Annals of Operations Research
https://doi.org/10.1007/s10479-026-07285-9

ORIGINAL RESEARCH

Efficient Forecast-Based Routing and Dynamic Time Window Management for Attended Home Deliveries

Mohammad Abdollahi[1] · Xinan Yang[1] · Michael Fairbank[2]

Received: 3 January 2025 / Accepted: 18 May 2026

Abstract
In light of the escalating popularity of online shopping and the urgent need to reduce unnecessary driving to mitigate environmental impacts, it has become increasingly important to provide cost-effective solutions for attended home deliveries. Extensive research efforts have been dedicated to addressing challenges related to integrating demand management and vehicle routing with time windows. In this paper, we present two key contributions. Firstly, we propose an enhanced method for estimating opportunity cost, by leveraging a dynamic-routing and distribution approach that incorporates forecast orders. This approach allows for more accurate revenue-loss assessments, ultimately leading to improved decision-making on the delivery charges. Secondly, we introduce a dynamic slot-combination strategy, aiming to fully exploit the flexibility that customers possess in receiving their delivery, which enhances overall route efficiency and customer satisfaction. Importantly, our proposed augmented time-windows approach can be easily implemented within existing systems, employing standard time windows, without necessitating any strategic changes or complex computational modifications to the routing system. To assess the performance of our proposed approach, we conducted exhaustive experiments on real data. The results demonstrate that our generated solution outperforms both recent and current state-of-the-art approaches in terms of profitability and delivery efficiency. This signifies the effectiveness and practicality of our proposed methodology in addressing the challenges associated with attended home deliveries.

Keywords Augmented Time Window · Dynamic Slotting · Attended Home Delivery · Dynamic Pricing · Logistics

✉ Mohammad Abdollahi
m.abdollahiuk@gmail.com

Xinan Yang
xyangk@essex.ac.uk

Michael Fairbank
m.fairbank@essex.ac.uk

1 School of Mathematics, Statistics and Actuarial Science, University of Essex, Colchester CO4 3SQ, United Kingdom

2 School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, United Kingdom

Published online: 01 September 2026

1 Introduction

In response to the rapid growth of e-commerce across various service sectors like e-retailers, couriers, and parcel carriers, different strategies have been explored to address challenges in last-mile delivery. This aspect has gained importance in ensuring efficient service in online business deliveries. One notable trend highlighted in a report by Mintel (2022) is the surprising 17.1% increase in online grocery sales in 2021, defying initial predictions of a decline due to COVID-19 pandemic control measures. In the UK, for instance, 58% of consumers purchased groceries online during this period.

Among the most noteworthy examples of last-mile delivery, Wang et al. (2014) argue that attended home deliveries (AHD) have received considerable attention among researchers. As part of the AHD program, Agatz et al. (2013) state that perishable foods, orders with safety concerns, and/or orders with high quantities must be delivered to the customer at home. Typically, the online service company is required to deliver customers' orders within a specified time slot that they have booked for delivery. By selecting a time slot, customers are able to minimise their attending time at home, whereas companies aim to incentivise customers to select a time slot that maximises its profits. These incentives can be financial, or can be by explaining how a specific time slot will reduce the amount of driving and greenhouse gas emissions, which is environmentally friendly.

The company's decision combines two interdependent problem domains: (i) planning cost-efficient delivery routes under capacity and time-window constraints (CVRPTW), and (ii) managing demand as customer requests stochastically arrive over the booking horizon. These decisions interact because accepting a request changes the future routing feasibility and costs, while the set of feasible slots and prices shown to customers influences which requests are accepted. Therefore, a practical solution for attended home delivery should simultaneously (a) maintain a route plan with low delivery cost and (b) offer customers attractive feasible time slots with competitive delivery fees. Since e-retailers begin the booking period with uncertainty about customer numbers and spatial–temporal order patterns, indiscriminately accepting all incoming orders–particularly from outlying areas–can create later slot shortages and reduce profitability.

