i.e., the last mile from distribution facilities to consumers' doorsteps, to the forefront logistics strategy and innovation. This phase of the supply chain is characterized by both high costs and considerable logistical complexity, necessitating sophisticated strategies to balance efficiency, profitability, and service quality; consequently, it has emerged as a primary focus for innovation and competitive differentiation within the logistics sector.

Of the most transformative technologies that have come out of this change, the most disruptive one has been the introduction of quick commerce, or Q-commerce. Not based on bulk orders and fixed delivery windows like most traditional models of e-commerce, Q-commerce operates on the premise of delivering small, recurrent orders within minutes, typically between 10 and 30. As such, this model requires the help of sophisticated logistics infrastructure, real-time processing, and flexible delivery pools. This commerce model has gained traction in thickly populated urban space, particularly in key European and North American cities, where companies such as Glovo, Getir, and

Uber Eats have grown rapidly. Its appeal lies in its ability to deliver near-instant gratification, aligning closely with the expectations of modern urban consumers.

However, the fast tempo and expectations of Q-commerce also bring new operational challenges. Perhaps the most elusive challenge is how to maximize delivery operations in real time, a goal which requires both accurate demand forecasting, and effective routing schemes. Although these two topics have each been heavily studied on their own, their marriage into a cohesive, reactive system that works well in rapidly moving urban environments remains an unaddressed and formidable problem. Additionally, the divergence of city designs, transportation infrastructure, and labor regulations from one municipality to the next creates further layers of complexity when attempting to deploy across-the-board solutions.

Therefore, this chapter will seek to examine these aforementioned challenges in detail, first by investigating global trends in logistics before zeroing in on the particular needs of Q-commerce. Drawing on existing research, it traces the evolution of work on demand prediction and courier optimization, while also delineating areas that remain underexplored. Specific attention will be given to the way that the particular conditions of urban environments in Europe and North America condition the application of these solutions. Ultimately, the scope of this chapter is to be able to give a clear impression of the state of the field today, and where exactly in that space the thesis hopes to bring new insights to, and further push development.

2.2 Global Context: Urban Logistics and the Rise of E-Commerce

The last decade has greatly transformed the global retail market, most notably driven by the increase in the use of e-commerce. With growing internet penetration, smartphone uptake, and shifting consumer tastes, online consumption has grown to become a prime source of consumption and business, both for the consumer and the enterprise. In 2024, global revenue from e-commerce surpassed $5.5 trillion, with forecasts indicating that it will reach levels of approximately $7 trillion by 2029 [Statista, 2025a]. Most of this growth comes from the Asian and American markets, which accounts more than 60% of the global revenue in e-commerce combined [Statista, 2025a]. Moreover, as online consumption continues to become more the norm, the share of e-commerce in total global retail sales is expected to rise from 12% in 2019 to over 21% by 2029 [Statista, 2025a].

These trends have put tremendous pressures on city logistics systems in particular. In contrast to conventional retail, which involves large shipments to centralized stores, fulfillment in e-commerce relies on getting millions of individual orders directly to the consumer. This transformation has shifted logistics from a business-to-business (B2B) to a business-to-consumer (B2C) one, thus requiring quicker fulfillment, flexible routing, and real-time responsiveness [Raj and Thandayudhapani, 2024]. The complexity of meeting these expectations in urban settings makes last-mile delivery the most expensive and logistically challenging phase of the supply chain [Fegde, 2025].

Urban areas present a different set of challenges to last-mile logistics. Congested traffic, limited parking, road barriers, and time-critical deliveries create inefficiencies. Fegde observes that last-mile delivery can cost as high as 53% of total logistics, primarily due to failed first-attempt deliveries and route variability [Fegde, 2025]. These city-related barriers have caused companies to implement new strategies, including the use of micro-fulfillment centers, electric vehicle usage, and the implementation of smart-routing technologies [Fegde, 2025].

Furthermore, the COVID-19 pandemic spurred the embrace of e-commerce across the population. As bricks-and-mortar stores closed during lockdowns, consumers switched in large numbers to the internet, a change in behaviour that persisted even once the restrictions were lifted. Zhu found that home delivery, mobile-first shopping, and online grocery purchases all increased during the pandemic and remained elevated in the post-COVID period[Zhu, 2024]. This structural shift reinforced the role of digital platforms in daily consumption and raised expectations for near-instant fulfillment and flexibility.

However, the acceleration of e-commerce has brought sustainability concerns to the forefront. More frequent deliveries create more traffic, higher emissions, and unnecessary, high levels of waste in containers and packing materials. As reported by Statista, almost 70% of online consumers abandoned their shopping carts in 2023, with 41% mentioning high delivery charges as the primary cause [Statista, 2025a]. This tension between consumer expectations for low-cost, fast delivery and the operational and environmental realities of fulfilling them has pushed both regulators and logistics providers to innovate. Efforts now include low-emission zones, electric vehicle fleets, consolidation hubs, and AI-powered route optimization [Raj and Thandayudhapani, 2024].

Combined, these pressures have remapped the logistics landscape. Urban logistics finds itself at the crossroads of speed, cost-effectiveness, and sustainability, a balancing act that set the stage for quick commerce (Q-commerce). This new model pushes the limits of responsiveness and presents new last-mile optimization challenges. As such, the following section will explore how Q-commerce evolved under these pressures.

2.3 The Emergence of Quick Commerce

Quick commerce rapidly gained ground as a distinct retail and logistics model, offering a more immediate alternative to traditional e-commerce and even same-day delivery. Rather than extending the logic of convenience alone, Q-commerce redefines the customer relationship around speed, precision, and proximity, enabling the delivery of curated essentials within as little as 10 to 30 minutes. This model is designed not just to fulfill needs, but to anticipate and trigger them, appealing to impulse, repetition, and situational demand.

What distinguishes Q-commerce most is not only the pace of fulfillment but the business architectures that support it. Companies have adopted different approaches depending on their capital structure, market maturity, and urban context. Broadly, three types of models have emerged: vertically integrated players operating proprietary networks of dark stores (such as Getir or Gorillas), aggregator platforms connecting users to third-party vendors (such as Uber Eats or Glovo), and hybrid models such as Blinkit or Zepto, which blend in-house fulfillment with marketplace flexibility [Sanchez, 2024]. These models respond differently to geographic constraints. Dark stores require dense, high-frequency urban demand, while aggregators scale more easily in fragmented retail environments.

From a logistics standpoint, the success of Q-commerce hinges on coordination between micro-fulfillment infrastructure and courier operations. Companies use predictive analytics and algorithmic scheduling to route deliveries, allocate couriers, and rebalance inventories across neighborhoods. Stojanov emphasizes that real-time data integration is essential, especially in high-traffic cities, where even marginal inefficiencies erode the viability of short delivery windows [Stojanov, 2022]. The supply chain, therefore, becomes hyper-localized and demand-sensitive, often operating through small nodes with high turnover and minimal stock variance.

These operational demands have led to significant innovation in fleet management and workforce design. Many platforms rely on flexible labor models, allowing them to quickly scale capacity in response to demand fluctuations. However, this comes with regulatory and ethical trade-offs, particularly around employment classification, algorithmic control of worker behavior, and wage variability. These tensions are especially relevant in European cities, where the expansion of gig-based delivery models has prompted increasing scrutiny of labor conditions and employment structures [Stojanov, 2022].

At the same time, customer retention has become an equally strategic concern, given the narrow margins and high operational costs of Q-commerce. Dr. Sharma and Goel highlight that platforms like Zepto and Blinkit have begun differentiating through service personalization, using app-based behavior tracking, loyalty rewards, and differentiated delivery tiers to encourage repeat usage[Goel and Sharma, 2025]. Globally, companies like Delivery Hero report that a majority of Q-commerce revenues are driven by returning users, underscoring the importance of not just acquiring but sustaining a high-frequency customer base [Stojanov, 2022].

While the business case for Q-commerce remains under scrutiny, especially regarding long-term profitability, its influence on urban delivery expectations is already visible. More than a temporary disruption, Q-commerce represents a structural shift in how cities, platforms, and consumers interact. Its reliance on decentralized infrastructure, real-time orchestration, and demand-aware logistics presents both a blueprint and a challenge for future last-mile systems. This is particularly relevant in the comparative context of North America and Europe, where urban design, regulation, and platform maturity vary significantly. The next section turns to these spatial and infrastructural constraints, focusing on how they shape the complexity of last-mile delivery in each context.

2.4 The Last-Mile Delivery Problem

Last-mile delivery is widely regarded as the most complex and cost-intensive segment of the logistics chain. In conventional e-commerce, it accounts for a disproportionately high share of total delivery costs, estimated at around 28 percent, largely due to inefficiencies in routing, traffic variability, and the decentralized nature of delivery points [Kumar and Chidambara, 2024]. In quick commerce, this complexity is further intensified. The last mile becomes more than a logistical function; it is the core component of the customer experience. In this model, speed and reliability are not just performance metrics but the value proposition itself, and the failure to execute consistently undermines the entire service.

Urban environments amplify the difficulty of executing last-mile operations at scale. As demand for fast delivery increases, so do emissions, congestion, and pressure on public space. According to Boggio-Marzet et al., urban last-mile delivery is expected to contribute to a 32 percent increase in carbon emissions by 2030 if current practices continue [Boggio-Marzet et al., 2023]. Dense traffic, scarce curbside access, and the fragmentation of deliveries across neighborhoods limit the feasibility of optimizing delivery density and routing. Infrastructure is often insufficient to support the volume and frequency demanded by Q-commerce, especially in historical city centers and mixed-use districts. These spatial constraints affect not only delivery times but also the consistency and sustainability of service.

A central challenge in last-mile logistics is the high rate of failed deliveries. These failures are often caused by the absence of recipients, inadequate address accuracy, or poor coordination between platforms and couriers. They result in repeated delivery attempts, increased operational costs, and higher emissions. Boggio-Marzet et al. emphasize that these inefficiencies are often structural, arising from a lack of alignment between public and private stakeholders [Boggio-Marzet et al., 2023]. While city governments typically prioritize sustainability and public interest, platforms focus on speed, customer satisfaction, and unit economics. This misalignment leads to missed opportunities for collaboration on micro-depots, shared delivery infrastructure, and low-emission policies.

Technological innovation has provided partial solutions. Advanced dispatch systems and dynamic matching algorithms have enabled platforms to assign deliveries in real time based on proximity, road conditions, and courier availability. Li et al. developed an adaptive matching method for city express delivery that significantly improved fulfillment times by dynamically reallocating orders based on real-time conditions. However, the effectiveness of such systems is dependent on reliable data and stable operational environments [Li et al., 2023]. In congested cities, where infrastructure is unpredictable and demand fluctuates hour by hour, even the best algorithmic solutions encounter friction.

Additionally, the spatial and temporal nature of demand makes optimization difficult. Unlike traditional models that benefit from delivery consolidation, Q-commerce involves fragmented, time-sensitive orders that leave little room for route efficiency. The systematic review by Kumar & Chidambara identifies that the non-linearity of urban demand and limitations in delivery zone flexibility hinder the potential for operational gains in last-mile logistics[Kumar and Chidambara, 2024]. Moreover, there is often a lag between system optimization and real-world feedback, particularly when data infrastructure is uneven or delayed.

Despite extensive innovation, the spatial implications of last-mile delivery remain underdeveloped in urban planning discourse. Kumar and Chidambara note that the integration of e-commerce logistics into city planning frameworks has not kept pace with the growth of digital retail [Kumar and Chidambara, 2024]. Physical infrastructure such as bike lanes, delivery zones, and curbside management continues to be planned separately from digital commerce systems. As a result, logistics platforms are forced to retrofit operations into city environments that were never designed for real-time, high-frequency delivery activity.

Ultimately, the last mile in Q-commerce is where the promise of ultra-fast fulfillment meets the limits of urban complexity. As demand grows and expectations rise, platforms are challenged to deliver not only faster, but smarter and more sustainably. The inability to optimize this final step without compromising speed or quality reveals the systemic fragility of current models. Addressing this problem requires an integrated approach that spans technology, labor, urban policy, and infrastructure design. The next section will explore the tools platforms use to forecast demand and assign couriers, as these are the levers most directly linked to performance in the last mile. As these operational complexities intensify, platforms have increasingly turned to advanced algorithmic approaches for matching couriers to orders and optimizing delivery flows.

2.5 Algorithmic Approaches to Courier Assignment and Routing in Q-Commerce

The operational complexity described in the previous section finds its sharpest expression in the assignment and routing decisions that underpin last-mile delivery in Q commerce. Once the contours of the last mile problem are established, the natural progression in the literature is to address the ways in which orders are matched with available couriers and how those couriers are guided through urban environments characterized by traffic variability, regulatory constraints, and fluctuating demand. It is within this context that algorithmic strategies have evolved from intuitive, human-designed heuristics to sophisticated, data driven systems capable of responding to real-time uncertainty.

2.5.1 Classical Approaches: Heuristics and Combinatorial Optimization

The earliest solutions to the courier assignment problem relied on straightforward heuristics. The nearest-neighbor approach, in which orders are assigned to the geographically closest available courier, offers simplicity and transparency. This approach continues to serve as a benchmark in both industry and research, primarily for its computational efficiency and intuitive logic. However, its inherent myopia and inability to incorporate broader system dynamics limit its effectiveness, particularly as platforms scale and urban environments grow denser and more complex.

Combinatorial optimization models emerged as a response to these shortcomings. The Hungarian Algorithm, originally developed by Kuhn and Munkres, is widely recognized as a foundational technique for the one-to-one assignment problem. By constructing a cost matrix, typically reflecting travel distances or estimated times, it can identify the minimum-cost assignment of couriers to orders at a given moment. Its mathematical elegance and computational speed make it particularly attractive for batch assignments and controlled environments. Still, its static formulation means it cannot natively accommodate the asynchronous and rapidly changing conditions characteristic of real-world Q commerce. Extensions such as minimum-cost flow models, integer programming, and metaheuristics like genetic algorithms and ant colony optimization have added flexibility for vehicle capacity, batching, and more nuanced constraints, yet often at the expense of increased computational demands and diminished real-time responsiveness [Vásconez et al., 2024][Reyes et al., 2018]. While classical algorithms provide computational efficiency, their lack of flexibility under dynamic, high-volume conditions motivates the integration of batching strategies.

2.5.2 Batching and Order Bundling in Last-Mile Assignment

Batching, i.e., the grouping of multiple orders for delivery by a single courier, has emerged as a critical operational strategy in last-mile and quick-commerce logistics. By consolidating orders into batches, platforms can increase vehicle utilization, reduce total travel distance, and mitigate the costs and inefficiencies associated with one-to-one assignment, especially in dense urban environments. However, batching also introduces complexity and operational trade-offs: larger batches may increase delivery times and the risk of missed service windows, particularly when order preparation or travel times are uncertain.

The assignment and batching problem is fundamentally combinatorial and becomes especially challenging in dynamic, real-world scenarios. Recent work demonstrates that the joint batching and assignment of orders to vehicles in dynamic road networks is NP-hard and inapproximable in polynomial time [Joshi et al., 2022, Joshi et al., 2021]. The *FoodMatch* algorithm, for example, recasts the batching and assignment process as a bipartite matching problem, then uses graph clustering to form cost-effective batches while ensuring scalability in real workloads. Experimental results on real food delivery data show that batching, when optimized, can reduce average delivery latency by up to 30% while increasing the number of orders delivered per kilometer [Joshi et al., 2022, Joshi et al., 2021].

Metaheuristic and hybrid algorithms have been increasingly adopted to address the computational challenges of batching in large-scale settings. Hybrid genetic algorithms, for instance, offer a flexible approach for jointly solving assignment and batching problems, accommodating operational constraints such as heterogeneous vehicle fleets and variable order sizes [Ou et al., 2024]. Recent systematic reviews highlight the effectiveness of these approaches for logistics and warehouse settings, providing a transferable foundation for quick-commerce delivery systems.

Beyond heuristic and algorithmic advances, recent analytical frameworks have integrated the full fulfillment pipeline, i.e., order picking, batching, and last-mile delivery, to capture the interdependencies between processes [Raj et al., 2024 These models reveal important managerial trade-offs: while increasing batch sizes typically lowers per-order costs, it can substantially increase both the mean and variance of delivery times, thereby threatening service reliability. Simulation and optimization studies consistently find that the optimal batch size depends sensitively on demand rates, promised delivery times, and resource availability. Under strict service-level constraints, platforms may need to revert to single-order dispatching to guarantee reliability, especially during peak periods [Raj et al., 2024].

In summary, batching is a powerful but complex lever in last-mile logistics. Its effectiveness depends on accurate demand forecasting, dynamic assignment logic, and the ability to balance operational efficiency with customer service commitments. This thesis contributes to the literature by empirically benchmarking both batching-enabled (machine learning-based) and non-batching (classical) assignment strategies within a modular simulation framework, providing simulation-based evidence of the real-world trade-offs of batching in quick-commerce delivery. These operational challenges have accelerated the adoption of machine learning and hybrid methods, which can dynamically adapt assignment and batching policies in response to real-time data.

2.5.3 Fuzzy Logic and Neuro-Fuzzy Systems

While machine learning enables adaptation to complex patterns in the data, there remain many operational scenarios in last mile logistics that are best described through rules capturing human reasoning under uncertainty. Fuzzy logic models provide a formal structure for encoding such ambiguous decision rules, allowing for nuanced assignments based on concepts such as being close enough, sufficiently available, or encountering heavy traffic. This interpretability is particularly valuable in settings where expertise and operational knowledge must be translated into algorithmic action.

The development of neuro-fuzzy systems, and most notably the adaptive neuro-fuzzy inference system, represents a significant step forward in this domain. ANFIS combines the transparent rule structure of fuzzy logic with the learning capabilities of neural networks, enabling the system to automatically tune membership functions and optimize decision rules from data [Presskila et al., 2025]. In Q-commerce and related e-logistics settings, neuro-fuzzy models such as ANFIS have proven highly effective at capturing the context-dependent trade-offs that arise in real-time decision-making. Their unique combination of qualitative, expert-driven rules with data-driven learning makes them especially well suited for environments characterized by operational uncertainty and rapidly changing priorities. As demonstrated by Hamdan et al., ANFIS models not only deliver high predictive accuracy for real-time e-order arrivals in complex logistics networks, but also maintain computational tractability and interpretability, outperforming traditional approaches such as ARIMA in both separate and combined order streams [Hamdan et al., 2023].

2.6 Simulation-Based Benchmarking of Assignment and Routing Strategies

A consistent challenge in last-mile delivery research is empirically comparing the performance of assignment and routing strategies across the diversity of real-world operational contexts. Because field experiments are often costly and subject to unobservable confounders, researchers increasingly turn to agent-based simulation models and digital twin environments to rigorously evaluate algorithmic approaches under controlled, yet realistic, conditions.