This work inherits the dynamic decision-making process of Abdollahi et al. (2023), maintaining a forecast route with both actual orders (time-windowed) and forecast orders that have no time-windows. Based on this forecast route, each slot's time budget (i.e., expected revenue income per unit time) can be estimated together with the incremental travel time/cost of inserting the new order. Collectively, these factors determine the opportunity cost–the potential profit foregone by allocating a slot to the current customer rather than saving it for future customers. This calculation is pivotal for implementing dynamic slot pricing. While Abdollahi et al. (2023) assume a fully persuadable customer choice, our research advances this concept by exploring the boundary of customer choices under pricing within a range $[d_{min}, d_{max}]$. This adds an extra layer of difficulty, as we are maintaining a set of forecast orders without time windows (to offer routing flexibility), while we need to guarantee the number of orders received in every time window is consistent with customer preference under a realistic (but *not fixed*; as we are optimising it) pricing policy. To deal with this difficulty, a CVRPTW-CAP [1] problem is proposed for the first time in the literature, and a carefully tailored Simulated Annealing (SA) approach is designed to solve it. The resulting approach incorporates the practical boundary of a realistic pricing scheme when estimating

[1] CAP = per-slot order caps (upper bounds) on the number of orders per time window, computed under admissible prices and checked for routing feasibility.

the opportunity cost, which improves prediction accuracy and results in better performing policies.

In addition, this work also exploits the benefits of deploying augmented slots in AHD management, and proposes a dynamic slot-combination strategy to further enhance the flexibility of the CVRPTW. Although the idea of flexible slot has been introduced by Strauss et al. (2020), our work differs in both solution methods and application. Strauss et al. (2020) use LP approximation to estimate opportunity cost, whereas our approach employs route-based dynamic programming with forecast orders and evaluates per-slot time budgets based on the current status of the delivery system. Additionally, unlike Strauss et al. (2020) who require pre-determined slot combinations for their LP model (or combinations of any two standard slots to facilitate the "customised flexible slots"), our system proposes the best slot combinations dynamically, adapting to the real-time status of the system. This dynamic nature of slot allocation in our model adds a layer of flexibility and responsiveness that Strauss et al. (2020)'s approach lacks, thereby reducing the risk of sub-optimal outcomes due to fixed, empirical initial combinations.

The contributions of this work are as follows:

- Enhanced Opportunity-Cost Estimation: The study enhances the estimation of opportunity costs by strategically distributing forecast orders across time slots, factoring in the popularity of slots. This refines the accuracy of estimating the potential costs and benefits of using different time slots.
- Dynamic Slotting with Extended Time Windows: The research deploys extended-time windows to enable dynamic slotting, thereby enhancing scheduling flexibility. Incentives are optimised dynamically to encourage customers to opt for extended-duration time slots, leading to improved slot availability and system efficiency.

The subsequent sections of the paper are summarised as follows: In Section 2, we present a comprehensive review of related research streams. Section 3 introduces a model for the AHD problem within the context of integrated demand management and dynamic routing. Further, we propose our contributions to the problem from two aspects in Sections 4 and 5, where we present a solution approach to the AHD problem and offer a novel strategy to enhance routing performance, respectively. Section 6 explains the calculation of the optimal delivery charges. The numerical results obtained from the two suggested approaches are discussed in Section 7. These results are further analysed and summarised in Section 8, followed by insights that were uncovered and future research directions. Additionally, Section 8 provides a comprehensive summary of the key findings and conclusions of this study.

2 Literature Review

Last-mile delivery involves transporting goods from a distribution centre to an end user's location. It is the final step in the delivery of goods, which is usually the most expensive portion of the process. The goal of last-mile delivery logistics is to ensure that demands are delivered in a timely, accurate, and cost-effective fashion. There are two major classes of problems associated with last-mile delivery, namely demand management and vehicle routing. Therefore, a satisfactory solution to last-mile delivery will be achieved if both objectives of both problems are taken into consideration. The following sections review the existing e-fulfilment literature, particularly attended home deliveries (AHD), which fits with the direction of our research. For a comprehensive overview of revenue management and latest development in this field, readers are advised to refer to studies (see (Agatz et al., 2013, 2008; Cordeau et al., 2023; Klein et al., 2020; Snoeck et al., 2020; Waßmuth et al., 2023)).