Multi-agent simulations (MAS) and agent-based models (ABM) offer a natural framework for representing the heterogeneous and interactive nature of modern delivery platforms. Recent work by Lu et al. employs a data-driven agent-based simulation to explore how static and dynamic order dispatch strategies, coupled with real-time routing and resource optimization, affect the operational performance of autonomous delivery vehicles in urban fresh food e-commerce[Lu et al., 2022]. Their findings demonstrate that scenario simulation and sensitivity analysis are essential for identifying bottlenecks and testing the robustness of dispatch and routing policies, especially as demand fluctuates and real-world constraints, such as limited vehicle capacity and customer time windows, are layered onto the model.

Similarly, Zou et al. leverage agent-based simulation to systematically compare the efficiency of alternative delivery strategies within the complex adaptive context of online-to-offline (O2O) platforms [Zou et al., 2023]. Their work highlights how agent-based approaches capture emergent behavior arising from the interplay between dispatchers, merchants, couriers, and customers. In their experiments, a TSP-based assignment strategy is shown to outperform simple proximity-based heuristics in terms of order completion rates and cost efficiency, though the authors also underscore the importance of tuning model parameters, such as courier load capacity and assignment policies, to local operational realities.

Extending this paradigm, Fan et al. introduce a hybrid simulation-optimization framework that integrates agent-based courier reinforcement learning with real dispatch data from Meituan, a leading Chinese delivery platform [Fan et al., 2024]. Their AD-MACRO framework demonstrates the potential for combining optimization-based matching with distributed RL-based adaptation, achieving measurable gains in delay reduction and equity in courier workloads. Notably, the model explicitly incorporates courier behavioral responses, such as order rejection and speed adaptation, reinforcing the need for simulation environments that account for both platform- and agent-level decision-making.

The simulation-based approach is not limited to courier-customer assignment. Andersson, in a comparative study of dynamic pickup and delivery problems, evaluates the operational benefits of advanced heuristic and metaheuristic routing algorithms within a synthetic environment that mirrors the constraints and temporal uncertainty of real-world courier operations [Andersson, 2021]. The study finds that leveraging optimization-informed assignment strategies, such as those based on large neighborhood search, can accommodate more requests and reduce operational costs relative to purely heuristic methods.

Recent conference proceedings also reflect this trend, with Kondratov and Tarasova presenting evidence that the aggregation of orders (bundling) and adaptive courier assignment can meaningfully enhance delivery efficiency in urban environments [Kondratov and Tarasova, 2024]. Their simulation results point to the critical role of strategic bundling and assignment coordination, especially in dense city contexts where batching opportunities are frequent and the operational space is highly dynamic.

Taken together, these simulation-based studies highlight the methodological shift in the literature toward digital experimentation as a tool for benchmarking assignment, batching, and routing strategies in last-mile logistics. By enabling controlled experimentation across a spectrum of scenarios, ranging from stochastic demand surges to variable traffic patterns, agent-based and hybrid simulation models provide a robust foundation for testing new algorithms and informing real-world platform design. This thesis builds directly on these advances, leveraging simulation to empirically evaluate the comparative performance of classical, machine learning, and fuzzy logic-based assignment strategies across contrasting urban contexts.

2.7 Advances in Dynamic, Integrated, and Real-Time Courier Assignment

Recent years have seen rapid innovation in the modeling, optimization, and empirical evaluation of courier assignment systems for Q-commerce and last-mile delivery. The contemporary landscape is characterized by an increasing convergence between real-time data, adaptive algorithms, and the explicit modeling of both operational constraints and human behavior. This section reviews the evolution from static assignment and routing toward dynamic, integrated, and data-driven frameworks.

A foundational challenge in urban courier assignment is the need to balance efficiency, service quality, and adaptability under uncertainty. Chang and Yen [Chang and Yen, 2012] formalize the city-courier routing and scheduling problem as a multi-objective multiple traveling salesman problem with strict time windows (MOMTSPSTW), capturing the unique constraints of urban environments: asymmetric networks, no-parking streets, strict service windows, and the importance of workload balancing. Their multi-objective Scatter Search framework generates Pareto-optimal solutions and highlights that workload balancing is often computationally more demanding than mere route optimization.

Building on such foundations, Ninikas et al. [Ninikas et al., 2014] and Restrepo et al. [Restrepo et al., 2019] advocate for the integration of tactical and operational planning, merging mass deliveries and dynamic requests in real-world courier operations. These studies introduce rolling-horizon frameworks, two-stage stochastic programming, and heuristic methods for real-time order insertion, demonstrating robust cost improvements and enhanced courier utilization in practical settings. Notably, Restrepo et al. show that jointly optimizing shift schedules and load assignment under stochastic demand yields solutions that are both cost-efficient and sensitive to worker preferences, which is key for platform viability and retention.

With the rise of on-demand platforms, the literature has increasingly moved toward real-time, context-aware assignment. Ferrucci and Bock [Ferrucci and Bock, 2014] model the Dynamic Pickup and Delivery Problem with Real-Time Control (DPDPRC), incorporating heterogeneous vehicles, real road networks, and dynamic events (traffic, breakdowns). Their Tabu Search-based real-time adaptation illustrates the tangible performance gains available from continuous plan adaptation, especially under tight time windows and operational volatility.

At the same time, the economic and behavioral dynamics of platform work are receiving greater attention. Chen and Hu [Chen and Hu, 2024] systematically analyze the trade-offs between dedicated (single-order) and batched (pooled) dispatching, demonstrating through spatial queuing models and simulation that customer patience, demand endogeneity, and service area size all critically shape optimal assignment strategy. Their findings emphasize the necessity of context-dependent dispatch policies, particularly in urban environments where customer tolerance for delay is limited.

Crucially, real-world assignment and routing are shaped by courier agency and behavioral complexity. Zhang et al. [Zhang et al., 2019] use large-scale empirical data from Ele.me to demonstrate that courier decision-making diverges significantly from platform-optimized routes, with perceived (rather than physical) distance and psychological factors often dominating actual routing behavior. Incorporating these insights through machine learning-based route prediction reduces overdue deliveries and aligns assignment strategies more closely with human operational realities.

Recent advances at the intersection of machine learning and optimization are further transforming assignment paradigms. Fan et al. [Fan et al., 2024] propose AD-MACRO, a two-layer framework that combines distributed multi-agent reinforcement learning with optimization-based matching, validated on Meituan's operational data. Their Grouped-Actor PPO method accommodates heterogeneous courier pools, models rejection behaviors, and incorporates fairness constraints, yielding substantial gains in delivery efficiency, equity, and system robustness.

Taken together, these contributions mark a decisive shift from static or myopic assignment models to architectures that are dynamic, context-sensitive, and robust to uncertainty and human variability. The most effective frameworks now integrate tactical and operational planning, utilize real-time data, and explicitly model the behavioral, economic, and regulatory realities of Q-commerce platforms. Reflecting these developments, this thesis adopts a city-specific modeling approach, illustrating how flexible, data-driven assignment strategies can be adapted and instantiated for distinct urban environments. By focusing on Rome and Montreal, i.e., cities with unique logistical, infrastructural, and platform characteristics, this work highlights the importance of context in both model design and empirical validation. As urban last-mile delivery continues to evolve, ongoing research is likely to further close the gap between algorithmic optimality and real-world complexity, enabling more resilient and adaptive assignment strategies.

2.8 Identified Research Gaps and Thesis Contribution

Despite substantial progress in last-mile delivery optimization, the assignment of couriers to orders remains a challenging and incompletely solved problem, particularly under real-time, stochastic, and operationally complex conditions. While recent literature has introduced a variety of machine learning, hybrid, and heuristic strategies, most empirical studies either benchmark purely algorithmic approaches or rely on black-box models, often neglecting interpretability, adaptability, and the operational realities of batching and order bundling.

Key gaps identified in the literature include:

- **Empirical benchmarking of batching-enabled assignment strategies:** While batching and order bundling have shown significant promise in reducing operational costs and improving efficiency, there remains a lack of controlled, simulation-based comparisons between batching-enabled (machine learning or metaheuristic) and non-batching (classical) assignment approaches under realistic urban demand.
- **Scenario diversity and synthetic data generation:** Many prior studies are limited by the availability of real-world data or rely on narrow operational scenarios. There is a need for systematic benchmarking pipelines that generate diverse, realistic synthetic datasets, including ground truth labels and stochasticity, to robustly evaluate assignment strategies across cities, demand levels, and network conditions.
- **Interpretable and context-aware models:** The potential of neuro-fuzzy systems such as ANFIS for dynamic, interpretable, and context-sensitive assignment in Q-commerce remains underexplored. Most industry and academic deployments still rely on either rigid rule-based heuristics or opaque black-box predictors.

This thesis addresses these gaps by:

- Developing a modular simulation framework that enables direct empirical comparison of batching-enabled (ML-based) and non-batching (classical optimization) assignment strategies under identical conditions;
- Implementing a scenario-diverse synthetic data generation pipeline, including explicit ground truth creation and noise injection, to support robust benchmarking across distinct urban contexts (Rome and Montreal);
- Applying and adapting neuro-fuzzy (ANFIS) and hybrid assignment models to provide interpretable, data-driven, and context-aware solutions for courier-order matching, and systematically evaluating their operational trade-offs against classical and batching-enabled approaches.

Through this approach, the thesis advances both methodological and practical understanding of assignment systems for Q-commerce, providing new insights into the interplay between batching, interpretability, operational efficiency, and urban context.

2.9 Conclusion of Literature Review

In summary, the literature reveals significant advances in modeling, optimizing, and empirically benchmarking courier assignment strategies for Q-commerce and last-mile delivery. However, persistent gaps remain in the development of interpretable, batching-enabled, and scenario-robust assignment systems that can adapt to the complexities of real-world urban logistics. The present thesis directly addresses these gaps by introducing a modular simulation framework for comparative benchmarking, developing scenario-diverse synthetic datasets with controlled ground truth generation, and evaluating the operational trade-offs of batching, interpretability, and context awareness through the application of ANFIS and hybrid models. The next chapter details the methodological framework and experimental protocol that underpin these contributions.

3 Methodology

3.1 Overview of Research Design

This chapter presents the methodological framework developed to address the courier assignment problem in last-mile delivery, with a focus on the operational dynamics of quick commerce platforms. While the overarching goal is to advance data-driven optimization strategies for Q-commerce, the empirical analysis is anchored in the context of restaurant delivery, selected for its rich data environment and operational similarity to broader Q-commerce logistics. This domain provides a realistic, high-frequency testbed that captures the time sensitivity, spatial density, and logistical complexity characteristic of modern last-mile operations.

The research methodology is structured around a modular and extensible simulation pipeline, composed of the following core components:

- **Synthetic data generation:** Orders and couriers are generated using context-aware procedures, calibrated to reflect realistic spatial, temporal, and operational patterns for each city.
- **Simulation engine:** A configurable simulation environment models urban delivery flows and courier behavior under a range of plausible scenarios, allowing for systematic benchmarking of assignment logic.
- **Algorithmic suite:** The framework integrates both classical optimization methods (such as the Hungarian method) and interpretable machine learning models, notably Adaptive Neuro-Fuzzy Inference Systems (ANFIS), to assign couriers to orders.
- **Evaluation metrics:** A diverse set of metrics is used to capture key performance objectives, including delivery latency, operational cost, fairness, and batching efficiency.

Rome and Montreal are selected as representative urban case studies. Rather than conducting a direct empirical comparison of city performance, these cities are chosen to illustrate both the adaptability of the modeling framework across different geographies, infrastructures, and regulatory environments, and to highlight meaningful operational contrasts. The intent is twofold: first, to demonstrate that the simulation pipeline and assignment strategies are robust and transferable across distinct urban contexts; and second, to analyze how local characteristics, particularly regulatory and compensation structures, influence the suitability and effectiveness of different assignment logics. In this way, the study provides insight not only into the generalizability of the modeling approach, but also into the nuanced trade-offs that arise when deploying these models in practice.

The overall research process unfolds through the following main steps:

- **Data generation and preprocessing:** Generation of context-aware synthetic orders and couriers for each city, ensuring realistic spatial, temporal, and operational attributes.
- **Simulation of urban delivery operations:** Modeling delivery flows and courier interactions within each city, capturing dynamic operational realities and constraints.
- **Assignment strategy implementation:** Assigning couriers to orders using a range of algorithmic approaches, including both classical optimization and interpretable machine learning models.
- **Empirical benchmarking:** Evaluating the performance of each assignment strategy using standardized metrics that reflect the complex objectives of last-mile delivery.

The remainder of this chapter details each methodological stage, beginning with data generation and preparation, followed by the simulation environment and assignment algorithms, and concluding with the evaluation framework, experimental protocol, and implementation details.

3.2 Libraries and Computational Tools

The implementation of all simulation, data processing, and modeling components relied on a robust open-source Python ecosystem, ensuring both reproducibility and scalability. The following core libraries and tools were used throughout the project:

- **pandas** [McKinney, 2010]: For data loading, cleaning, preprocessing, and tabular manipulation.
- **numpy** [Harris et al., 2020]: For numerical computations and efficient array operations.
- **scikit-learn** [Pedregosa et al., 2011]: For machine learning utilities including train-test splitting, standardization, principal component analysis (PCA), and model evaluation metrics.
- **joblib** [developers,]: For efficient serialization and saving/loading of models and scalers.
- **osmnx** [Boeing, 2017]: For street network extraction and urban graph construction from OpenStreetMap data.
- **folium**: For interactive mapping and geographic visualization.
- **googlemaps**: For geocoding and travel-time estimation via the Google Maps API.
- **tqdm**: For progress bars in data processing loops.
- **PyTorch** [Paszke et al., 2019]: As the neural computation backend for custom and third-party neuro-fuzzy models.
- **xanfis**[Thieu, 2025]: For modern ANFIS and neuro-fuzzy modeling, including the BioAnfisRegressor implementation.

- **scipy** [Virtanen et al., 2020]: For optimization routines, including the linear sum assignment (Hungarian algorithm).

Other standard Python libraries, such as `pickle`, `uuid`, `random`, `datetime`, `os`, and `copy`, were employed for supporting tasks including serialization, unique identifier generation, random sampling, time management, and file operations.

All analysis and experimentation were conducted using Python 3.9+ in a local development environment (macOS), with full versioning and dependency management to ensure result reproducibility. Additional utilities, such as `matplotlib` and `seaborn`, were used for exploratory analysis and visualization where appropriate.

3.3 Simulation Pipeline with Machine Learning Model Training

The methodological pipeline developed for this thesis distinguishes greatly between traditional assignment strategies and modern machine learning-based approaches, both in terms of workflow and computational requirements. Figure 9 illustrates the overall process.

For classical assignment strategies (such as Naive and Hungarian), simulation can proceed immediately once the synthetic agents (orders and couriers) have been generated from preprocessed input data. These algorithms are model-free, i.e., they make assignment decisions based on well-defined heuristics or optimization routines, without the need for prior exposure to historical or simulated operational data; consequently, their implementation requires no pre-simulation model training.

In contrast, machine learning-based strategies, including ANFIS and hybrid approaches, require a pre-simulation model training phase. Before these assignment modules can be used in the main simulation, they must first be trained on synthetic data that captures the operational characteristics, spatial layouts, and demand patterns unique to each city and scenario. This synthetic dataset is generated by simulating a wide variety of order-courier interactions and outcomes, effectively creating a "digital twin" of the delivery environment. The machine learning models (e.g., ANFIS) are then trained to predict assignment outcomes, learning the complex relationships between features such as travel time, vehicle type, weather, and local demand. Only after this supervised training step can the models be deployed for real-time decision-making within the main simulation loop.

The overall simulation pipeline therefore consists of:

- **Input Data Collection and Preprocessing:** Raw data on restaurants, weather, and urban street networks (extracted via OSMnx) is cleaned, geocoded, and filtered to form the basis for simulation.
- **Synthetic Agent Generation:** Orders and couriers are instantiated with attributes reflecting realistic spatial and temporal variation.
- **(For ML strategies) Synthetic Dataset Creation and Model Training:** A pre-simulation process generates a training dataset of courier-order interactions and labels, on which machine learning models are trained to optimize assignment decisions for each operational context.

- **Simulation Execution:** The simulation engine iterates through discrete time steps, at each moment invoking the selected assignment strategy (classical or ML-based) to match available couriers to new orders.
- **Performance Logging and Analysis:** Throughout the simulation, key metrics (such as delivery latency, cost, courier utilization, and batching efficiency) are systematically logged for later benchmarking and visualization.

This clear separation between model-free and model-based assignment strategies ensures that each approach is implemented under conditions that reflect its practical requirements. It also allows for a fair and transparent empirical comparison: classical strategies operate "out of the box," while machine learning strategies leverage data-driven adaptation only after an initial investment in scenario-specific training.

To ensure fair benchmarking, the entire simulation pipeline was repeated three times for each scenario: once using Google Maps API travel times with both the classical assignment strategies and the machine-learning based strategies, once using only haversine distances for assignment, and once using machine learning-based models (ANFIS/Hybrid) with batching enabled and haversine-derived features. All other components and agent initializations were held constant across modes.

The remainder of this chapter details each stage of the pipeline, with careful attention to the data flows and decision points specific to each assignment approach.

3.4 Data Collection and Preparation

The empirical backbone of this study relies on a combination of publicly available and open-source datasets, selected to provide realistic, high-frequency scenarios for simulating last-mile delivery in urban environments. The guiding principle in dataset selection was the need to capture both the spatial and temporal dimensions of demand and supply, as well as operational factors such as weather, which can significantly influence delivery logistics.

3.4.1 Restaurant Location Data

To simulate realistic order origins and destinations in each city, geolocated restaurant datasets were used as proxies for Q-commerce fulfillment points. For Rome, the restaurant data was sourced from the Foodi-ML dataset, a Github repository containing over 1.5 million unique images and more than 9.5 million store and product records gathered from the Glovo application across 37 European countries[1]. For Montreal, a comprehensive DoorDash delivery dataset was obtained from Kaggle, encompassing thousands of restaurant locations and associated delivery records across Canada[2]. These datasets ensure a realistic spatial distribution of orders and reflect the heterogeneous density and clustering patterns found in real urban food delivery networks.

[1] `https://github.com/Glovo/foodi-ml-dataset`

[2] `https://www.kaggle.com/datasets/satoshiss/food-delivery-in-canada-door-dash`

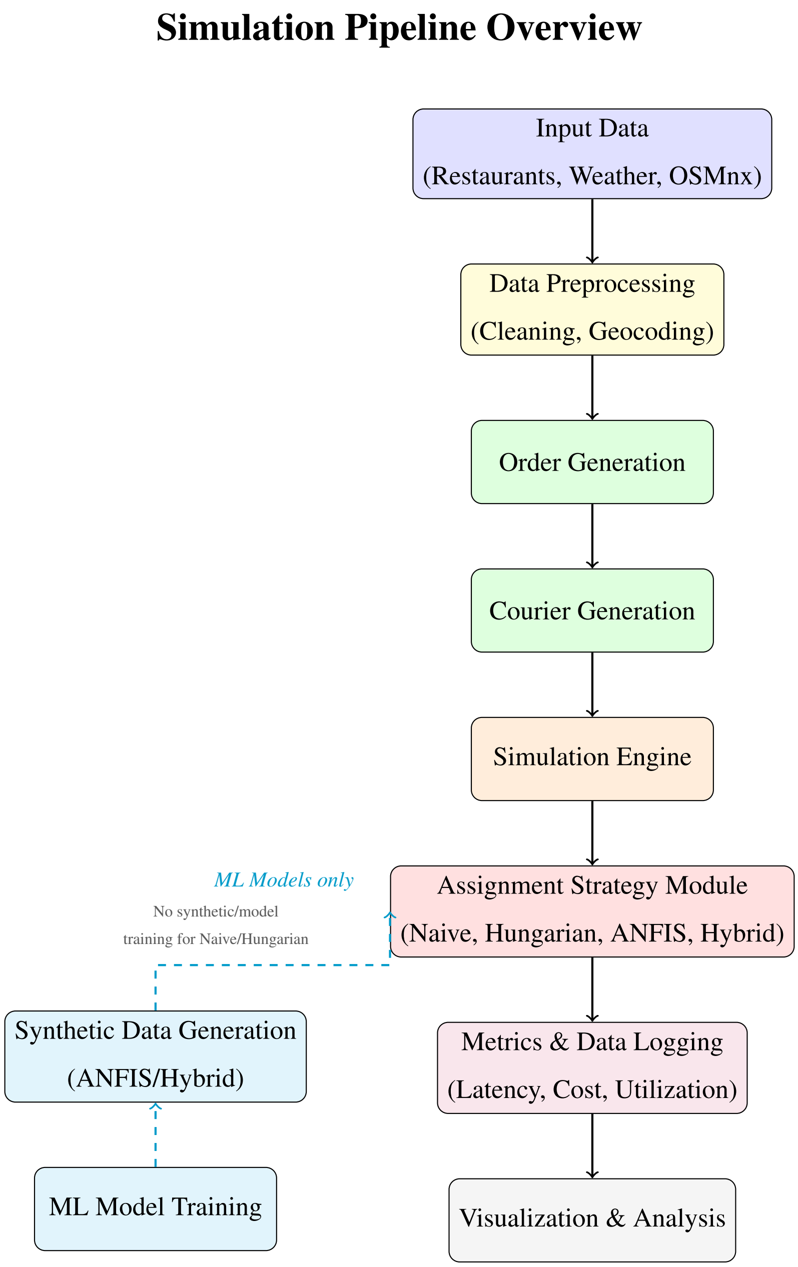


Figure 9: Simulation pipeline for courier assignment. The entire pipeline is executed in three experimental modes: (1) using real travel-time data from the Google Maps API with classical assignment strategies, (2) using only haversine distance estimates for classical assignment, and (3) using machine learning models (ANFIS/Hybrid) with batching enabled and haversine-based features. Each mode processes the same synthetic agents and scenario data to ensure comparability of results.

3.4.2 Weather Data

Given the substantial impact of weather on delivery times, demand variability, and courier availability, historical weather data was incorporated for both Rome and Montreal. For Rome, weather records covering the simulation period from 2021 to 2025 were retrieved from Meteostat, which aggregates station-based meteorological data including temperature, precipitation, wind speed, and other relevant variables[3]. Montreal weather data was collected from WeatherStats, a Canadian meteorological repository offering downloadable, high-resolution daily weather metrics for comparable periods[4]. These weather datasets were aligned temporally with the simulated order timelines, allowing for context-aware demand generation and dynamic courier assignment.

3.4.3 Data Cleaning and Preprocessing

All datasets underwent an extensive, multi-stage cleaning and preprocessing pipeline tailored to the specifics of each city and data source.

For the restaurant data, the initial step was to retain only records located within the urban boundaries of Rome and Montreal, using city and country codes as filters. Additional columns unrelated to spatial or operational modeling (such as product details, reviews, and URLs) were dropped to reduce noise and redundancy. To ensure uniqueness, restaurant names were normalized by removing accents and special symbols, standardizing case, and collapsing duplicates, thereby avoiding issues arising from minor naming discrepancies.

Geocoding was performed via the Google Maps API to obtain accurate latitude and longitude coordinates for each unique restaurant entry. Restaurant locations were mapped and spatial filters were applied to focus on operationally significant urban zones. In Rome, restaurants were selected from historically dense neighborhoods such as Monti, Trastevere, Trevi, and Campo Marzio. In Montreal, a subset was chosen from core boroughs and neighborhoods with high delivery density, including Ville-Marie, Le Plateau-Mont-Royal, Saint-Henri, Vieux-Montréal, and Mile-End. For computational tractability and balanced scenario design, the final dataset was further subsampled to create a manageable, representative simulation environment. Ultimately, 29 restaurants were retained for the Rome dataset and 40 for the Montreal dataset, reflecting the application of all geographic and operational filtering criteria.

For each restaurant, reverse geocoding (using Nominatim) assigned neighborhood or borough labels, enabling targeted analyses of intra-city variation and operational "hot spots." Entries with missing coordinates or failed geocoding were excluded. The cleaned and filtered datasets were then standardized to a uniform column schema for seamless integration into the simulation framework.

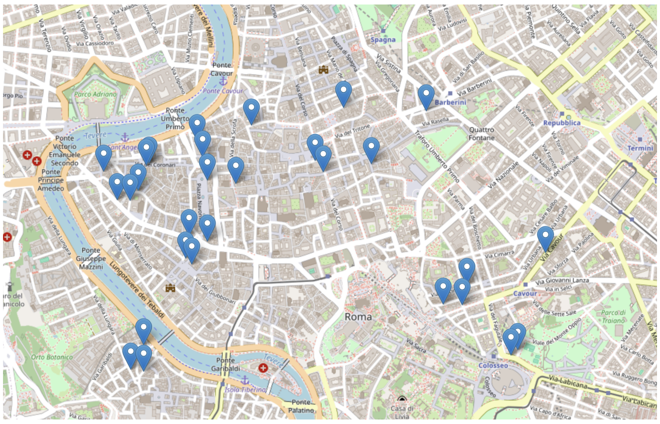
Weather data preprocessing followed a parallel pipeline. For both Rome and Montreal, irrelevant columns (such as wind direction or solar hours) were dropped. Missing values in precipitation or snow were filled with zeros, and weather "condition" labels (clear, rain, snow) were programmatically assigned using domain thresholds. Date columns were parsed and standardized to ensure accurate temporal alignment with simulated orders. The final weather datasets included only the essential variables (temperature, precipitation, snow, and derived weather condition flags) and were

[3] https://meteostat.net/en/place/it/rome?s=16235&t=2021-03-16/2025-04-23

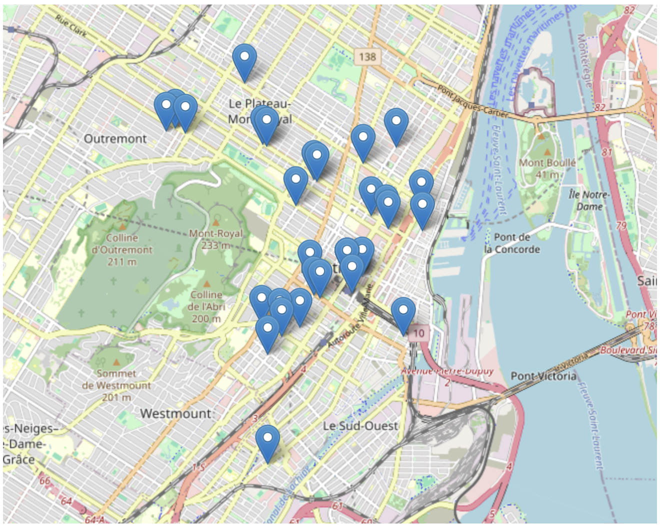
[4] https://montreal.weatherstats.ca/download.html

stored in a harmonized format across both cities.

Finally, all datasets were assigned unique identifiers, and city/neighborhood fields were added where relevant. This multi-layered preprocessing ensured that the simulation environment reflected the real geographic, operational, and meteorological conditions in each city, while also providing a clean, reproducible foundation for empirical modeling and benchmarking.



(a) Restaurant locations in central Rome



(b) Restaurant locations in Montreal

Figure 10: Spatial distribution of sampled restaurant locations used in simulation for (a) Rome and (b) Montreal.

3.4.4 Limitations

While these datasets provide a robust empirical foundation, it is acknowledged that they represent proxies for general Q-commerce activity and may not capture all nuances of instant delivery operations (such as dark store networks or non-restaurant fulfillment nodes). However, the high frequency, granularity, and spatial coverage of restaurant delivery data make it a suitable and practical testbed for last-mile courier assignment research, particularly given the current limitations in publicly available Q-commerce datasets.

3.5 Simulation Environment

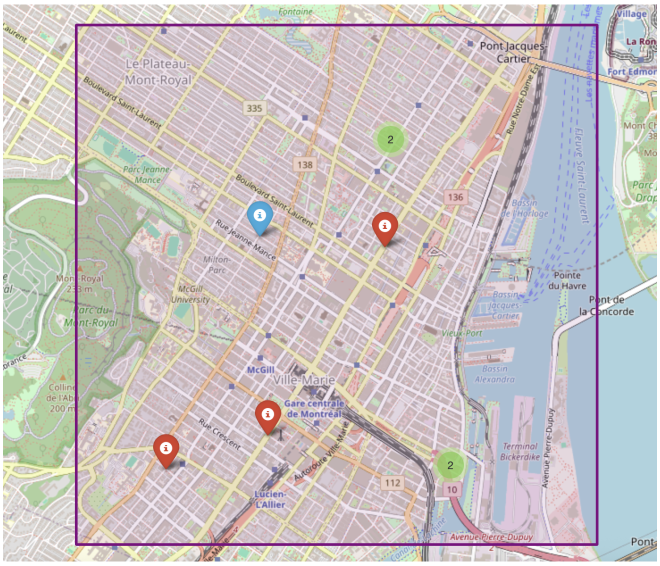
The simulation environment was designed to closely replicate operational realities in urban last-mile delivery, leveraging both spatial network data and empirical parameters from real-world platforms. The framework is modular, enabling flexible benchmarking of assignment strategies across a variety of scenarios. All stochastic elements (e.g., order timing, courier assignment, delivery locations) were controlled using fixed random seeds to ensure reproducibility of simulation results across multiple runs. Simulation parameters such as the number of couriers, number of daily orders, and demand multipliers were set to reflect realistic operating conditions based on platform norms and empirical data where available.

3.5.1 Urban Network Representation

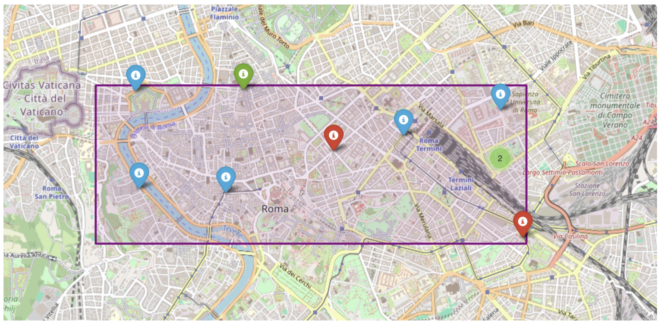
The street networks of Rome and Montreal were modeled using OpenStreetMap (OSM) data, imported via the OSMnx library. For each city, a bounding box encompassing the high-density urban core was defined, and the corresponding drivable road graph was extracted. These graphs serve as the geographic substrate for all agent movements, distance calculations, and travel-time estimates within the simulation.

3.5.2 Courier Generation

Couriers are represented as agent objects initialized at the beginning of each simulation day. For both cities, courier starting locations are randomly assigned to nodes within the street graph, ensuring realistic spatial dispersion. In Montreal, bicycle couriers are further restricted to downtown bounds to reflect real operational patterns. Vehicle type distributions reflect platform and city-specific norms, with Rome having a higher prevalence of bikes and scooters, and Montreal skewed toward cars. Each courier is assigned a capacity based on vehicle type (bike/scooter: 1; car: 2), and is parameterized by dynamic attributes such as status, location, total distance traveled, and operational availability. See Figure 11a for Montreal and Figure 11b for Rome for a visualization of courier starting locations in each urban environment. Furthermore, although the agent objects are designed to resemble real couriers, the simulation assumes that these agents do not experience fatigue or require breaks. As a result, the same set of 10 couriers can be continuously monitored and dispatched throughout the entire simulation period.



(a) Initial courier distribution in Montreal.



(b) Initial courier distribution in Rome.

Figure 11: Sampled starting locations of couriers for simulation in (a) Montreal and (b) Rome. Downtown areas are bounded for bikes in Montreal and high-density neighborhoods in Rome.

3.5.3 Order Generation

Order demand is synthetically generated for each simulation day. Pickup locations are sampled from the geocoded set of filtered restaurants, while delivery locations are randomly sampled from the spatial bounds of the city, projected onto the road network. Order times are allocated using a fixed time-of-day distribution that replicates empirical demand peaks (morning, lunch, evening). Order sizes and preparation times are randomly assigned based on realistic probability distributions. Weather-dependent and temporal multipliers are applied to baseline order counts to simulate surges during adverse weather or high-demand periods. An example of the generated order data is shown in Figure 12, illustrating the structure and key operational attributes assigned during the order simulation process.

order_id	pickup_lat	pickup_lon	delivery_lat	delivery_lon	order_time	period	prep_time_min	order_size_label	order_size
montreal_order_0	45.513585	-73.5729521	45.4959059	-73.5603228	11:27:00	lunch	12	small	0.25
montreal_order_1	45.5018869	-73.56739190000000	45.4953787	-73.5608843	11:40:00	lunch	7	medium	0.5
montreal_order_2	45.517529	-73.58109810000000	45.4918793	-73.5815284	10:04:00	morning	5	medium	0.5
montreal_order_3	45.4961841	-73.5818934	45.5229521	-73.5689431	11:37:00	lunch	12	small	0.25
montreal_order_4	45.5018869	-73.56739190000000	45.5304764	-73.589608	10:37:00	morning	10	medium	0.5
montreal_order_5	45.5018869	-73.56739190000000	45.5045436	-73.5819554	10:46:00	morning	15	small	0.25
montreal_order_6	45.4794972	-73.58089440000000	45.4944753	-73.5565797	10:51:00	morning	13	medium	0.5
montreal_order_7	45.50197180000000	-73.5648776	45.5127653	-73.580783	11:39:00	lunch	17	small	0.25
montreal_order_8	45.4996912	-73.57301100000000	45.5317974	-73.5652797	10:40:00	morning	16	medium	0.5
montreal_order_9	45.5018869	-73.56739190000000	45.493347	-73.5633679	11:36:00	lunch	14	large	0.8
montreal_order_10	45.4994033	-73.57194250000000	45.501213	-73.5859558	11:56:00	lunch	13	small	0.25

Figure 12: Example of generated order data for Montreal, illustrating simulated pickup and delivery locations, order timing, size, and other operational attributes.

3.5.4 Travel Time and Distance Calculation

Travel times and distances between all origin-destination pairs are estimated using two complementary methods, depending on the experimental scenario.

For experiments involving realistic routing, travel times are obtained by querying the Google Maps Distance Matrix API. This provides high-fidelity estimates that account for actual street topology, routing constraints, and real-time traffic conditions. For each unique origin-destination pair and vehicle mode (bike, scooter, or car), the API returns both travel time (in minutes) and route distance (in kilometers).

However, due to the privacy restrictions associated with Google Maps data, as well as the significant financial and operational costs incurred when scaling up API requests, a parallel set of simulations is performed using only the Haversine formula to estimate straight-line (great-circle) distances. Constant average speeds are assigned to each vehicle type (15 km/h for bikes, 30 km/h for scooters, and 40 km/h for cars), from which travel times are computed. This approach enables comprehensive benchmarking and reproducibility without reliance on proprietary or rate-limited external services.

To maximize efficiency, all queried travel times and distances are cached using unique route keys, reducing redundant API calls. If the API is unavailable or fails after several retries, the system automatically falls back to the Haversine-based estimate. This hybrid approach ensures robust simulation under both high-fidelity and resource-constrained settings.

All travel time and distance calculations are encapsulated within modular functions, facilitating straightforward switching between precise and approximate routing as required by each simulation mode.

3.5.5 Courier Status and Event Scheduling

Courier behavior is tracked and updated at each discrete time step (typically one minute). States include idle, active, in transit, at pickup, or at delivery. Courier capacity, working hours, and assigned order lists are dynamically updated based on simulation events (e.g., assignment, pickup, delivery, or break). Shift patterns and breaks are simulated to mirror on-demand workforce practices, and operational metrics such as active minutes, idle minutes, and total earnings are continuously tracked.

3.5.6 Payment and Incentives

Courier compensation was modeled to reflect city-specific labor market norms and platform practices. For Rome, the simulated pay structure was based on the Rider National Collective Labor Agreement (CCNL), which sets a gross minimum wage of €10 per hour (equivalent to €0.167 per minute), supplemented as necessary to guarantee the contractual minimum for all active delivery time[5]. In Montreal, given the lack of detailed public documentation for local food delivery platforms, it was assumed that labor and compensation conditions were broadly analogous to those recently established in the United States for gig-economy delivery workers (such as Uber Eats)[6]. The model applies a guaranteed minimum hourly wage, a per-mile compensation for vehicle usage, and incorporates dynamic "trip supplements" for adverse weather, high demand, or limited courier supply[7]. These assumptions enable the exploration of how economic incentives might affect courier availability and platform service levels, while aligning the simulation with real-world pay structures.

3.5.7 Simulation Loop and Data Logging

The simulation advances in one-minute increments, processing order arrivals, courier updates, assignment decisions, and delivery events at each step. All relevant system states, i.e., order status, courier metrics, assignment outcomes, and travel logs, are recorded for downstream analysis. The modular structure supports the integration and evaluation of multiple assignment algorithms, as detailed in the following section.

This simulation setup provides a realistic and extensible foundation for evaluating the performance and adaptability of various courier assignment strategies under diverse operational and environmental conditions.

3.5.8 Limitations of the Simulation Approach

While the simulation environment incorporates realistic spatial, temporal, and behavioral dynamics, several limitations are acknowledged. The urban layouts are based on OSM data and may not capture all micro-level routing constraints, such as temporary road closures or delivery access restrictions. Behavioral models for couriers are simplified and do not account for all real-world factors, including shift preferences, multi-app activity, or personal incentives

[5] https://delivery.glovoapp.com/it/frequently-asked-questions-about-the-new-change

[6] https://help.uber.com/en/driving-and-delivering/article/know-more-about-delivery-fees?nodeId=5aecf430-8e00-4608-ba0a-8bba5b104023

[7] https://help.uber.com/driving-and-delivering/article/prop-22-mileage-payments?nodeId=328b7da3-7152-49b0-8132-27832b3b06d3

beyond those modeled. In addition, reliance on open-source restaurant and weather data, as proxies for Q-commerce operations, may omit certain complexities of instant delivery platforms such as dark store fulfillment or true real-time demand surges. Despite these limitations, the framework provides a robust and extensible basis for controlled benchmarking of assignment algorithms.