The literature in AHD can be divided into two dominant factors to be optimised, namely, time windows or delivery prices, by which researchers attempt to maximise an e-retailer's profitability. While the former factor implies which time windows are the best for customers of a specific area, the latter focuses on how the time windows should be priced to meet demand management targets. Both factors depend on correctly specified parameters, i.e., Opportunity Cost (OC), which calculates the monetary loss from selling an available time slot too soon. Accurately determining opportunity costs requires anticipating future orders to estimate their impact on overall profit. Various methods have been suggested in the literature to estimate the opportunity cost, which are summarised below with two different focuses, i.e., Insertion Cost (IC) and Revenue Loss (RL).

2.1 Insertion cost estimation

Insertion cost refers to the marginal increment of delivery cost with the acceptance of a new order. Apart from Asdemir et al. (2009) who use constant delivery costs, nearly all other works in AHD apply insertion heuristics to tentative routes to estimate the insertion cost. Campbell and Savelsbergh (2005), who are the first to incorporate dynamic vehicle routing with the scheduling problem for home delivery services, carry out a feasibility check on tentative routes and approximate the opportunity cost based on the insertion cost using an insertion heuristic. In cases where the current route plan allows the incoming request to be inserted into the route, they decide whether to accept or reject the request according to the profit or cost that the request may offer. This approach is extended by Campbell and Savelsbergh (2006) to incorporate customer behaviour models and pricing optimisation. A simple linear model is deployed to determine how likely a time slot is to be selected by a customer under a given price scheme. Same insertion heuristic is applied to determine feasible slots of insertion and to calculate their insertion costs, which are then fed into the pricing optimisation model to determine price reductions.

Similar insertion-based approaches are also used in later studies including (Abdollahi et al., 2023; Klein et al., 2018; Köhler et al., 2020; Yang et al., 2016) who deploy Multinomial logit (MNL) model for customer behaviour instead of simple linear models. To make the methodology anticipatory, some works introduce forecast orders into their tentative routes (e.g., Abdollahi et al., 2023) and/or progressively construct foreseeing tentative routes through iterations (e.g., Yang et al., 2016; Köhler et al., 2020). Despite the different ways of maintaining tentative routes, heuristically calculated insertion cost is one decisive factor contributing to the opportunity cost estimation.

2.2 Revenue loss estimation

Another main component of opportunity cost is revenue loss, which signifies the monetary value of the time and capacity required to incorporate the new request into the route. Instead of estimating opportunity cost, Asdemir et al. (2009) propose a dynamic programming (DP) model whose value function captures the overall anticipated profit at a system state. To simplify the problem they choose to ignore the entire routing component by assuming fixed delivery costs and capacity across all service areas and slots when formulating the DP. This makes the small scale problems solvable by backward induction, but prevents the approach from being scaled. To deal with the dimensionality, Yang and Strauss (2017) use approximate dynamic programming with linear regression, Strauss et al. (2020) use LP approximation to obtain estimated anticipated profits/opportunity costs. Both approaches are proven to work well on large scale applications.

Other closely relevant revenue loss estimation methods include route-based approach as used by (Abdollahi et al., 2023; Klein et al., 2018; Koch & Klein, 2020; Mackert et al., 2019; Yang et al., 2016). Both (Klein et al., 2018; Mackert et al., 2019) use seed-based approach for routing problem and propose mixed-integer (non-)linear programming problems that capture both vehicle routing and customer choices, with Klein et al. (2018) focusing on anticipatory pricing decisions and Mackert et al. (2019) focusing on finite-mixture multinomial logit (MNL) choice model and differentiated slotting. In Klein et al. (2018), the MILP are solved periodically with forecast orders to make the overall approach anticipatory. These are amongst the first works that combine accepted orders with a forecast of expected customers in route planning, nevertheless, the computational complexity involved in the repeat solving of the MILP prevents the approach from being applicable on large scale industrial applications.

Both (Koch & Klein, 2020; Yang et al., 2016) employ tentative route plans with time-windowed forecast orders to estimate insertion cost and revenue loss. They both maintain two (set of) tentative route plans, one with actual orders only and one consists of artificial requests to represent future customers, which are obtained through iterative simulations. On revenue loss estimation, Yang et al. (2016) simply incorporate fixed penalty terms to reflect the revenue loss when inserting the new order into a time slot in the artificial route is infeasible, whereas Koch and Klein (2020) deploy a more advanced learning approach to estimate the time budget of slots through simulation runs, which are then used to calculate revenue loss.

One fundamental difference between our approach and Koch and Klein (2020) is that they use time-windowed forecast orders and rely on simulation runs to align forecast order distribution across time slots with the optimised pricing. This learning-based approach requires intensive offline training and the result is influenced by the selection of training parameters. In addition, once converged, the number of forecast orders in each time slot is no longer adjustable according to the specific booking status of the day. In comparison, we use forecast orders without time windows, which can be moved from one slot to another so as to provide a dynamically adjustable forecast route that is responsive to every little change in the booking status.

On the other hand, Abdollahi et al. (2023) build anticipatory route plans by leveraging historical data, which inherit the number of orders and geographical location of customers without considering their chosen time windows. The forecast route is dynamically updated and re-optimised upon the acceptance of new orders, so as to provide up-to-date insertion cost and revenue loss estimations for new arrivals. As forecast orders are not time-windowed, they can be moved freely to adapt to the changes brought by the newly inserted orders through re-optimisation, and this is believed where the improved efficiency come from. Nevertheless, given that forecast orders lack specific time windows, the distribution of planned delivery times for forecast orders may deviate from customers' natural slot preferences and/or be unachievable under any pricing policies, which decrease the accuracy of insertion cost and revenue loss estimation. This paper improves Abdollahi et al. (2023) by estimating the highest committable order numbers for each time slot through simulation and incorporating them as realistic upper bounds that guide the dynamic routing optimisation.

2.3 Time window design and management

Following is a review of the literature regarding time window design and management, which is closely related to our second contribution. As a standard practice in AHD, customers are offered a range of fixed length, pre-determined time slots. Time slot availability assessment is tailored to each customer, taking into consideration the customer's location, order size, previously booked time slots and the orders collected so far, etc. According to (Lin & Mahmassani, 2002; Punakivi & Saranen, 2001), although catering to customer desires for shorter waiting times is beneficial, excessive focus on this aspect can drive up delivery charges and even lead to potential order rejections. In contrast, Gevaers et al. (2014) demonstrate that longer time windows can bolster e-retailer profitability, offering another perspective to consider.

Strauss et al. (2020) propose an approach that exploits customers' flexibility in home delivery by combining regular time windows to form flexible slots. After dispatching the delivery van, the customer is promptly informed of the specific delivery window. Discount on delivery charge is offered to promote the usage of flexible slots. A LP approximation is constructed to estimate opportunity cost. Our problem setting aligns with the work by Strauss et al. (2020), involving tailored delivery prices derived from current and future customer preferences. While as mentioned in Section 1, we use different approaches to achieve our aim of insertion cost and revenue loss estimation, and allowing dynamically proposed time window combinations according to the booking status.

Our research also fits within the category in which e-retailers can introduce flexibility into time window management by offering both extended and standard time windows. Köhler et al. (2020) propose a strategy where new customers are presented with a selection of short and/or long time windows based on factors like travel time, current booking horizon time, insertion time, and time span associated with order insertion. While their model provides fixed delivery fees for both short and long windows, disregarding potential customer preferences for lower charges to accommodate company flexibility. In our study, pricing optimisation is considered together with dynamic slotting, which incentives extended time windows so as to boost booking likelihood and ultimately leading to a higher acceptance rate.

2.4 Dynamic vehicle routing

In attended home delivery, requests come in progressively until a cut-off time. Each request has a preferred delivery time, contingent on availability, and originates from various locations and times. To keep total delivery costs low, it is crucial to have an operationally feasible delivery sequence. Dynamic routing systems are therefore vital in managing the stochastic arrival of customer requests and advising feasible slots to offer.