3.6 Synthetic Data Generation for Model Training

For machine learning-based assignment strategies, particularly those involving ANFIS and hybrid models, a critical pre-simulation step involves the creation of synthetic training datasets tailored to the operational context of each city and experimental scenario. This process is necessary because real-world order-courier interaction data is either unavailable or insufficiently detailed for supervised learning at scale.

The synthetic data generation process proceeds as follows:

- **Order and Courier Sampling:** Orders and couriers are generated according to empirically grounded distributions that reflect the temporal, spatial, and operational heterogeneity observed in real-world Q-commerce platforms. Features include order size, preparation time, pickup and delivery locations, courier vehicle type, and courier availability.
- **Feature Engineering:** For each potential courier-order pair, relevant features are computed, including travel time estimates (using the Haversine distance and city-specific vehicle speeds, or Google Maps API for the API-based scenarios), current workload, vehicle attributes, weather conditions, and local/global demand indicators.
- **Ground Truth Label Generation:** For each simulated order-courier pool, the ground truth delivery time label is defined as the minimum achievable delivery time across all eligible couriers. This value is computed as the sum of (i) travel time from the courier to the order's pickup location (using the Haversine function), (ii) the greater of order preparation time or courier's travel time to pickup (whichever is longer), and (iii) travel time from pickup to delivery location. This label represents the system-optimal assignment outcome under the given operational conditions.
- **Noise Injection (Haversine/Batched Models):** To improve model generalization and mitigate overfitting, controlled Gaussian noise is added to ground truth labels in haversine-based and batching scenarios. This function perturbs the ground truth delivery time by adding zero-mean Gaussian noise with a standard deviation proportional to the value, simulating real-world variability and helping the model learn smoother decision boundaries.
- **Scenario Diversity:** To ensure models generalize to real-world variability, datasets are constructed to span a broad spectrum of operational scenarios. This includes systematically varying:
  - *Order volumes:* Low, medium, and high demand settings to reflect both routine and peak periods.
  - *Weather conditions:* Sampling from clear, rain, and snow events to model their impact on delivery operations and travel times.
  - *Time of day:* Generating orders across morning, lunch, evening, and late-night periods to capture temporal

patterns in demand and traffic.

  - *Courier fleet composition:* Including a mix of vehicle types, capacities, and initial locations to simulate diverse urban fleets.
  - *Geographical spread:* Dispersing pickup and delivery locations across the city map, ensuring coverage of central, peripheral, and edge-case neighborhoods.
  - *Demand surges and outliers:* Injecting rare high-demand or severe-weather days to expose models to challenging edge cases.

By explicitly sampling across these dimensions, the synthetic datasets support training and evaluation under a wide variety of realistic urban delivery conditions, fostering robust model performance.

The resulting synthetic datasets are used to train and validate machine learning models prior to their deployment in the simulation engine. This approach enables controlled experimentation with different model architectures, feature sets, and assignment strategies, while avoiding the privacy, cost, and scalability limitations inherent in proprietary or real-world delivery data.

3.7 Assignment Strategies

A variety of courier assignment strategies were implemented and benchmarked in the simulation environment, reflecting both classical operations research methods and recent advances in data-driven optimization. Each method was evaluated on its efficiency, adaptability, and suitability for real-time last-mile delivery operations.

3.7.1 Experimental Evolution and Rationale

The methodological development of assignment strategies in this study was shaped by both empirical findings and practical constraints encountered during experimentation. Initially, all assignment algorithms, including the machine learning-based ANFIS models, were evaluated using travel times and distances derived from the Google Maps API. While classical optimization algorithms do not require training, the machine learning models were benchmarked on API-derived metrics to provide a realistic assessment of last-mile delivery conditions. However, commercial API usage constraints and data privacy considerations made it infeasible to generate large-scale labeled datasets or to conduct extensive ML model training under these conditions.

As a result, the experimental protocol was adapted and refined through several distinct modes:

- **API Mode:** All assignment algorithms were evaluated using Google Maps API travel times and distances. This provided the most realistic benchmarking environment but was limited to smaller-scale experiments due to API access and cost constraints.
- **Haversine Mode:** Machine learning models were trained and evaluated entirely using travel times and distances computed with the Haversine formula. Classical optimization algorithms were also evaluated in this mode for direct comparison. This approach ensured consistency between training and evaluation, as well as scalability

for large-scale experiments.

- **Batching Mode (ML only):** Once the effectiveness of the machine learning-based assignment strategies was validated under Haversine-based conditions, experimentation was extended to include batching scenarios, where multiple orders could be assigned to a single courier. Since batching logic is not feasible within classical optimization algorithms such as the Hungarian method, these experiments focused exclusively on machine learning strategies (ANFIS and hybrid models), using Haversine-derived features for both training and evaluation.

Throughout this sequence of experimental iterations, it was observed that the Hungarian algorithm often outperformed the original ANFIS models in API-based scenarios. This finding motivated the development of a hybrid assignment approach that integrates the system-wide optimality of the Hungarian algorithm with the context-aware adaptability of ANFIS. The hybrid model was introduced to leverage the respective strengths of both paradigms and to provide a more robust assignment solution.

This sequence of methodological adaptations ensured that each assignment strategy was evaluated under consistent and realistic assumptions, with training and evaluation data aligned to the scenario under study. The resulting protocol not only improves the validity of comparative benchmarking, but also provides actionable insights into the operational trade-offs between classical and data-driven assignment approaches in last-mile delivery logistics.

3.7.2 Naive Assignment (Greedy Nearest-Neighbor)

Overview

The naive assignment strategy serves as a baseline heuristic for courier-order matching, reflecting the simplest real-time decision logic commonly used in operational settings. At each simulation time step, new orders are assigned to the nearest available courier who is idle, present within the service area, and possesses sufficient capacity to fulfill the order. This approach operates in a strictly myopic, one-to-one fashion, with no consideration of batching or system-wide optimization.

Algorithm

For each unassigned order, the set of eligible couriers is filtered based on three criteria:

- **Idle status**
- **Current availability at or before the present simulation minute**
- **Available capacity greater than or equal to the order's size**

Among eligible couriers, proximity to the pickup location is calculated using the Haversine (great-circle) distance between the courier's current coordinates and the order's pickup point. The closest courier is then selected for assignment.

Estimated travel times and distances for both the pickup leg (courier to restaurant) and the delivery leg (restaurant to customer) are computed using one of two methods, depending on the experimental mode. In the first simulation, the `get_realistic_travel_time` function is used, which incorporates network-based travel times and mode-specific speed adjustments by querying the Google Maps Distance Matrix API, thereby capturing the effects of actual road networks. In the second simulation, a purely geometric approach is employed: travel times and distances are estimated using the `travel_time_haversine` function, which computes great-circle (Haversine) distances between locations and divides by a fixed average speed specific to each vehicle type (15 km/h for bikes, 30 km/h for scooters, and 40 km/h for cars).

In both modes, after assignment, the courier's operational state (including location, available capacity, next available time, active minutes, and cumulative distance) is updated accordingly. Order attributes are also revised to reflect assignment, estimated waiting time, and the assigned courier's identifier.

The step-by-step assignment procedure for the naive, nearest-neighbor approach is detailed in Algorithm 1.

Implementation Details

This method is implemented as a loop over all new orders within each simulation time step. Once assigned, a courier is removed from the pool of available couriers for that minute, ensuring that each courier is only matched to a single order per event. No global optimization or batching is performed; each order is handled in isolation, considering only immediate courier availability.

Travel time and distance calculations are performed according to the simulation mode: in API-based simulations, the `get_realistic_travel_time` function queries the Google Maps Distance Matrix API to estimate route times and distances; in Haversine-based simulations, the `travel_time_haversine` function estimates travel using straight-line (great-circle) distances and fixed average speeds by vehicle type. This approach allows for rigorous comparison of assignment strategies under both high-fidelity and efficient, large-scale simulation settings.

Algorithm 1 Naive Greedy Courier Assignment (API and Haversine Modes)

```
for each new order o in new_orders do
    if o is already assigned then
        continue
    end if
    eligible ← all couriers c in available_couriers such that c.status = idle, c.available_at ≤ current_minute, c.available_capacity ≥ o.order_size
    if eligible is empty then
        continue
    end if
    closest ← arg min over c in eligible of distance between c and o.pickup
    if simulation mode is API then
        Estimate travel time and distance to pickup and delivery using get_realistic_travel_time
    else
        Estimate travel time and distance to pickup and delivery using travel_time_haversine
    end if
    Update closest courier's state (active minutes, location, capacity, etc.)
    Mark o as assigned and update assignment details
    Remove closest from available_couriers
end for
```

Strengths and Limitations

The naive assignment approach offers rapid, interpretable, and computationally inexpensive assignment decisions, making it suitable for real-time environments with moderate order volumes. However, it neglects system-wide efficiency, courier workload balancing, batching opportunities, and anticipation of future demand. This can result in suboptimal utilization, higher operational costs, and potential courier idleness in dense urban or high-frequency demand scenarios.

3.7.3 Hungarian Algorithm (Minimum-Cost Matching)

Overview

The Hungarian algorithm (also known as the Kuhn-Munkres algorithm) is a classical combinatorial optimization technique used to solve the assignment problem, matching agents (couriers) to tasks (orders) to minimize total cost. In this simulation framework, the algorithm is invoked at each assignment event to allocate couriers to new orders, subject to vehicle capacity constraints, with the objective of minimizing the sum of pickup times and order waiting times. The approach yields an optimal, system-wide solution for one-to-one courier-order matching at each discrete event and serves as a high-performance benchmark for other assignment strategies. Batching is not considered in this model.

Algorithm

At each decision point, a cost matrix is constructed, where each row represents an available courier and each column represents a new order. The entry (i, j) encodes the expected total service time if courier i is assigned to order j, combining the estimated pickup time (courier to pickup location) and any necessary wait time (if the order's preparation time exceeds the courier's travel time). Assignments that are infeasible, due to insufficient courier capacity, are assigned a large dummy cost to exclude them from optimal matching. The linear sum assignment problem is then solved using the Hungarian algorithm, implemented via the SciPy library, yielding a set of optimal courier-order pairs that minimize overall cost. The assignment logic for optimal courier-order matching, including construction of the cost matrix and application of the linear sum assignment algorithm, is presented in Algorithm 2.

Mathematical Formulation

The assignment problem is solved by minimizing the total cost:

$$\min_{x_{ij} \in \{0,1\}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{1}$$

subject to

$$\sum_{j=1}^{n} x_{ij} \leq 1 \quad \forall i, \tag{2}$$

$$\sum_{i=1}^{m} x_{ij} \leq 1 \quad \forall j, \tag{3}$$

$$x_{ij} = 0 \quad \text{if the capacity constraint is violated} \tag{4}$$

where c_{ij} represents the cost (i.e., pickup time plus wait time) of assigning courier i to order j, and $x_{ij} = 1$ if courier i is assigned to order j, 0 otherwise.

Implementation Details

The algorithm operates as follows:

- Construct a cost matrix for all feasible courier-order pairs, where cost is the sum of travel time and (if applicable) wait time.
- **Simulation Mode:** In API-based simulations, travel times and distances are estimated using the `get_realistic_travel` function, which queries the Google Maps Distance Matrix API and accounts for realistic routing. In Haversine-based simulations, travel is computed using the `travel_time_haversine` function, based on straight-line (great-circle) distances and average speeds by vehicle type.
- Solve the assignment problem with the `linear_sum_assignment` function from SciPy, extracting optimal matches.
- For each valid match, update courier and order states, thereby setting locations, times, active minutes, capacities, and assignment flags accordingly.
- Remove matched couriers from the available pool for the current event to avoid double assignment.

This dual-mode structure enables benchmarking under both high-fidelity (API-based) and large-scale, reproducible (Haversine-based) conditions.

Strengths and Limitations

The Hungarian algorithm guarantees an optimal assignment of couriers to orders at each decision step, balancing workloads and minimizing immediate operational delays. However, it does not account for future demand, multi-order batching, or dynamic order arrival beyond the current event window. Its computational complexity increases with the number of couriers and orders, but remains tractable for batch sizes typical of real-world delivery platforms.

Algorithm 2 Hungarian Assignment for Courier-Order Matching (API and Haversine Modes)

```
if no available couriers or no new orders then
    return
end if
Initialize C as cost matrix of size [couriers × orders], filled with large default cost
for each courier i do
    for each order j do
        if courier i capacity < order j size then
            continue
        end if
        if simulation mode is API then
            Compute pickup time and wait time using get_realistic_travel_time
        else
            Compute pickup time and wait time using travel_time_haversine
        end if
        Set C[i, j] to pickup time + wait time
    end for
end for
Solve assignment: row_ind, col_ind = linear_sum_assignment(C)
for each valid (i, j) pair do
    Update courier and order states (location, capacity, times, assignment flag)
    Remove courier from available pool
end for
```

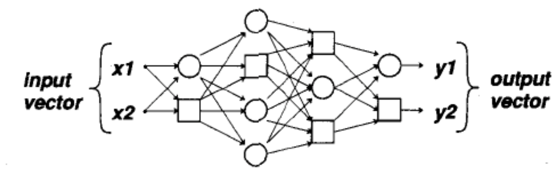
3.7.4 Adaptive Neuro-Fuzzy Inference System (ANFIS) Assignment

Overview and Motivation

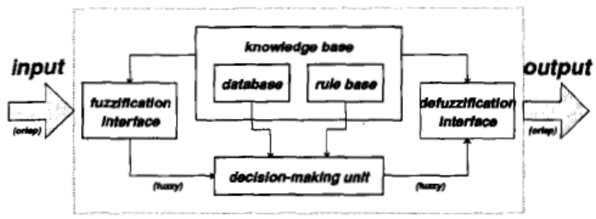
The Adaptive Neuro-Fuzzy Inference System (ANFIS), first formalized by Jang [Jang, 1993], is a neuro-symbolic architecture that blends the learning capabilities of neural networks with the interpretability and qualitative reasoning of fuzzy inference systems. In the context of last-mile courier assignment, the operational environment is characterized by uncertainty, fluctuating demand, heterogeneity in courier and order attributes, and competing business objectives. These challenges demand assignment strategies that are both data-driven and capable of incorporating expert knowledge or operational heuristics.

ANFIS offers a unique solution in this setting by combining the flexibility of machine learning with the transparency of rule-based systems. Unlike classical optimization algorithms, which may be rigid and limited in handling real-time context, or traditional “black-box” neural networks, which can lack interpretability, ANFIS models can capture subtle, context-dependent trade-offs (e.g., between speed, cost, and workload) while providing transparent, human-readable rules that aid in diagnosis and improvement.

In this study, ANFIS is used as a regression-based decision module: at each assignment event, the system predicts the expected delivery latency for every eligible courier-order pair, based on a high-dimensional, context-rich feature set. The courier with the lowest predicted latency is assigned the order, enabling context-sensitive, adaptive optimization that incorporates both operational data and domain insights. This hybrid reasoning makes ANFIS particularly well-suited for the complex, fast-changing landscape of last-mile quick commerce logistics.



(a) ANFIS layered architecture (neuro-fuzzy network).



(b) Block diagram of a fuzzy inference system (FIS).

Figure 13: (a) Schematic of an ANFIS model, combining neural and fuzzy inference components; (b) Conceptual diagram of a fuzzy inference system, highlighting the flow from input fuzzification to output defuzzification. ANFIS extends the FIS framework by enabling data-driven learning of rules and membership functions.

Implementation Details

This study employs a modern implementation of ANFIS (using the `BioAnfisRegressor` from the `xanfis` library), which applies evolutionary algorithms to train model parameters. The ANFIS model receives a set of features describing each courier-order pair: travel times (pickup-to-courier and pickup-to-delivery), courier workload, vehicle type, weather, order preparation time, time-of-day, available capacity, order size, and both local and global demand forecasts. The model is trained on synthetic data specific to each city.

- **API Mode:** In initial experiments, ANFIS models were trained and evaluated with a high-dimensional feature set (not pairwise) derived from Google Maps API travel times and operational context. Due to the large number of features, Principal Component Analysis (PCA) was applied for dimensionality reduction prior to model fitting.
- **Haversine Pairwise Mode:** In subsequent experiments, the protocol was revised to use a smaller, pairwise feature set based on haversine-derived travel times and operational attributes for each courier-order pair, allowing for direct matching of couriers and orders. Here, PCA was omitted due to the reduced dimensionality.

The number and type of membership functions (such as sigmoid or Gaussian) and the number of fuzzy rules were tuned as hyperparameters, with the final API-based models using 15 rules, sigmoid membership functions, and genetic algorithm optimization. The BioAnfisRegressor leverages a Genetic Algorithm (GA) for optimization, suitable for the complex, nonconvex parameter space of ANFIS.

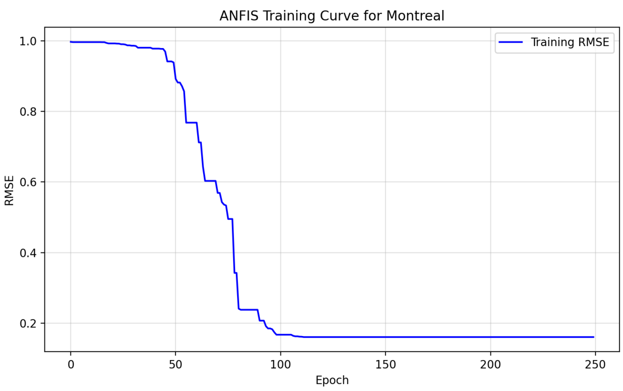
For model evaluation in the API-based phase, the dataset was split into 80% for training and 20% for testing, with all reported predictive metrics (RMSE, MAE, R^2) computed on the held-out test set. In Montreal, the final model yielded a root mean squared error (RMSE) of 0.742 minutes, mean absolute error (MAE) of 0.460 minutes, and an R^2 score of 0.9661. In Rome, the corresponding values were 0.620 minutes (RMSE), 0.323 minutes (MAE), and R^2 of 0.9709.

After the API phase, the experimental protocol switched to haversine-based features and a reduced, pairwise input scheme, enabling a much smaller feature set and rendering PCA unnecessary. In these subsequent experiments, both with and without batching, the predictive performances are displayed in Table 3 for both cities.

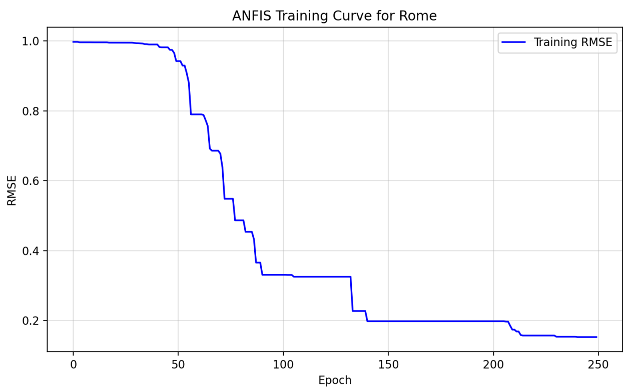
Table 3: ANFIS Model Performance by Mode and City

City	**Mode**	**RMSE (min)**	**MAE (min)**	**R^2**
Montreal	API	0.742	0.460	0.9661
Montreal	Haversine	1.538	1.144	0.9672
Montreal	Batching	1.017	0.740	0.9948
Rome	API	0.620	0.323	0.9709
Rome	Haversine	1.254	0.940	0.9484
Rome	Batching	0.817	0.573	0.9906

Inference is performed in real time during simulation: for each unassigned order, the ANFIS model predicts the expected delivery latency for all eligible couriers, and the courier with the lowest predicted latency is selected. Courier and order states are updated accordingly. Detailed training curves for each city are shown in Figure 14.