Building on the need for efficient dynamic routing, one strategy to minimise costs is to offer incentives for customers to select more cost-effective delivery choices. Yildiz and Savelsbergh (2020) delve into this concept within the vehicle routing problem that suggests discount offering to alter a pre-agreed delivery day to another day with customers. Dynamic programming and backward recursion are deployed to solve small scale problems with a single vehicle, assuming all orders are known beforehand. One major difference from this work to ours lies in the fact that, in Yildiz and Savelsbergh (2020), each day's delivery planing is independent from another. This makes the approach not directly applicable to AHD applications where route planing has to be done for the whole day and there are strong correlations between the delivery plan of different slots.

Further advancing the field, Keskin et al. (2023) introduce the concept of dynamic multi-period vehicle routing with touting. This approach integrates 'touting'–proactively encouraging potential customers to place orders–into vehicle routing strategies, demonstrating significant reductions in travel distances and improved efficiency in last-mile deliveries. Similarly, Çelen Naz Ötken et al. (2023) introduce a novel business model in e-grocery, emphasising profit maximisation by leveraging idle time and vehicle capacity for opportunity sales (OS). The approach engages potential customers along delivery routes with the opportunity to make purchases, enhancing sales and profit without compromising regular service. In contrast to these methods' reliance on end-of-day adjustments to fill in gaps in predetermined routes and/or engaging customers re-actively through notifications, our approach incorporates an incentivising mechanism during the customer booking phase. By encouraging customers to select slots that align with more efficient and profitable delivery routes right from the start, our method enables real-time optimisation of the delivery schedule and therefore enhances operational efficiency and flexibility.

3 Dynamic pricing formulation in attended home delivery

We consider an e-retailer providing delivery services to customers who visit the website during the booking horizon. As customers visit the website during this period, they log in, input their home addresses, complete their shopping carts, choose a preferred delivery day, and proceed to select a suitable delivery time slot. On the delivery day, vans are loaded and dispatched at the day's outset. Vans then proceed through their assigned deliveries without returning to the distribution centre between stops. At each customer's location, there exists a service time required for the delivery handover process.

Figure 1 clarifies the timeline and the two perspectives used in our model. The *booking horizon* is discretised into decision epochs ("stages") $t \in \{1, \ldots, T\}$, where T is the booking cut-off; the discretisation is chosen so that at most one customer arrival is associated with an epoch t (early epochs may be empty and later epochs denser). The *delivery day* is divided into delivery time slots ("slots") $s \in S$, which are the service windows used to execute routes. The figure also contrasts the *customer view* (available slots and delivery charges displayed) with the *system view*, where tentative route plans are maintained to assess slot feasibility as orders are accepted.

Importantly, customers must select a *specific delivery day* before the slot-feasibility and pricing routine is invoked. Therefore, the decision problem we formulate is defined *per delivery day*: for each delivery day, there is a fixed booking window that opens K days in advance (in our experiments, $K = 21$) and closes at the cut-off time on the day prior to delivery. Delivery slots are day-specific and do not overlap across delivery days; hence, we maintain an independent booking-horizon model (state and tentative route plan) for each delivery day.

Given the discrete booking epochs defined above, we model the dynamic pricing problem as a finite-horizon Markov decision process (MDP), equivalently a finite-horizon dynamic program (DP). If a customer arrives at epoch t, we denote the corresponding order by epoch index t, with basket revenue r_t. Although we model at most one arrival per epoch, no computation is performed at empty epochs; the online routine is invoked only when an order arrives. Throughout the booking horizon, the state of the system at epoch t is represented by a matrix $\mathbb{X}(t)$ with dimensions $|A| \times |S|$. The element $[a, s]$ indicates the count of accepted orders up to epoch t for delivery in area $a \in A$ and time slot $s \in S$. We represent the column-major