(a) ANFIS training curve for Montreal for model with APIs.



(b) ANFIS training curve for Rome for model with APIs.

Figure 14: Convergence of ANFIS model training for each city. Training curves show RMSE over 250 epochs, demonstrating rapid convergence and stable final performance for both Montreal and Rome.

Implementation Note

A minor but essential modification was made to the `CustomANFIS` class in `base_anfis.py` of the `xanfis` library to ensure method accessibility and compatibility with model saving/loading. Specifically, the assignments

```
self._get_strength = self.__get_strength_by_prod
self._get_strength = self.__get_strength_by_mean
self._get_strength = self.__get_strength_by_blend
```

were replaced with:

```
self._get_strength = self._get_strength_by_prod
self._get_strength = self._get_strength_by_mean
self._get_strength = self._get_strength_by_blend
```

This change was required for all ANFIS-based models used in both the standalone ANFIS assignment and the Hybrid Pairwise Assignment strategies. Removing double underscores prevents Python's name mangling, ensuring that the correct method is assigned and accessible during serialization, deserialization, and inference. The modified class supports reliable saving and loading of ANFIS models within the simulation pipeline. The updated version of the library is documented in the project repository.

Strengths and Limitations

ANFIS enables the assignment system to learn complex heuristics directly from data, while retaining the potential for human-in-the-loop tuning and interpretability. Major strengths include adaptability to changing demand and operational contexts, strong performance in batching and context-sensitive logic, and transparency in decision rules. However, limitations include increased computational complexity relative to linear or rule-based models, especially during training and real-time inference, and challenges with scaling to very large courier or order pools without further optimization. Model performance is sensitive to the alignment between training features and evaluation context, as highlighted in the experiments.

The detailed assignment and prediction logic is presented in Algorithm 3.

For a comprehensive overview of the theoretical architecture and learning algorithm, see [Jang, 1993].

Algorithm 3 Courier Assignment via ANFIS Regression

Require: Available couriers, new orders, current minute, weather condition, city, API key, trained ANFIS model, feature scalers, PCA transformer, all orders

Ensure: List of completed orders, updated courier pool

```
Compute weather_code from weather condition
Compute period_of_day from current minute
Initialize completed_orders as empty list
for each order in new orders do
    if order is already assigned or order time is in the future then
        continue
    end if
    Identify eligible couriers (available at current minute, enough capacity)
    if no eligible couriers then
        continue
    end if
    Initialize preds as empty list
    for each eligible courier do
        Extract feature vector for order-courier pair
        Repeat feature vector to construct a 10-courier input if needed
        Apply feature scaling and PCA transformation
        Predict delivery latency using the trained ANFIS model
        Store predicted latency and corresponding courier in preds
    end for
    Select courier with minimum predicted latency from preds
    Compute actual travel time and distance for pickup and delivery legs
    Update selected courier state: location, available time, capacity, active minutes, total distance, assigned orders
    Mark order as assigned, record assigned courier, wait time, etc.
    Append order to completed_orders
    Remove selected courier from available pool
end for
return completed_orders, available_couriers
```

3.7.5 Hybrid Pairwise Assignment: ANFIS-Enhanced Hungarian Matching

Overview and Motivation

The hybrid pairwise assignment strategy combines the global optimality of the Hungarian algorithm with the predictive adaptability of a trained ANFIS regression model. Unlike static cost matrices based on naive distance or time, this method uses a city-specific ANFIS model to generate dynamic, operationally informed cost predictions for each courier-order pair, enabling context-aware matching that responds to real-time delivery conditions.

This approach evolved as a direct response to the limitations observed in both classical (Hungarian) and standalone ANFIS models, especially under varying demand and batching scenarios. The hybrid strategy allows the system to blend classical optimization with learned assignment heuristics, yielding improved flexibility and robustness in simulation.

Methodology

At each assignment event, a cost matrix is built where each entry (i, j) corresponds to the ANFIS-predicted delivery latency for courier j and order i. In North American experiments (e.g., Montreal), this latency is further blended with a modeled operational cost using an adaptive α parameter:

$$\text{score}_{ij} = \alpha \cdot \widehat{L}_{ij} + (1-\alpha) \cdot \widehat{C}_{ij}$$

where $\widehat{L}_{ij}$ is the ANFIS-predicted latency and $\widehat{C}_{ij}$ is the modeled cost (wage, distance, bonus). α is tuned dynamically by context (weather, time, day), reflecting local labor practices. In Rome, assignments are based solely on predicted latency ($\alpha = 1$), in line with local wage structures.

Features for each courier-order pair include pickup and delivery times/distances (all calculated via the haversine method), courier workload, vehicle and capacity, order size, local/global demand, and spatial indicators. After initial experiments with high-dimensional API-based features and PCA, the final model uses a streamlined, pairwise feature vector of 15 dimensions, without PCA, for efficiency and consistency between training and deployment.

The cost matrix populated with these scores is passed to the Hungarian algorithm, which extracts the optimal courier-order assignment for the event.

Implementation Details

The hybrid pairwise ANFIS-Hungarian assignment method is built around several key design choices to maximize both realism and operational flexibility. First, a dedicated ANFIS regression model is trained for each city, using feature vectors that represent individual courier-order pairs. This is a departure from the original ANFIS approach, which pooled candidate couriers in fixed-size batches; the pairwise design allows for finer-grained, context-specific predictions and enables efficient integration with combinatorial optimization.

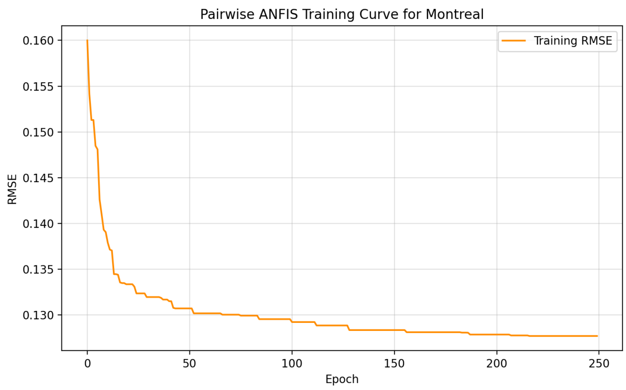
For each assignment event, the ANFIS model receives a 15-dimensional feature vector for every feasible courier-order pair. These features include: pickup and delivery travel times and distances (calculated using the haversine formula), courier workload (active and idle minutes), vehicle type and capacity, available capacity, order size, order preparation time, time-of-day, spatial centrality (distance to city center), and both local and global demand forecasts. No PCA is used in this mode, since the feature space is already compact and interpretable; all features are standardized prior to training and inference.

A key innovation of the hybrid approach is adaptive cost blending, implemented only for the North American (Montreal) scenario. Here, the ANFIS-predicted delivery latency is linearly blended with a modeled operational cost (which accounts for wage, distance, and context-sensitive bonuses) using an α parameter. This parameter is dynamically tuned by real-time conditions such as weather, hour, and day of week, thus mirroring hybrid earning structures and regulatory requirements in platforms like Uber Eats and DoorDash. In the Rome case, where platform compensation is less variable and regulatory floors are common, assignments are based solely on predicted latency ($\alpha = 1$).

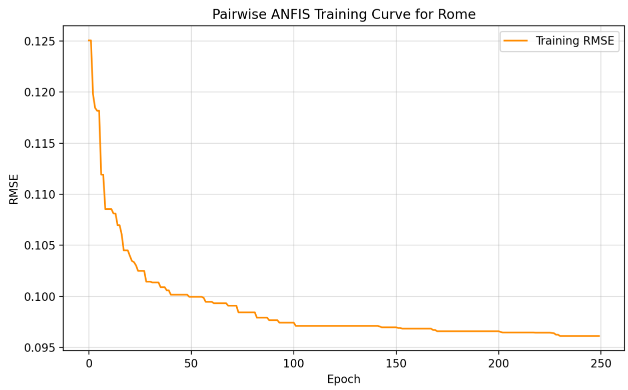
During simulation, for each assignment event:

- The cost matrix is populated with predicted scores for all feasible courier-order pairs using the trained ANFIS model (with cost blending applied in Montreal).
- The Hungarian algorithm (`linear_sum_assignment` from SciPy) is then used to extract an optimal set of assignments, minimizing overall blended cost or latency.
- After assignment, all courier and order states are updated, including location, capacity, availability, active minutes, and delivery counts.

The hybrid model is trained and validated separately for each city. As in other strategies, 80% of the data is used for training and 20% for testing. Across all experimental scenarios, i.e., API-based (with PCA), haversine-based without batching, and haversine-based with batching, the pairwise ANFIS models demonstrated strong predictive accuracy and robust generalization. A summary of evaluation metrics for each mode and city is provided in Table 4, highlighting the model's reliable performance across diverse urban and operational conditions.



(a) Pairwise ANFIS training curve for Montreal.



(b) Pairwise ANFIS training curve for Rome.

Figure 15: Convergence of the pairwise ANFIS model training for each city. Training curves show RMSE over 250 epochs, demonstrating rapid convergence and stable final performance for both Montreal and Rome.

The training curves shown in Figure 15 illustrate the convergence behavior of the pairwise ANFIS models for both Montreal and Rome. In each case, the root mean squared error (RMSE) decreases rapidly over the initial epochs, followed by a steady plateau as the model parameters are refined. This pattern indicates that the hybrid ANFIS-Hungarian approach is able to effectively learn from the synthetic training data and adapt to the underlying operational complexity of each city. The relatively low and stable final RMSE in both cases reflects robust model fit and suggests that the chosen feature set, learning algorithm, and cost blending strategies are well-suited to the last-mile courier assignment problem. Together with the summary statistics presented in Table 4, these results demonstrate both rapid convergence and strong generalization, confirming the practical value of the pairwise ANFIS approach under diverse simulation conditions.

Table 4: Summary of Pairwise ANFIS Model Evaluation Metrics Across Experimental Modes (held-out test set)

Mode	City	RMSE (min)	MAE (min)	R^2 Score
API, no batching (with PCA)	Montreal	0.742	0.460	0.9661
API, no batching (with PCA)	Rome	0.620	0.323	0.9709
Haversine, no batching	Montreal	0.666	0.498	0.9891
Haversine, no batching	Rome	0.612	0.454	0.9884
Haversine, batching	Montreal	1.082	0.859	0.9797
Haversine, batching	Rome	1.110	0.878	0.9699

Implementation Note

A minor but essential modification was made to the `CustomANFIS` class in `base_anfis.py` of the `xanfis` library to ensure method accessibility and compatibility with model saving/loading. Specifically, the assignments

```
self._get_strength = self.__get_strength_by_prod
self._get_strength = self.__get_strength_by_mean
self._get_strength = self.__get_strength_by_blend
```

were replaced with:

```
self._get_strength = self._get_strength_by_prod
self._get_strength = self._get_strength_by_mean
self._get_strength = self._get_strength_by_blend
```

This change was required for all ANFIS-based models used in both the standalone ANFIS assignment and the Hybrid Pairwise Assignment strategies. Removing double underscores prevents Python's name mangling, ensuring that the correct method is assigned and accessible during serialization, deserialization, and inference. The modified class supports reliable saving and loading of ANFIS models within the simulation pipeline. The updated version of the library is documented in the project repository.

Strengths and Limitations

This hybrid strategy combines the strengths of combinatorial optimization and machine learning, adapting in real time to changing urban conditions, regulatory contexts, and operational priorities. Its main limitation remains computational cost, particularly as the pool of candidate assignments grows, and dependence on the quality and coverage of the training data.

The main assignment logic is outlined in Algorithm 4.

Algorithm 4 Hybrid ANFIS-Hungarian Pairwise Assignment

```
for each unassigned order at current event do
    for each eligible courier do
        Extract 15-dimensional feature vector for (o,c)
        Scale and transform features (standardization)
        Predict delivery latency with trained ANFIS model
        if city is Montreal then
            Blend latency and cost with α parameter
        end if
        Assign predicted score to cost matrix entry (o,c)
    end for
end for
Apply Hungarian algorithm to cost matrix for optimal assignment
for each matched order-courier pair do
    Update states (location, availability, delivery count, etc.)
end for
```

3.8 Evaluation Metrics

The evaluation of assignment strategies in this study is grounded in a set of operational metrics commonly used in last-mile delivery and Q-commerce literature. Given the proof-of-concept nature of this simulation framework, all orders are enforced to be delivered in each scenario. This design choice ensures that any observed differences in performance can be directly attributed to the assignment algorithm itself, rather than to confounding effects from dropped or rejected orders. As a result, service rate and drop rate metrics are omitted, and focus is placed on efficiency, cost, and workload distribution.

The core evaluation metrics are as follows:

- **Average Delivery Latency:** The mean (and standard deviation) of time in minutes from order creation to successful delivery, aggregated across all orders.
- **Average Wait Time:** The mean (and standard deviation) of waiting time in minutes experienced by orders before pickup, reflecting both preparation delays and assignment lag.
- **Operational Cost:** The total and average simulated cost of courier operations (e.g., earnings, bonuses, mileage compensation), calculated per courier and per order according to the pay models described in Section 3.2.
- **Courier Utilization:** For each courier, statistics on active minutes (time spent delivering), idle minutes (time available but not delivering), total distance traveled, and number of completed deliveries.
- **Workload Distribution:** Standard deviation and, where relevant, coefficient of variation in courier workloads (e.g., deliveries completed, active time), providing a proxy for fairness and balancing.

All reported metrics are computed as means (and standard deviations, where appropriate) across all orders and couriers in the simulation. In addition to aggregate results, per-order and per-courier data are retained and analyzed to provide deeper insight into the distributional properties of each strategy, such as histograms of delivery time, boxplots of wait time, and analysis of tail events (e.g., fraction of orders delivered within a given time threshold). This dual approach enables both direct algorithm comparison and a nuanced understanding of assignment impacts under diverse operational conditions.

For each scenario (city, algorithm, date), results are summarized in aggregate tables and visualized via comparative plots in the following chapter. Where relevant, per-order analyses are presented to highlight outliers, tail risks, or fairness considerations.

3.9 Experimental Protocol

To ensure fair, transparent, and reproducible benchmarking of all assignment strategies, this study adopts a standardized experimental protocol for simulation and evaluation. Each experiment is defined by a fixed scenario, comprising city (Rome or Montreal), simulation date, and assignment algorithm. The protocol is designed to minimize confounding factors and enable direct comparison of system performance across methods and operational contexts.

- **Scenario Selection**: For each city, a set of representative dates was selected from the full data range (2021-03-16 to 2025-04-23), capturing variability in demand, weather, and operational conditions. Unless otherwise noted (e.g., in multi-day batch experiments), each scenario corresponds to a single simulation day, with identical order and courier pools for all assignment methods.
- **Initialization and Inputs**: At the start of each run, the simulation engine initializes the urban network, order pool, courier agents, and relevant weather data for the selected date and city. All assignment methods operate on the same inputs within each scenario, ensuring comparability.
- **Randomness and Reproducibility**: Where stochasticity is present (e.g., random assignment of starting courier positions, random order times), random seeds are set before scenario initialization and consistently applied across runs to guarantee that all strategies are evaluated on identical data realizations.
- **Assignment Methods**: Each assignment strategy, i.e., naive, Hungarian, ANFIS, and hybrid pairwise, is executed as an interchangeable module within the simulation loop. Assignment events are triggered in discrete one-minute increments, with each method applying its respective logic to match couriers and orders at each step.
- **Data Logging**: At every time step, key system states and agent attributes (order status, courier workload, assignment details, travel logs, etc.) are logged in structured formats (CSV/JSON) for downstream analysis. Upon completion of each simulation run, summary statistics and output files (order/courier logs, scenario summaries) are saved for further aggregation and visualization.
- **Result Aggregation**: For each city and date, performance metrics (delivery latency, operational cost, courier utilization, etc.) are computed and compared across assignment methods. While the primary protocol involves

single runs per date, the simulation engine supports batch runs and averaging over multiple dates for robustness, if desired.

This protocol enables systematic and controlled evaluation of assignment algorithms under a variety of realistic urban logistics scenarios, ensuring that observed performance differences are attributable solely to the assignment strategy, not to random variation in input data or simulation configuration.

All code and data required to reproduce these results are available in the project repository: `https://github.com/DavidPaquette99/From-Data-to-Delivery-Data-Driven-Engineering-of-Last-Mile-Optimization-for-Quick-Commerce-in-Rome-`

3.10 Summary

This chapter has introduced a comprehensive methodological framework for benchmarking courier assignment strategies in last-mile Q-commerce delivery, integrating classical optimization techniques with interpretable machine learning approaches. The proposed pipeline encompasses empirical data collection and preprocessing, agent-based simulation of urban logistics, and the implementation and evaluation of both classical (Naive, Hungarian) and machine learning-enhanced (ANFIS, Hybrid) assignment modules.

Key components include:

- Use of high-frequency, spatially detailed restaurant and weather datasets from two distinct urban environments (Rome and Montreal), providing a realistic and transferable testbed for simulation.
- Modular simulation architecture enabling flexible experimentation with assignment algorithms under controlled, reproducible conditions.
- Development and integration of interpretable ML models (ANFIS, Hybrid) trained on synthetically generated courier-order assignment data, supporting adaptive, context-aware decision making.
- A transparent and standardized experimental protocol, ensuring fair comparisons across methods, cities, and scenarios.
- Evaluation metrics grounded in real-world operational priorities, focusing on efficiency, cost, and fairness, and supported by both aggregate and per-order analyses.

Together, these methodological choices ensure that the results presented in the following chapter are robust, interpretable, and attributable to the assignment logic rather than to artifacts of data processing or simulation design. The next chapter presents the empirical findings and comparative analysis arising from this framework, offering insights for both research and practical deployment of last-mile delivery systems.

4 Results

This chapter presents the empirical findings from the simulation-based benchmarking of courier assignment strategies in last-mile delivery for quick commerce platforms. Building upon the methodological framework established in previous chapters, we systematically evaluate the performance of each assignment algorithm, ranging from classical heuristics to machine learning-enhanced and batching-enabled approaches, across a series of controlled, realistic urban delivery scenarios in both Montreal and Rome.

The analysis centers on core performance metrics that are critical in last-mile operations: average delivery latency, courier utilization, operational costs, and workload distribution. By examining outcomes across diverse demand patterns, weather conditions, and urban layouts, we aim to surface not only the strengths and limitations of each assignment approach but also the nuanced trade-offs inherent to dynamic, real-world logistics environments. Where relevant, findings are contextualized with reference to the broader literature, connecting simulation outcomes to both theoretical and practical developments in last-mile optimization.