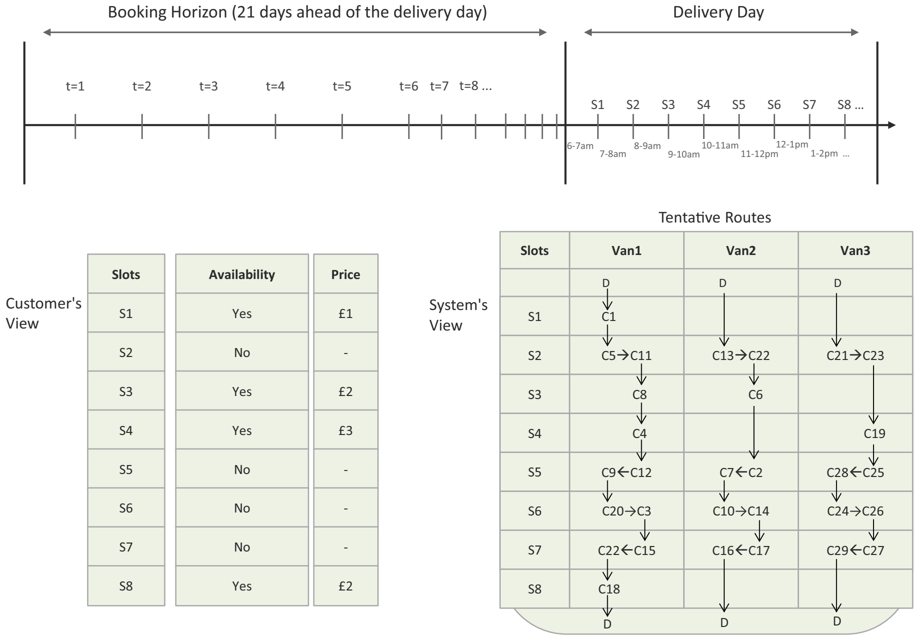

Fig. 1 Booking epochs (stages t) in the order-collection horizon versus delivery time slots (slots s) on the delivery day (routing execution); customer vs. system view

vectorisation of $\mathbb{X}(t)$ by $\vec{x}_t$ and denote the finite set of feasible states by $\mathcal{X}$, where $\vec{x}_t \in \mathcal{X}$. If a customer in area a books slot s, the system transitions to $\vec{x}_{t+1} = \vec{x}_t + \mathbf{1}_{as}$; otherwise, $\vec{x}_{t+1} = \vec{x}_t$, where $\mathbf{1}_{as}$ denotes a unit increment at (a, s).

At epoch t and for area a, let the feasible time-slot set be $F_a(\vec{x}_t) = \{s \in S \mid \vec{x}_t + \mathbf{1}_{as} \in \mathcal{X}\}$, where S is the set of delivery time slots and $\vec{x}_t + \mathbf{1}_{as} \in \mathcal{X}$ indicates that accepting (a, s) preserves routing feasibility under the CVRPTW time-window and capacity constraints. The delivery charges shown to a customer in area a are collected in $\vec{d}_{a,t} = \{d_{as,t} \mid s \in F_a(\vec{x}_t),\ d_{as,t} \in [d_s^{\min}, d_s^{\max}]\}$.

In real-world scenarios, determining the probability of a customer selecting a time slot s at epoch t, or opting not to select any time slot, is typically a challenge. In this context, we build upon the Multinomial Logit (MNL) model, as introduced by McFadden et al. (1973). This model defines the probability of a customer choosing a time slot, $P_s(\vec{d}_{a,t}; F_a(\vec{x}_t))$, or not choosing any time slot, $P_0(\vec{d}_{a,t}; F_a(\vec{x}_t))$, at epoch t, based on the feasible set of available time slots and the delivery charge vector $\vec{d}_{a,t}$ offered to the customer (in area a). Since the probabilities sum to unity, we have $P_0(\vec{d}_{a,t}; F_a(\vec{x}_t)) = 1 - \sum_{s \in F_a(\vec{x}_t)} P_s(\vec{d}_{a,t}; F_a(\vec{x}_t))$. For simulation purposes, we adopt this model, which posits that these probabilities conform to the following equation:

$$P_s(\vec{d}_{a,t}; F_a(\vec{x}_t)) = \frac{\exp(\beta_0 + \beta_s + \beta_d d_{as,t})}{\sum_{k \in F_a(\vec{x}_t)} \exp(\beta_0 + \beta_k + \beta_d d_{ak,t}) + 1}. \tag{3.1}$$

where β_0, β_s, and β_d represent the base utility of all choices, the utility corresponding to slot s, and the utility sensitivity to the delivery charge $d_{as,t}$, respectively (See Appendix D, Table 6). These β parameters capture the utilities that customers consider when choosing a time slot. To derive the values of the β parameters, we leveraged historical purchasing data and employed an optimisation process that sought to maximise the likelihood of observed customer choices. This enabled us to estimate the values of these key parameters accurately. We use $P_0(\vec{d}_{a,t};\, F_a(\vec{x}_t))$ to denote the probability of a customer not choosing any of the time slots offered to them, i.e., not making a booking. Note that delivery charges are bounded within pre-defined limits, i.e., $d_s^{\min} \le d_{as,t} \le d_s^{\max}$. The service offering is not degraded: the same routing feasibility constraints determine which slots can be offered (no hidden changes to delivery quality or eligibility). Moreover, $\beta_d < 0$, so higher delivery fees *decrease* choice probabilities; any acceptance gains produced by the proposed approach are therefore driven primarily by feasibility/availability expansion rather than by price increases.

The parameter λ_t denotes the probability of an arrival in epoch t, obtained by discretising a time-dependent Poisson arrival process so that at most one arrival is associated with an epoch. This process starts with a lower arrival rate and gradually escalates as the booking horizon advances. We assume that each customer will attempt to book only one time slot at a given epoch. Additionally, μ_a represents the probability that an arrival is from area $a \in A$, where A encompasses all possible areas.

Under these assumptions, we write the Bellman recursion following Yang et al. (2016). Let $V_t(\vec{x}_t)$ denote the value function in the DP formulation for the state $\vec{x}_t$ at epoch t. This function represents the maximum expected potential profit-to-go from epoch t until the final epoch (T); maximised over all possible choices of delivery-price vectors offered to customers over all future epochs after t, while assuming customers arrive according to the Poisson process described above. Therefore, the objective is to determine the optimal delivery-price vector $\vec{d}^*_{a,t}$ to offer to the customer at epoch t to maximise the expected profit $V_t(\vec{x}_t)$.

$$
\begin{aligned}
V_t(\vec{x}_t) = \max_{\vec{d}_{a,t}} \Bigg\{ & \left(\sum_a \lambda_t \mu_a \sum_{s \in F_a(\vec{x}_t)} P_s(\vec{d}_{a,t};\, F_a(\vec{x}_t)) \Big[r_t + d_{as,t} + V_{t+1}(\vec{x}_t + \mathbf{1}_{as}) \Big] \right) + \\
& \Big[1 - \sum_a \lambda_t \mu_a \sum_{s \in F_a(\vec{x}_t)} P_s(\vec{d}_{a,t};\, F_a(\vec{x}_t)) \Big] V_{t+1}(\vec{x}_t) \Bigg\} \\
= & \left\{ \max_{\vec{d}_{a,t}} \sum_a \lambda_t \mu_a \sum_{s \in F_a(\vec{x}_t)} P_s(\vec{d}_{a,t};\, F_a(\vec{x}_t)) \Big[r_t + d_{as,t} - \big(V_{t+1}(\vec{x}_t) - V_{t+1}(\vec{x}_t + \mathbf{1}_{as}) \big) \Big] \right\} + V_{t+1}(\vec{x}_t), \\
& \forall \vec{x}_t \in \mathcal{X},
\end{aligned}
\tag{3.2}
$$

with the boundary condition being:

$$
V_{T+1}(\vec{x}_{T+1}) = -C(\vec{x}_{T+1}) \quad \forall \vec{x}_{T+1} \in \mathcal{X} \tag{3.3}
$$

where $C(\vec{x}_t)$ represents the minimum cost for solving the CVRPTW, ensuring the delivery of all orders within the chosen time slots, in which $\vec{x}_t$ belongs to the set of feasible delivery schedules $\mathcal{X}$. In cases where there is no feasible solution to the CVRPTW, $C(\vec{x}_t)$ is set to infinity ($+\infty$).