The chapter is organized as follows. Section 4.1 introduces the simulation scenarios, experimental design, and summary of evaluation metrics. Section 4.2 presents the quantitative results for each assignment strategy in both urban contexts. Section 4.3 explores scenario-specific insights, robustness checks, and sensitivity analyses. Section 4.4 provides an integrated synthesis and interpretation of the results, highlighting their implications for urban logistics and the operational realities of Q-commerce platforms.

4.1 Experimental Setup and Evaluation Metrics

This section provides an overview of the experimental scenarios, parameter choices, and evaluation criteria used in the comparative analysis of courier assignment strategies. All simulations were conducted using the modular framework described in Chapter 3, ensuring that each assignment method was assessed under consistent operational conditions and with full reproducibility.

4.1.1 Summary of Key Simulation Parameters

Table 5 summarizes the principal scenario parameters for both Montreal and Rome. Travel time calculations differ by assignment strategy: API-based simulations use Google Maps Distance Matrix API with fallback to haversine when needed; all other modes (Haversine, Haversine_Batched) compute times using straight-line (great-circle) distance and mode-specific average speeds.

Note: For API-based experiments, travel times and distances are computed via the Google Maps Distance Matrix API, with haversine fallback in case of API constraints. For all Haversine and Haversine_Batched scenarios, travel time is computed using the great-circle (haversine) distance between points and divided by a mode-specific average speed ($15\,\mathrm{km/h}$ *for bikes,* $25\,\mathrm{km/h}$ *for scooters,* $40\,\mathrm{km/h}$ *for cars), as implemented in* `travel_time_haversine`.

Table 5: Summary of Key Simulation Parameters

Parameter	**Montreal**	**Rome**
Number of couriers	10	10
Base number of orders per day	100	100
Simulation period	2021–2025	2021–2025
Order size distribution	1–2 items	1–2 items
Courier vehicle types	Car, bike, scooter	Bike, scooter, car
Courier capacity (car/bike)	2 / 1	2 / 1
Preparation time (min)	5–20	5–20
Spatial bounds	Real city neighborhoods	Real city neighborhoods
Travel time calculation	API mode: Google Maps API Haversine modes: Haversine distance + avg speed	API mode: Google Maps API Haversine modes: Haversine distance + avg speed
Weather data source	WeatherStats Canada	Meteostat
Assignment interval	1 min	1 min
Batching enabled	Yes (ML/Hybrid only)	Yes (ML/Hybrid only)

4.1.2 Demand Multipliers and Dynamic Order Adjustment

To realistically simulate temporal fluctuations and context-dependent demand surges, order volumes for each simulation day were dynamically scaled by empirically derived multipliers. These demand multipliers reflect observed increases in delivery platform activity during peak times and adverse weather. Table 6 summarizes the values used for Montreal and Rome. Multipliers are compounded multiplicatively when multiple conditions are met (e.g., rainy Saturday evening).

Table 6: Order Demand Multipliers Used in Simulation

Condition	**Montreal Multiplier**	**Rome Multiplier**
Rain (weather)	1.6	1.4
Snow (weather)	1.8	1.5
Saturday (day of week)	1.3	1.3
Friday or Sunday (day)	1.2	1.2
Evening (19:00–22:00), Fri/Sat/Sun	1.6	1.6
Evening (19:00–22:00), Mon–Thu	1.4	1.4

Note: The base number of orders per day is contextually adjusted using these multipliers according to weather, day of week, and hour.

Rationale for Demand Multiplier Selection. The demand multipliers implemented in this study were chosen to reflect empirically observed fluctuations in platform activity during peak periods and under adverse weather, grounded in both industry reports and peer-reviewed literature. For instance, multiple studies and aggregated platform data confirm that food delivery orders surge dramatically during inclement weather (rain and snow) and on weekends or evenings, as consumers seek convenience and shelter from challenging outdoor conditions [Joshi et al., 2022, Joshi et al., 2021]. Specifically, research in major Chinese cities found a 9–13% increase in order volume during heat waves and up to a 20% surge during heavy precipitation or snow events, with similar spikes documented in North America and Europe [Lydersen, 2024, DoorDash, 2024]. These empirical findings justify the elevated weather multipliers for Montreal and Rome, with slightly higher coefficients for Montreal to account for more frequent and severe winter events.

Weekend and evening multipliers are supported by both operational data and industry trends: Saturday evenings, in particular, represent the highest order density, with platforms like DoorDash reporting over 30% increases in order volume during these times [DoorDash, 2024]. The late-night and early-morning surges, especially among younger demographics, reinforce the inclusion of time-specific demand coefficients.

Multipliers are set conservatively to avoid unrealistic amplification, but compounded multiplicatively in scenarios where multiple high-demand factors co-occur (e.g., a rainy Saturday evening), in line with real-world surge pricing and incentive structures [Lydersen, 2024].

In summary, the demand scaling factors are grounded in a synthesis of published research, platform analytics, and simulation best practices, providing both realism and interpretability in the modeling of fluctuating urban delivery demand.

4.1.3 Assignment Strategies

The following courier assignment strategies were evaluated in all simulation scenarios:

- **Naive (Greedy Nearest-Neighbor):** Assigns each incoming order to the geographically closest available courier who meets capacity and status criteria, based on a fast distance metric. This approach serves as a transparent, operational baseline and reflects simple heuristics commonly used in early-stage or resource-constrained platforms.

- **Hungarian Algorithm:** Employs a minimum-cost matching (Kuhn-Munkres) algorithm to optimally assign available couriers to unassigned orders, constructing a cost matrix from the sum of pickup travel time and order wait time for each feasible pair. This yields globally optimal, one-to-one assignments at each decision event, but does not support batching.

- **ANFIS (Adaptive Neuro-Fuzzy Inference System):** Implements a data-driven, interpretable regression model that predicts delivery latency for each feasible courier-order pair. In API-based scenarios, the model leverages high-dimensional contextual features (requiring dimensionality reduction); in haversine and batching scenarios, a compact pairwise feature set is used. ANFIS-based assignment is unique in its support for order batching, allowing a courier to be assigned multiple orders if operationally efficient.[8]

[8] Batching is permitted in the ANFIS and Hybrid strategies, with feasibility checks to ensure that multi-order assignments remain within

- **Hybrid Pairwise (ANFIS-Enhanced Hungarian):** Integrates ANFIS-predicted latencies as dynamic costs in a Hungarian assignment framework. For each event, the cost matrix is populated with either predicted delivery latency (Rome) or a contextually blended score (Montreal, via adaptive α), combining ANFIS latency and modeled operational costs. The Hungarian algorithm then determines the optimal system-wide assignment, balancing both learned and operational objectives.

Each strategy was implemented as a modular component in the simulation engine, ensuring that all assignment logic, including batching (where enabled), was consistently applied and all assignment decisions were fully logged for downstream performance analysis.

4.1.4 Machine Learning Model Configuration

The machine learning-based assignment strategies (ANFIS and Hybrid) were configured and tuned following best practices for context-rich operational regression tasks. The model architecture and training pipeline evolved to reflect differences between API-based and haversine/batching scenarios, as detailed below.

Table 7: Machine Learning Model Configuration (ANFIS and Hybrid Strategies)

Component	**Setting / Value**
Dimensionality Reduction	PCA (API-based), none (pairwise/haversine)
Number of PCA Components	Determined per city (8–12 typically, retain >90% variance)
ANFIS Model	BioAnfisRegressor (xanfis)
Number of Fuzzy Rules	15 (tuned for accuracy and interpretability)
Membership Function Type	Sigmoid
Optimizer	Genetic Algorithm (BaseGA)
GA Parameters	250 epochs, population size 30
Loss Function	RMSE (Root Mean Squared Error)
Random Seed	42 (for reproducibility)
Input Features	API: High-dimensional context; Haversine/Batching: 15-dimensional pairwise vector (pickup, delivery, workload, vehicle, weather, order size, capacity, period, demand, spatial centrality)
Batching Support	Yes (ANFIS/Hybrid only, with feasibility and efficiency checks)
Training Data	Synthetic, city-specific, scenario-diverse (see Section 3.7)

Note: Principal component analysis (PCA) was applied after standard scaling in the API-based model, with the number of components selected to retain at least 90% of variance. In pairwise and batching-enabled scenar-

acceptable delivery time windows.

ios, PCA was omitted due to reduced feature dimensionality. Batching feasibility was enforced via explicit logic during assignment. The ANFIS models were trained separately for each city on synthetic data reflecting realistic operational scenarios and demand variation, with hyperparameters tuned for both predictive performance and model interpretability.

4.1.5 Evaluation Metrics

Performance was evaluated using the operational and fairness metrics detailed in Section 3.8. These include average delivery latency, wait time, operational cost, courier utilization, and workload distribution. All metrics are scenario-specific and reported per simulation run, with results summarized and compared below.

4.2 Results and Comparative Analysis

The following chapter presents empirical results from the simulation-based benchmarking of courier assignment strategies in last-mile quick-commerce delivery, as defined in the methodology. Unlike most prior work, this study systematically contrasts assignment approaches not only across cities (Montreal, Rome) and operational days, but also across three simulation modes reflecting different levels of operational fidelity and batching capability: API-based (using Google Maps for travel time), Haversine (using straight-line distances), and Haversine_Batched (enabling multi-order batching).

For each simulation mode, we present and interpret aggregate performance across the four main assignment strategies: Naive, Hungarian, ANFIS, and Hybrid Pairwise. Performance is benchmarked on key operational metrics such as delivery latency, wait time, operational cost, courier utilization, and workload fairness. This structure allows for a robust evaluation of each algorithm's strengths and weaknesses under different logistical and data realism constraints, as well as the practical trade-offs associated with batching.

4.2.1 API-Based Scenario: Realistic Routing Without Batching

This scenario evaluates courier assignment strategies under the most operationally realistic conditions in the study, with all travel times and route distances computed via the Google Maps Distance Matrix API. This high-fidelity approach accurately reflects actual road networks, real-time travel speeds, and spatial barriers present in Montreal and Rome, thus serving as a benchmark for practical platform deployment. Batching is not enabled in this mode, isolating the effect of assignment logic under realistic single-order delivery conditions.

Note: The number of simulation runs for the API-based scenario was limited to four representative dates per city due to the high cost and quota constraints imposed by the Google Maps Distance Matrix API. Each simulation requires thousands of route queries to ensure high-fidelity, real-time travel time estimation. While this restricts the breadth of the temporal analysis compared to the Haversine-based scenarios, the selected dates capture a range of demand and weather conditions, providing a robust comparative benchmark for operational performance under realistic routing constraints.

Montreal Results (API Mode)

Table 8 summarizes results for Montreal across four simulated dates. All assignment strategies achieve comparable levels of order fulfillment. However, the Hungarian algorithm consistently delivers the lowest average cost and wait times, with the naive approach performing similarly but slightly less efficiently. Machine learning-based methods (ANFIS, Pairwise) yield higher costs and, particularly in some scenarios, increased wait times, reflecting the greater variability introduced by data-driven assignment.

In summary, the Hungarian algorithm emerges as the most cost-effective and time-efficient strategy in API-based Montreal scenarios, with Naive assignment performing similarly but with less consistency. Data-driven methods did not surpass classical optimization in this operational context.

Table 8: Simulation Results by Strategy for Montreal (API-Based Scenario)

Date	Strat.	Deliv.	Dist. (km)	Cost ($)	AvgD (km)	AvgT (min)	AvgW (min)	AvgC ($)
2023-06-29	Naive	223	1358.63	2419.83	6.09	26.85	3.27	10.85
2023-06-29	**Hungarian**	**223**	**1305.73**	**2329.23**	**5.86**	**25.85**	**2.94**	**10.44**
2023-06-29	ANFIS	223	1401.99	2452.40	6.29	27.14	17.00	11.00
2023-06-29	Pairwise	223	1406.39	2439.57	6.31	26.97	16.39	10.94
2022-09-06	Naive	140	793.47	1359.53	5.67	26.29	4.36	9.71
2022-09-06	**Hungarian**	**140**	**819.91**	**1343.46**	**5.86**	**25.80**	**3.13**	**9.60**
2022-09-06	ANFIS	140	897.68	1437.20	6.41	27.50	1.07	10.27
2022-09-06	Pairwise	140	899.87	1427.04	6.43	27.26	0.87	10.19
2022-09-07	Naive	140	822.43	1381.02	5.87	26.62	3.62	9.86
2022-09-07	**Hungarian**	**140**	**827.26**	**1358.58**	**5.91**	**26.10**	**2.91**	**9.70**
2022-09-07	ANFIS	140	891.17	1445.09	6.37	27.71	1.40	10.32
2022-09-07	Pairwise	140	874.77	1401.87	6.25	26.83	0.81	10.01
2024-01-04	Naive	251	1553.08	2777.51	6.19	27.40	3.29	11.07
2024-01-04	**Hungarian**	**251**	**1522.07**	**2672.57**	**6.06**	**26.29**	**2.94**	**10.65**
2024-01-04	ANFIS	251	1590.51	2791.76	6.34	27.47	32.31	11.12
2024-01-04	Pairwise	251	1618.56	2799.01	6.45	27.48	27.47	11.15

Rome Results (API Mode)

Table 9 presents results for Rome across the same dates. Again, fulfillment rates remain consistent across strategies. The Hungarian algorithm continues to outperform others in both cost and wait time, while machine learning-based strategies display slightly higher mean delivery times and increased variability in wait times.

In summary, API-based experiments in Rome reinforce the finding that classical combinatorial optimization, specifically the Hungarian algorithm, remains the most robust and cost-effective assignment method under realistic

Table 9: Simulation Results by Strategy for Rome (API-Based Scenario)

Date	Strat.	Deliv.	Dist. (km)	Cost ($)	AvgD (km)	AvgT (min)	AvgW (min)	AvgC ($)
2024-11-13	Naive	140	883.74	642.34	6.31	27.53	3.14	4.59
2024-11-13	**Hungarian**	**140**	**863.26**	**612.16**	**6.17**	**26.24**	**2.20**	**4.37**
2024-11-13	ANFIS	140	921.37	651.16	6.58	27.91	1.10	4.65
2024-11-13	Pairwise	140	921.37	651.16	6.58	27.91	1.10	4.65
2021-11-19	Naive	191	1147.94	853.16	6.01	26.80	2.91	4.47
2021-11-19	**Hungarian**	**191**	**1138.44**	**840.51**	**5.96**	**26.40**	**2.52**	**4.40**
2021-11-19	ANFIS	191	1201.56	874.17	6.29	27.46	5.56	4.58
2021-11-19	Pairwise	191	1200.68	868.99	6.29	27.30	5.59	4.55
2023-03-16	Naive	140	855.32	631.17	6.11	27.05	2.96	4.51
2023-03-16	**Hungarian**	**140**	**838.49**	**613.51**	**5.99**	**26.29**	**2.59**	**4.38**
2023-03-16	ANFIS	140	889.96	646.65	6.36	27.71	1.09	4.62
2023-03-16	Pairwise	140	889.96	646.65	6.36	27.71	1.09	4.62
2025-01-04	Naive	207	1284.31	949.50	6.20	27.52	3.05	4.59
2025-01-04	**Hungarian**	**207**	**1225.17**	**910.50**	**5.92**	**26.39**	**2.69**	**4.40**
2025-01-04	ANFIS	207	1299.38	950.34	6.28	27.55	11.42	4.59
2025-01-04	Pairwise	207	1314.17	959.99	6.35	27.83	12.09	4.64

routing assumptions. Data-driven and hybrid strategies captured context but did not translate their predictive accuracy into operational gains within the constraints of the simulation.

4.2.2 Haversine Scenario: Geometric Routing Without Batching

In the Haversine scenario, assignment strategies were evaluated across 1,500 operational days for Montreal, with routing based on straight-line distances and no batching enabled. Since all algorithms operated under identical order volumes and courier capacities, observed differences are directly attributable to the assignment logic.

Montreal Results

Table 10 reports the mean performance metrics for each strategy. While the total number of deliveries remained virtually constant across all algorithms, clear differences emerge in efficiency and cost.

The Pairwise strategy consistently delivered the lowest average delivery time (16.32 min), wait time (1.28 min), and cost per order ($6.41), indicating a modest operational advantage under geometric routing assumptions. Hungarian and Naive performed similarly but lagged in speed and cost, while ANFIS was competitive but did not surpass Pairwise on any primary metric. Although these differences are subtle, the results suggest context-aware strategies can provide incremental gains even in simplified routing environments.

To complement these aggregate results, Figure 16 presents the mean active and idle minutes per courier across

Table 10: Aggregate Simulation Results by Assignment Strategy for Montreal, Haversine Scenario (N = 1,500 days)

Strat.	**Tot_Dist. (km)**	**Tot_Cost ($)**	**AvgD (km)**	**AvgT (min)**	**AvgW (min)**	**AvgC ($)**
Naive	870.3	1409.0	4.08	16.93	1.97	6.59
Hungarian	923.0	1407.2	4.36	16.74	1.39	6.59
Anfis	920.9	1389.3	4.35	16.40	1.87	6.47
Pairwise	889.2	1374.8	4.18	**16.32**	**1.28**	**6.41**

assignment modes, with error bars indicating the standard deviation. While all strategies achieve comparable averages, context-aware approaches such as Pairwise and ANFIS show reduced variability in workload distribution, highlighting a secondary benefit in terms of fleet balance and operational fairness.

To complement these aggregate results, Figure 16 presents the mean active and idle minutes per courier across assignment modes, with error bars indicating the standard deviation. While all strategies achieve comparable averages, context-aware approaches such as Pairwise and ANFIS show reduced variability in workload distribution, highlighting a secondary benefit in terms of fleet balance and operational fairness.

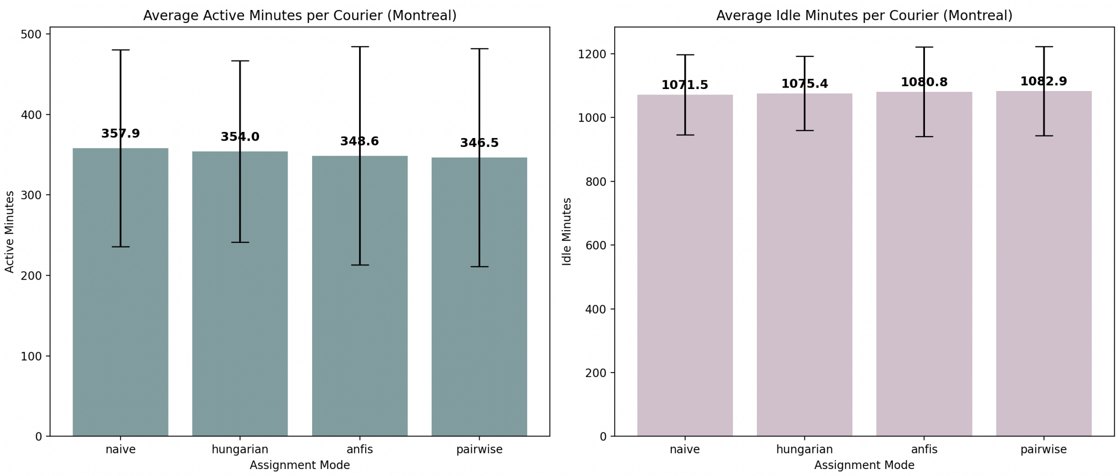


Figure 16: Average active minutes (left) and idle minutes (right) per courier by assignment strategy in Montreal. Error bars indicate one standard deviation.

Boxplots in Figure 17 illustrate the distribution of average delivery times for each assignment strategy under different days of the week (left) and various weather conditions (right). The Pairwise approach not only achieves the lowest average delivery time but also demonstrates robust performance across temporal and environmental contexts, with less pronounced variability compared to baseline algorithms. This robustness suggests the operational value of adaptive assignment logic, particularly in dynamic urban environments.

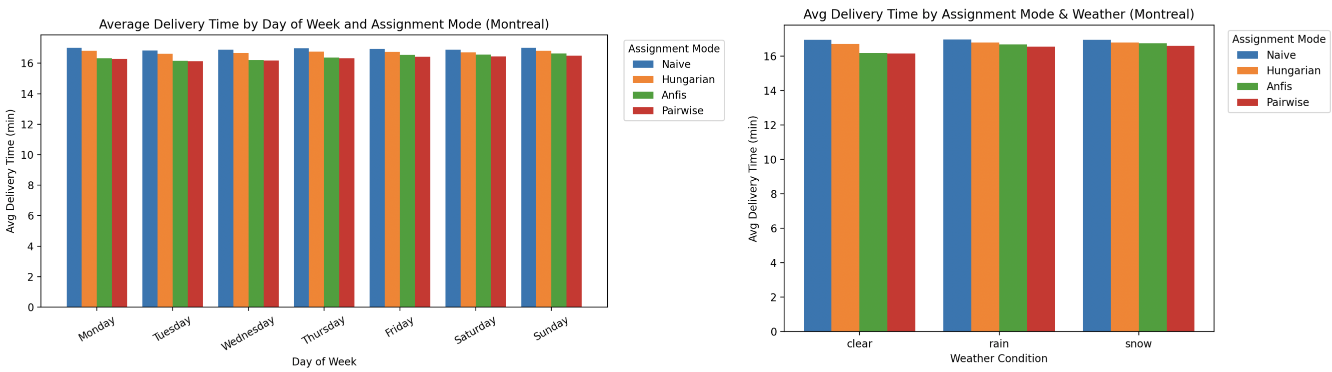


Figure 17: Distribution of average delivery times by assignment strategy in Montreal under different days of the week (left) and weather conditions (right), Haversine scenario.

Rome Results

Table 11 summarizes the corresponding results for Rome. As in Montreal, the assignment logic was the sole source of variation.

Table 11: Aggregate Simulation Results by Assignment Strategy for Rome, Haversine Scenario (N = 1,500 days)

Strat.	**Tot_Dist. (km)**	**Tot_Cost ($)**	**AvgD (km)**	**AvgT (min)**	**AvgW (min)**	**AvgC ($)**
Naive	**567.48**	579.14	**2.95**	17.25	0.65	3.00
Hungarian	603.17	573.20	3.14	17.07	0.40	2.97
Anfis	600.24	559.26	3.13	16.58	0.52	2.88
Pairwise	584.90	**556.61**	3.04	**16.52**	**0.37**	**2.87**

The Pairwise approach again produced the lowest cost per order ($2.87), fastest average delivery time (16.52 min), and shortest wait time (0.37 min), with ANFIS close behind. Hungarian and Naive trailed modestly, though all strategies provided similar fulfillment rates.

Figure 18 presents the mean active and idle minutes per courier across assignment modes, with error bars indicating the standard deviation. As in Montreal, all strategies show similar average values, but context-aware approaches such as Pairwise and ANFIS exhibit slightly reduced variability, suggesting a more balanced distribution of courier workloads.

Figure 19 illustrates the distribution of average delivery times for each assignment strategy under different days of the week (left) and varying weather conditions (right). Pairwise and ANFIS again deliver the fastest and most consistent performance across both temporal and environmental contexts, with notably reduced spread in delivery times compared to the baseline approaches. This consistency highlights the operational robustness of adaptive assignment logic in the face of daily and environmental variability.

Taken together, these results for Rome reinforce the finding that context, aware assignment strategies, especially Pairwise, offer modest but meaningful improvements in both efficiency and workload balance under geometric routing assumptions. These operational insights provide a foundation for further analysis of real-world implications, which are addressed in Section 4.3.

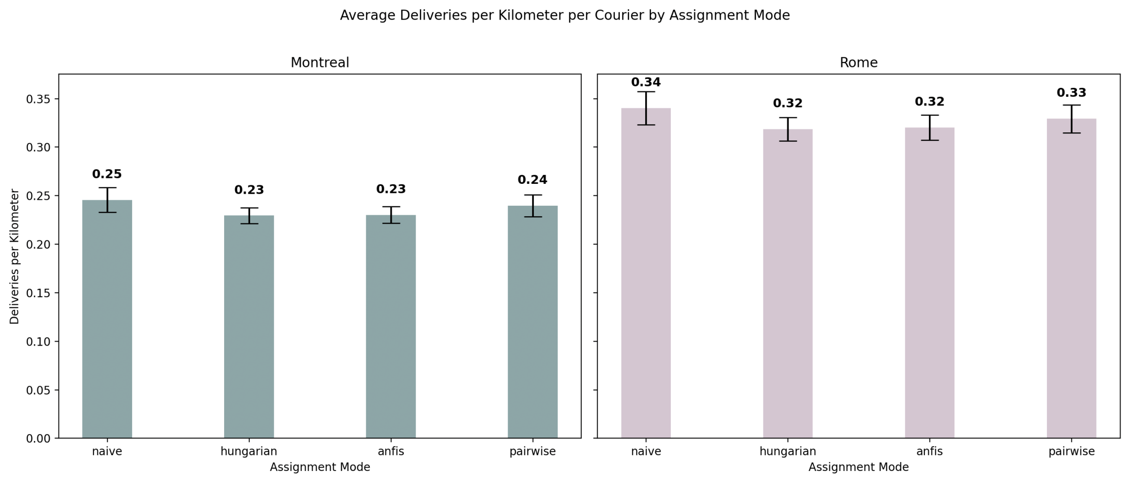


Figure 18: Average active minutes (left) and idle minutes (right) per courier by assignment strategy in Rome. Error bars indicate one standard deviation.

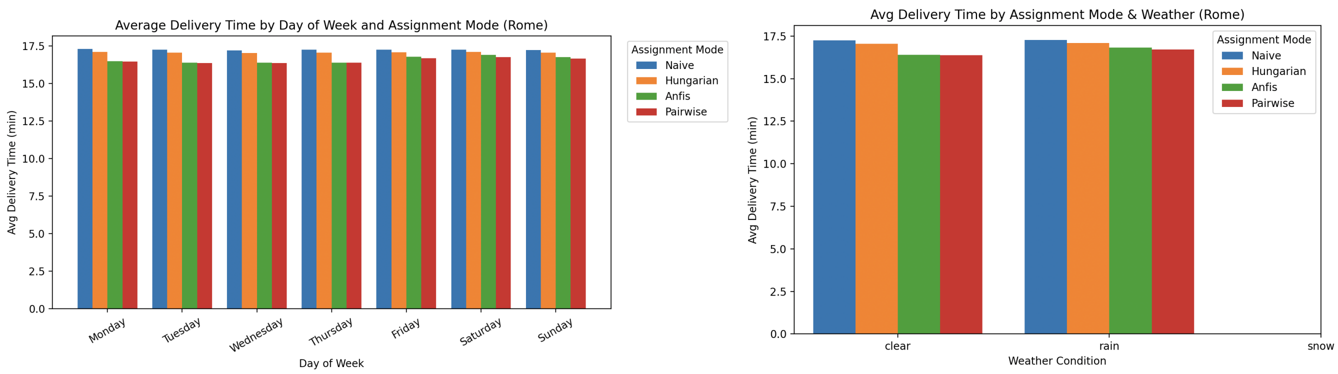


Figure 19: Distribution of average delivery times by assignment strategy in Rome under different days of the week (left) and weather conditions (right), Haversine scenario.

Summary

In both cities, while all assignment algorithms delivered a comparable number of orders, context-aware strategies, particularly Pairwise, consistently offered modest improvements in delivery speed and cost per order. Beyond average performance, the advanced strategies also reduced variability in key operational metrics, yielding more balanced courier workloads and higher efficiency as reflected in deliveries per kilometer. These findings reinforce the value of adaptive assignment logic for both cost and fairness, even when routing and batching complexities are minimized. Broader implications for platform management and supply chain sustainability are discussed in Section 4.3.

4.2.3 Haversine with Batching: Assessing the Impact of Multi-Order Assignment for ML Models

To complement the results of the baseline Haversine scenario, this section examines the effects of enabling multi-order batching for machine learning–based assignment algorithms. The analysis is restricted to ANFIS and Pairwise strategies, with the objective of determining how batching influences last-mile delivery performance. By comparing these findings to the non-batching scenario, the practical value of advanced assignment logic under realistic batching constraints can be quantified.

Aggregate Results

Table 12 presents aggregate simulation outcomes for ANFIS and Pairwise assignment strategies in Montreal and Rome under the Haversine Batching scenario. These results are directly comparable to previous aggregate metrics, allowing for a clear evaluation of the incremental benefits associated with batching.

Table 12: Aggregate Simulation Results for ML Assignment Strategies under Haversine Batching (N = 1,500 days)

City	Strat.	Tot_Dist. (km)	Tot_Cost ($)	AvgD (km)	AvgT (min)	AvgW (min)	AvgC ($)	Tot_Batches
Montreal	Anfis	914.01	1396.25	4.31	16.56	2.04	6.52	2.80
Montreal	Pairwise	917.15	1386.41	4.34	16.42	1.36	6.48	1.77
Rome	Anfis	601.17	570.28	3.13	16.96	1.83	2.95	6.30
Rome	Pairwise	601.77	562.04	3.14	16.70	0.42	2.90	3.34

Across both cities, the Pairwise model demonstrates lower average delivery times, wait times, and costs compared to ANFIS when batching is enabled. The total number of batches executed is also consistently lower for Pairwise, indicating a more efficient batching policy, especially pronounced in Rome, where the number of batches is nearly halved. These results highlight the operational advantage of advanced assignment algorithms in environments where batching is available, further enhancing efficiency and fleet utilization relative to single-order dispatch.

4.3 Discussion and Findings

Comparative evaluation of assignment strategies across both geometric routing scenarios, with and without batching, yields a series of operational and strategic insights for the optimization of last-mile delivery, directly informing supply chain management practice.

In the Haversine scenario without batching, context-aware strategies, particularly Pairwise, consistently demonstrated advantages over classical approaches such as Naive and Hungarian. These gains, while modest in absolute value, were manifested in reduced mean delivery times, lower average costs, and a more equitable distribution of workload among couriers. Such improvements in fairness, evidenced by tighter variance in active and idle minutes, are particularly relevant for labor management and courier retention, a persistent challenge in the gig economy segment of last-mile logistics.

The introduction of multi-order batching accentuated these differences, especially in more complex urban environments like Rome. Here, the Pairwise model not only minimized delivery and wait times, but also executed substantially fewer batches than ANFIS, translating into fewer vehicle trips and more efficient route consolidation. These gains go beyond individual algorithmic performance: they have system-level impacts, including reductions in operational cost, environmental footprint (via lower total kilometers traveled), and overall fleet size requirements. For supply chain leaders, such efficiencies directly translate to improved scalability and reduced marginal costs per additional order, a core objective in the economics of platform-based delivery.

Robustness analysis further revealed that both ANFIS and Pairwise maintained stable performance across diverse temporal and environmental conditions, including day-of-week and weather variability. This resilience is crucial for supply chain continuity, as disruptions from unpredictable demand peaks or adverse weather often strain conventional routing and assignment logic. The ability of adaptive, ML-driven strategies to sustain service levels under such volatility positions them as key enablers of more resilient, customer-centric last-mile systems.

Urban contrasts: Montreal versus Rome.

While both cities showed similar trends regarding the relative strengths of each assignment strategy, some key contextual differences are worth emphasizing. In Montreal, where the street network is generally more grid-like and delivery demand is more dispersed, the operational advantage of batching and advanced assignment logic was more muted. In contrast, Rome's dense, irregular urban fabric and concentrated demand clusters created more opportunities for route consolidation and batching efficiency, amplifying the impact of intelligent assignment. Additionally, local regulatory structures, such as labor rules and compensation schemes, shape both the feasibility and benefit of assignment strategies. For example, Rome's greater regulatory emphasis on courier protections may further increase the value of workload balancing and fairness offered by context-aware algorithms. These findings underscore the need for platform operators to tailor optimization strategies to the local urban topology, regulatory environment, and demand patterns, rather than seeking one-size-fits-all solutions.

From a broader supply chain perspective, these findings underscore the value of integrating dynamic, data-driven assignment algorithms into urban delivery networks. More efficient batching, in particular, represents a form of operational leverage: by enabling couriers to serve multiple orders per trip without significant degradation of service time, platforms can achieve a step-change in both productivity and environmental sustainability. This has downstream effects on urban congestion, emissions, and the public perception of quick commerce services.

At a strategic level, the adoption of such intelligent assignment logic allows operators to more effectively manage fluctuating demand, allocate resources dynamically, and tailor service offerings to evolving market needs. For practitioners, this means not only lower direct costs, but also enhanced flexibility in workforce deployment and the potential to offer differentiated service tiers (e.g., express, scheduled, or green delivery options, etc.) without unsustainable increases in complexity.

In general, the results presented here illustrate how advanced assignment and batching strategies can serve as a foundation for more efficient, robust, and sustainable last-mile delivery systems, critical priorities for modern supply chain management as urban logistics continues to evolve.

4.4 Limitations

Several simplifying assumptions and practical limitations should be acknowledged in interpreting the results of this study:

- **Courier behavior:** Couriers in the simulation neither experience fatigue nor have the autonomy to refuse or reassign orders. In reality, human behavior introduces additional variability, including shift changes, breaks, refusals, and dynamic availability.
- **Synthetic data and lack of real-world validation:** The majority of model training and simulation experiments were conducted using synthetic datasets, with input distributions and event patterns designed to mimic

real-world operations. While this enables controlled experimentation and scalability, it may not capture all the complexity, variability, or edge cases present in actual urban delivery systems. Future validation on real operational data is essential to confirm generalizability and practical impact.

- **Temporal demand patterns:** Orders are generated only within fixed time groups, i.e., 10:00 to 12:00, 12:00 to 15:00 and 18:00 to 22:00, reflecting peak demand windows. This simplification does not account for off-peak dynamics or unexpected surges.
- **Idle time calculation:** Courier idle minutes, as reported in the results, include all non-active time, potentially spanning periods before the first order and after the last order of the day, rather than strictly representing inefficiency during active shifts. Thus, absolute idle time values should be interpreted with caution and are most meaningful when compared across strategies rather than in isolation.
- **Wait time calculation:** Due to a coding oversight, the wait time metric was not reliably updated for all orders, resulting in an artificially high frequency of zeros in the reported results. As a consequence, the distribution of wait times in the present analysis may not accurately reflect actual system performance, and these values should be interpreted with caution.
- **Geometric routing assumptions:** The use of Haversine distance for route estimation is optimistic, neglecting critical real-world factors such as traffic congestion, road network constraints, accidents, and signal delays. Actual travel times may be considerably higher and more variable.
- **Batching constraints and practical challenges:** The batching implementation serves primarily as a proof-of-concept. In practice, forcing more batching or dynamically adjusting batching thresholds could yield better efficiency, but would require a system designed for large-scale, real-time optimization. Achieving this demands close collaboration between engineering and data science teams.
- **Order heterogeneity and operational complexity:** Real delivery systems handle a diverse mix of order types (e.g., food, non-food), each with unique constraints (such as food temperature sensitivity). Some stores may have dedicated couriers or share couriers with neighboring stores, further complicating assignment logic.
- **Constraint realism:** The eligibility constraints for courier-order matches in this study are a small subset of those found in actual platforms. Real deployments must handle additional rules, exceptions, and business logic, including hot-swapping, dynamic re-optimization, and order reassignments.
- **System uncertainty and real-world events:** Even after assignment, actual execution is not guaranteed; device failures, vehicle breakdowns, or customer cancellations can intervene at any time. Continuous tracking and rapid reoptimization are required in practice to maintain efficiency and service quality.
- **Decision timing and reactivity:** Decisions made too early in the process are often suboptimal, as later information can shift priorities. Practical systems benefit from postponing non-urgent assignments to improve accuracy, but this was only partially modeled in the current framework.

Overall, the present study should be regarded as a proof-of-concept, offering a foundation for further research and practical development. Substantial room remains for increasing realism and operational robustness, particularly by integrating richer constraints, more dynamic batching logic, network-based routing, and explicit modeling of human and environmental factors.

5 Conclusion

5.1 Concluding Remarks

This thesis set out to address the research questions outlined in Section 1.4 by systematically benchmarking both classical and machine learning–based courier assignment algorithms under diverse operational scenarios within the quick commerce context. Through empirical and comparative analyses in Montreal and Rome, several core insights emerged that speak directly to the research objectives.

Comparative performance of assignment algorithms. The results confirm that classical combinatorial optimization, as exemplified by the Hungarian algorithm, provides strong and consistent performance across all tested environments, most notably achieving the lowest delivery latency and operational cost in high-fidelity, API-based routing scenarios. Data-driven approaches, including ANFIS and hybrid pairwise models, demonstrated strong predictive ability during training but did not consistently deliver superior operational outcomes. However, context-aware ML models—particularly when batching is enabled or geometric routing is assumed—yielded modest but meaningful improvements in delivery efficiency and workload balance. These operational gains, while context-dependent, demonstrate that advanced assignment logic can enhance both fairness and performance, especially in environments where batching and route consolidation are feasible.

Strengths and limitations of machine learning and neuro-fuzzy models. While advanced ML and neuro-fuzzy methods offer promising flexibility and, in some cases, improved interpretability, the simulation results highlight practical challenges in scalability, real-time deployment, and robustness. The persistent gap between predictive accuracy and operational impact suggests that machine learning strategies alone are insufficient for complex, dynamic environments unless further integrated into hybrid frameworks.

Urban contrasts and transferability. Cross-city comparison revealed broadly similar performance trends in both Montreal and Rome. Yet, the impact of advanced assignment strategies was more pronounced in Rome, where dense urban structure and concentrated demand provided greater opportunity for batching and route optimization. In contrast, the dispersed geography of Montreal led to more muted gains from batching. These findings reinforce the importance of tailoring optimization strategies to the unique infrastructure, regulatory context, and demand patterns of each market.

Trade-offs, edge cases, and practical lessons. The simulation framework surfaced critical trade-offs, including the operational complexity of batching, the effects of demand clustering, and issues of workload fairness among couriers. These results underscore the importance of comprehensive scenario-based evaluation and highlight the potential of hybrid approaches that blend the reliability of combinatorial optimization with the adaptability of ML-driven methods.

In summary, this work reaffirms the enduring value of classical optimization for last-mile delivery while clarifying both the operational promise and present limitations of modern data-driven strategies. Achieving robust, scalable, and context-sensitive solutions in urban logistics requires not only technical innovation, but also a careful alignment of algorithms, data, and real-world constraints. The findings presented here provide a data-driven foundation for ongoing advancements in last-mile delivery, contributing practical guidance for both researchers and practitioners in the evolving urban logistics ecosystem.

5.2 Future Work

While this thesis establishes a robust proof-of-concept for data-driven courier assignment in last-mile delivery, several promising directions remain for future research:

- Scaling the simulation to larger, more complex urban environments to assess the generalizability and scalability of assignment strategies.
- Incorporating stronger incentives and algorithmic mechanisms to encourage batching, and investigating optimal batching policies.
- Integrating real-time data sources and external APIs (e.g., live traffic, weather) to more accurately reflect operational realities.
- Enhancing courier realism by assigning richer attributes (such as fatigue, preferences, and variable availability) to better capture human behavior.
- Modeling stochastic demand and incorporating demand forecasting to improve resource allocation and assignment responsiveness.
- Increasing the robustness and fidelity of operational metrics (e.g., more accurate idle and wait time measurement), alongside more advanced error handling.
- Simulating true road network routing and congestion effects by leveraging mapping APIs or open-source urban graph data.
- Allowing for human-in-the-loop decisions, order refusals, and dynamic shift patterns.
- Extending the model to support multiple order types and differentiated service levels, reflecting the full range of platform offerings.
- Exploring dynamic pricing, incentive schemes, and their influence on assignment and batching outcomes.
- Validating results against industry datasets or via pilot deployments with real delivery platforms.
- Quantifying environmental and social impacts, including emissions reductions and workforce equity.
- Developing operational dashboards or visualization tools for decision support and live monitoring.

Pursuing these directions will further enhance the realism, operational impact, and practical applicability of advanced assignment algorithms in the evolving landscape of last-mile logistics.

References

[Accenture, 2021] Accenture (2021). The sustainable last-mile. `https://www.accenture.com/content/dam/accenture/final/a-com-migration/r3-3/pdf/pdf-148/accenture-sustainable-mile-pov.pdf`.

[Andersson, 2021] Andersson, T. (2021). *A Comparative Study on a Dynamic Pickup and Delivery Problem: Improving routing and order assignment in same-day courier operations*. PhD thesis, KTH Royal Institute of Technology.

[Ardoni, 2022] Ardoni, M. V. (2022). Perché gorillas & co chiude in italia: quale lezione per il quick commerce. `https://www.agendadigitale.eu/mercati-digitali/ecommerce/perche-gorillas-co-chiude-in-italia-quale-lezione-per-il-quick-commerce/`.

[Awati, 2022] Awati, R. (2022). Geofencing. `https://www.techtarget.com/whatis/definition/geofencing#:~:text=Geofencing%20creates%20a%20virtual%20geographical,%2D%20or%20RFID%2Denabled%20software`.

[Berger, 2022] Berger, R. (2022). Quick commerce: a lasting revolution? `https://www.rolandberger.com/en/Insights/Publications/Quick-commerce-a-lasting-revolution.html`.

[Blair, 2001] Blair, J. (2001). Kozmo to end operations; 1,100 people to lose jobs. `https://www.nytimes.com/2001/04/12/nyregion/kozmo-to-end-operations-1100-people-to-lose-jobs.html`.

[Boeing, 2017] Boeing, G. (2017). Osmnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems*, 65:126–139.

[Bogdanova, 2021] Bogdanova, M. (2021). Quick commerce in western europe: Trends, operational models and prospects. `https://www.euromonitor.com/article/quick-commerce-in-western-europe-trends-operational-models-and-prospects`.

[Boggio-Marzet et al., 2023] Boggio-Marzet, A., Villa-Martínez, R., and Monzón, A. (2023). Selection of policy actions for e-commerce last-mile delivery in cities: An online multi-actor multi-criteria evaluation. *Transport Policy*, 142:15–27.

[Carbonaro, 2024] Carbonaro, G. (2024). The oldest country in europe: What's behind italy's ageing problem. `https://www.euronews.com/2024/02/28/the-oldest-country-in-europe-whats-behind-italys-ageing-problem`.

[Chang and Yen, 2012] Chang, T.-S. and Yen, H.-M. (2012). City-courier routing and scheduling problems. *European Journal of Operational Research*, 223(2):489–498.

[Chen and Hu, 2024] Chen, M. and Hu, M. (2024). Courier dispatch in on-demand delivery. *arXiv preprint arXiv:2312.06353*.

[Company, 2025] Company, M. . (2025). The state of grocery retail 2025. `https://www.mckinsey.com/industries/retail/our-insights/state-of-grocery-europe#/download/%2F~%2Fmedia%2Fmckinsey%2Fi` ndustries%2Fretail%2Four%20insights%2Fstate%20of%20grocery%20europe%202025%20quest%20for%20growth%2Fthe-state-of-grocery-retail-europe-2025.pdf%3FshouldIndex%3Dfalse.

[Consumers Council of Canada, 2024] Consumers Council of Canada (2024). Canadian grocery shopping habits shift as prices rise. https://www.consumerscouncil.com/canadian-grocery-shopping-habits-shift-as-prices-rise/.

[developers,] developers, J. Joblib: running python functions as pipeline jobs. https://joblib.readthedocs.io/.

[DHL, 2023] DHL (2023). Everything you need to know about last mile delivery. https://www.dhl.com/discover/en-id/logistics-advice/logistics-insights/what-is-last-mile-delivery.

[DoorDash, 2024] DoorDash (2024). Online ordering trends report 2024. Hospitality Magazine, retrieved from https://www.hospitalitymagazine.com.au/food-delivery-trend-report-shows-late-night-orders-are-booming/.

[Fan et al., 2024] Fan, Y., Cao, Y., Xiong, X., and Wei, K. (2024). Optimization of courier-order matching and delivery performance for in-time delivery systems. Submitted to Transportation Science, SSRN preprint.

[Fegde, 2025] Fegde, N. (2025). Challenges and solutions in managing last-mile delivery in urban areas. *International Journal of Logistics Management*, 29(1):1–10.

[Ferrucci and Bock, 2014] Ferrucci, F. and Bock, S. (2014). Real-time control of express pickup and delivery processes in a dynamic environment. *Transportation Research Part B: Methodological*, 63:1–14.

[Glovo, 2020] Glovo (2020). Hi, this is jarvis. https://about.glovoapp.com/stories/hi-this-is-jarvis/.

[Glovo, 2023] Glovo (2023). E-bike per il delivery a noleggio! https://delivery.glovoapp.com/it/discounts/cycle/.

[Goel and Sharma, 2025] Goel, K. and Sharma, D. (2025). Quick commerce and customer retention: A study of blinkit and zepto. *INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, 09:1–9.

[Government of Canada, 2023] Government of Canada (2023). Government of canada protects federally regulated workers. https://www.newswire.ca/news-releases/government-of-canada-protects-federally-regulated-workers-867821624.html.

[Hamdan et al., 2023] Hamdan, I., Aziguli, W., Zhang, D., et al. (2023). Machine learning in supply chain: prediction of real-time e-order arrivals using anfis. *International Journal of System Assurance Engineering and Management*, 14(Suppl 1):549–568.

[Harris et al., 2020] Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., et al. (2020). Array programming with numpy. *Nature*, 585(7825):357–362.

[INFORMS, 2016] INFORMS (2016). Ups on-road integrated optimization and navigation (orion) project. https://www.informs.org/Impact/O.R.-Analytics-Success-Stories/Optimizing-Delivery-Routes.

[Instacart, 2022] Instacart (2022). Instacart now partners with canada's top five grocers, adds ten new retailers. https://www.instacart.com/company/pressreleases/instacart-now-partners-with-canadas-top-five-grocers-adds-ten-new-retailers/.

[Jang, 1993] Jang, J.-S. (1993). Anfis: adaptive-network-based fuzzy inference system. *IEEE transactions on systems, man, and cybernetics*, 23(3):665–685.

[Joshi et al., 2021] Joshi, M., Singh, A., Ranu, S., Bagchi, A., Karia, P., and Kala, P. (2021). Batching and matching for food delivery in dynamic road networks. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pages 2099–2104.

[Joshi et al., 2022] Joshi, M., Singh, A., Ranu, S., Bagchi, A., Karia, P., and Kala, P. (2022). Foodmatch: Batching and matching for food delivery in dynamic road networks. *ACM Trans. Spatial Algorithms Syst.*, 8(1).

[Kondratov and Tarasova, 2024] Kondratov, P. and Tarasova, E. (2024). Optimizing food delivery efficiency: The impact of order aggregation and courier assignment strategies. In *Optimization, Learning Algorithms and Applications. 4th International Conference, OL2A 2024, Proceedings, Part II*, pages 50–66. Springer.

[Kumar and Chidambara, 2024] Kumar, I. and Chidambara (2024). A systematic literature review and bibliometric analysis of last-mile e-commerce delivery in urban areas. *Urban, Planning and Transport Research*, 12(1):2357577.

[Leonard, 2021] Leonard, M. (2021). Ups adds dynamic routing to orion, saving 2-4 miles per driver. https://www.supplychaindive.com/news/ups-orion-route-planning-analytics-data-logistics/601673/.

[Li et al., 2023] Li, Y., Wu, Q., Huang, X., Xu, J., Gao, W., and Xu, M. (2023). Efficient adaptive matching for real-time city express delivery. *IEEE Transactions on Knowledge and Data Engineering*, 35(6):5767–5779.

[Lu et al., 2022] Lu, M., Huang, C., and Teng, J. (2022). Multi-agent simulation for online fresh food autonomous delivery. *Journal of System Simulation*, 34(6):1185–1195.

[Lydersen, 2024] Lydersen, K. (2024). Extreme weather is fueling a surge in food delivery—at a cost to workers. Inside Climate News, retrieved from https://insideclimatenews.org/news/07012025/todays-climate-extreme-weather-food-delivery-workers/.

[MacConchie, 2024] MacConchie, A. (2024). Empire expands its e-commerce via instacart, uber eats partnerships. https://www.supermarketnews.com/grocery-technology/empire-expands-its-ecommerce-via-instacart-uber-eats-partnerships.

[Mattia, 2023] Mattia, M. D. (2023). Uber eats dismette le operazioni in italia. https://www.uber.com/it/newsroom/uber-eats-dismette-le-operazioni-in-italia/.

[McKinney, 2010] McKinney, W. (2010). Data structures for statistical computing in python. *Proceedings of the 9th Python in Science Conference*, 445:51–56.

[Mintel, 2023] Mintel (2023). Italy online grocery retailing market report. https://store.mintel.com/report/italy-online-grocery-retailing-market-report.

[NielsenIQ, 2024] NielsenIQ (2024). Everything you need to know about u.s. quick commerce. https://nielseniq.com/global/en/insights/analysis/2024/everthing-you-need-to-know-about-us-quick-commerce/.

[Ninikas et al., 2014] Ninikas, G., Athanasopoulos, T., Zeimpekis, V., and Minis, I. (2014). Integrated planning in hybrid courier operations. *The International Journal of Logistics Management*, 25(3):611–634.

[Ou et al., 2024] Ou, S., Ismail, Z. H., and Sariff, N. (2024). Hybrid genetic algorithms for order assignment and batching in picking system: A systematic literature review. *IEEE Access*, 12:23029–23042.

[Parodi, 2021] Parodi, E. (2021). Milan prosecutors order food delivery groups to hire riders, pay 733 mln euros in fines. https://www.reuters.com/business/milan-prosecutors-order-food-delivery-groups-hire-riders-pay-733-mln-euros-fines-2021-02-24/.

[Paszke et al., 2019] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12:2825–2830.

[Presskila et al., 2025] Presskila, X. A., Raj, J. R. F., Kumar, P. S., Sankar, G. R., Rajkumar, G. V., and Sundaravadivel, P. (2025). A hybrid learning-based q-commerce framework evolving from e-commerce. In *2025 International Conference on Machine Learning and Autonomous Systems (ICMLAS)*, pages 295–302.

[Raj et al., 2024] Raj, G., Roy, D., de Koster, R., and Bansal, V. (2024). Stochastic modeling of integrated order fulfillment processes with delivery time promise: Order picking, batching, and last-mile delivery. *European Journal of Operational Research*, 316(3):1114–1128.

[Raj and Thandayudhapani, 2024] Raj, G. and Thandayudhapani, S. (2024). Evolution of e-commerce logistics: Global trends and implementations. *ComFin Research*, 12:42–45.

[Research, 2024] Research, B. (2024). Canada quick commerce market overview, 2024. https://www.bonafideresearch.com/product/6408595318/canada-quick-commerce-market.

[Research, 2021] Research, N. E. M. (2021). Grocery retail in italy. https://assets.ctfassets.net/pn8wbiqtnzw9/7bSAZfy5rn0VuTHR4anP8Y/ec7c98b9c0f8f4c45092a6c6166c2410/NZTE_MR_Grocery_retail_in_Italy_May_2021_Final.pdf.

[Restrepo et al., 2019] Restrepo, M. I., Semet, F., and Pocreaud, T. (2019). Integrated shift scheduling and load assignment optimization for attended home delivery. *Transportation Science*, 53(4):1150–1174.

[Reyes et al., 2018] Reyes, D., Erera, A., Savelsbergh, M., Sahasrabudhe, S., and O'Neil, R. (2018). The meal delivery routing problem. *Optimization Online*, 6571:2018.

[Saini, 2019] Saini, M. (2019). Why walmart failed in germany. `https://medium.com/@mohitsainiauthor/why-walmart-failed-in-germany-3fdcc6469b89`.

[Sanchez, 2024] Sanchez, P. L. (2024). An analysis of the drivers of consumers' purchasing behavior in quick commerce platforms. Master's thesis, Aalto University, School of Business.

[Show, 2024] Show, T. R. R. (2024). 21: Is quick commerce dead or pioneering the future? `https://podcasts.apple.com/us/podcast/21-is-quick-commerce-dead-or-pioneering-the-future/id1513303140?i=1000648004857`.

[Statista, 2023a] Statista (2023a). Grocery shopping frequency in canada in 2024. `https://www.statista.com/statistics/1441143/grocery-shopping-frequency-in-canada/`.

[Statista, 2023b] Statista (2023b). How shoppers rank factors for choosing grocery stores in canada as of 2023. `https://www.statista.com/statistics/1394135/canadian-shoppers-grocery-store-ranking-factors/`.

[Statista, 2023c] Statista (2023c). Revenue of the leading online food delivery companies worldwide in 2023. `https://www.statista.com/statistics/1360183/revenue-leading-online-food-delivery-companies-worldwide/`.

[Statista, 2024a] Statista (2024a). Food shopping behavior in canada. `https://www.statista.com/topics/2871/food-shopping-behavior-in-canada/#topicOverview`.

[Statista, 2024b] Statista (2024b). Online food delivery bookings by brand in canada 2017-2028. `https://www.statista.com/forecasts/998501/online-food-delivery-bookings-by-brand-in-canada`.

[Statista, 2024c] Statista (2024c). Top grocery delivery apps in canada in 2024, by downloads. https://www.statista.com/statistics/1428710/top-grocery-delivery-apps-canada/.

[Statista, 2025a] Statista (2025a). E-commerce worldwide - statista dossier. https://www.statista.com/study/10653/e-commerce-worldwide-statista-dossier/.

[Statista, 2025b] Statista (2025b). E-commerce worldwide: Market insights and data. `https://www.statista.com/outlook/emo/ecommerce/worldwide`.

[Statista, 2025c] Statista (2025c). Quick commerce worldwide: Market insights and data. `https://www.statista.com/outlook/emo/online-food-delivery/grocery-delivery/quick-commerce/worldwide`.

[Stogdon, 2024] Stogdon, E.-J. (2024). Cash or card in italy? `https://wise.com/gb/blog/cash-or-card-in-italy`.

[Stojanov, 2022] Stojanov, M. (2022). Q-commerce – the next generation e-commerce. 1:17–34.

[Thieu, 2025] Thieu, N. V. (2025). X-anfis: An extensible and cross-learning anfis framework for machine learning tasks.

[Tuncay and Ersen, 2022] Tuncay, E. and Ersen, H. (2022). Getir buys grocery app rival gorillas in $1.2 bln deal - ft. https://www.reuters.com/markets/deals/getir-buys-grocery-app-rival-gorillas-12-bln-deal-ft-2022-12-09/.

[UNCTAD, 2021] UNCTAD (2021). How covid-19 triggered the digital and e-commerce turning point. https://unctad.org/news/how-covid-19-triggered-digital-and-e-commerce-turning-point.

[UT Permian Basin, 2022] UT Permian Basin (2022). The evolution of e-commerce. https://online.utpb.edu/about-us/articles/business/%EF%BF%BCthe-evolution-of-e-commerce/.

[Ville de Montréal, 2024] Ville de Montréal (2024). Cycling and bike paths. https://montreal.ca/en/topics/cycling-and-bike-paths.

[Virtanen et al., 2020] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., et al. (2020). Scipy 1.0: Fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272.

[Vásconez et al., 2024] Vásconez, J. P., Schotborgh, E., Vásconez, I. N., Moya, V., Pilco, A., Menéndez, O., Guamán-Rivera, R., and Guevara, L. (2024). Smart delivery assignment through machine learning and the hungarian algorithm. *Smart Cities*, 7(3):1109–1125.

[Xu et al., 2025] Xu, H., Shao, J., and Sun, Y. (2025). Research on order batching optimization based on improved nsga-ii algorithm. *PLOS ONE*, 20(2):e0319182.

[Yadav, 2024] Yadav, A. (2024). Revolutionize your supply chain: Mastering geofencing logistics for enhanced efficiency. https://networkon.io/resources/blog/geofencing-in-logistics/.

[YouGov, 2024] YouGov (2024). Items consumers would consider quick delivery services for in italy in 2024 [graph]. https://www.statista.com/statistics/1499171/italy-consumers-quickly-delivery-services/.

[Zhang et al., 2019] Zhang, Y., Liu, Y., Li, G., Ding, Y., Chen, N., Zhang, H., He, T., and Zhang, D. (2019). Route prediction for instant delivery. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(3):124:1–124:25.

[Zhu, 2024] Zhu, H. (2024). The impact of covid-19 on global retail consumer behavior: Regression model and graphical approach. *SHS Web of Conferences*, 207:04017.

[Zou et al., 2023] Zou, G., Gao, M., Tang, J., and Yilmaz, L. (2023). Simulation of online food ordering delivery strategies using multi-agent system models. *Journal of Simulation*, 17(3):297–311.