The state transition above captures permanent insertion: once an order for area a, slot s is accepted, it is added to $\vec{x}_t$ and replaces a forecast order in the tentative route. The value function $V_{t+1}(\cdot)$ reflects this in two respects: the feasible offer set $F_a(\vec{x}_t)$ is updated to $F_a(\vec{x}_t + \mathbf{1}_{as})$ so that only slots satisfying the revised CVRPTW capacity and time-window constraints remain available for future customers; the continuation value $V_{t+1}(\vec{x}_t + \mathbf{1}_{as})$ accounts for the updated routing cost after insertion, which accumulates through to the terminal cost $C(\vec{x}_{T+1})$ in the boundary condition (3.3).

Note that $P_s(\vec{d}_{a,t};\, F_a(\vec{x}_t))$ denotes the probability in Equation (3.1), where a customer selects time slot s from $F_a(\vec{x}_t)$ given $\vec{d}_{a,t}$. The DP formulation mentioned above encounters two challenges that render it computationally complex. Firstly, the size of the state space $\mathcal{X}$ expands exponentially with the number of area–slot combinations and the feasible order-count levels, resulting in computational intractability. Secondly, achieving an optimal solution for the vehicle routing problem (3.3) becomes difficult when dealing with large-scale problems. To address the first challenge, obtaining a reliable estimate of the opportunity cost is crucial for determining the optimal delivery-price vector $\vec{d}^*_{a,t}$. This is because the decision regarding $\vec{d}^*_{a,t}$ involves a balance between the immediate profit $(r_t + d_{as,t})$ and the expected opportunity cost $\big(V_{t+1}(\vec{x}_t) - V_{t+1}(\vec{x}_t + \mathbf{1}_{as})\big)$. Further elaboration on approximating the opportunity cost is provided in Section 4. In Section 5, we delve into the exploration of routing efficiency enhancements through augmented time windows, addressing the second challenge.

4 Route-based demand management and opportunity-cost approximation

At a high level, the solution method proceeds as follows. First, an initial forecast route is constructed from historical demand information using forecast orders (Section 4). Second, slot-wise order caps are estimated (Section 4.1) and used to regularise the placement of forecast orders in the tentative route, solved via simulated annealing (Section 4.2). Third, when a customer arrives, augmented time windows are formed by merging contiguous standard slots that share the same insertion context (Section 5), and the opportunity cost is approximated through insertion cost and revenue loss, from which delivery prices are computed (Section 6). If a slot is chosen by the customer, the selected forecast order is replaced by the actual order and the route is re-optimised (Section 4.2).

In Equation (3.2), the term $\big(V_{t+1}(\vec{x}_t) - V_{t+1}(\vec{x}_t + \mathbf{1}_{as})\big)$ represents the difference in profit at time step $t+1$ between not allocating slot s in area a to the current customer and allocating it. This difference illustrates the potential profit loss, which is commonly referred to as OC. A substantial challenge in dynamic pricing problems is accurately approximating the OC, which critically relies on a crucial factor known as the revenue loss (RL). Much prior research has focused on addressing the estimation of RL, as seen in works by (Klein et al., 2018; Strauss et al., 2020; Yang & Strauss, 2017).

To address the demand challenges inherent in the AHD problem, we adopted the methodology proposed by Abdollahi et al. (2023). Specifically, this entails continually tracking customer arrivals and their bookings across the horizon while maintaining (and updating) two lists of actual orders (with time windows) and forecast orders (without time windows)– and running a warm-started simulated annealing (SA) process that is invoked upon each accepted order to refine a current best route plan so it remains feasible and near-optimal for the evolving set of orders. The whole process of this approach is depicted in Figure 2. This method involves the creation of a forecast delivery route plan using historical data as a warm start at the beginning of the booking period. Forecast orders are generated using a Simple Moving-Average (SMA) model to predict the total number of orders. Each forecast order's attributes, including address, order size and revenue, are randomly simulated based on historical data. Previous selected time slots are intentionally ignored when generating forecast orders. Following Abdollahi et al. (2023), forecast orders retain attributes that can be inferred relatively reliably from historical data, such as customer location, order size and revenue, while their delivery-slot choices are left unspecified because these depend on the pricing policy and the evolving booking status.

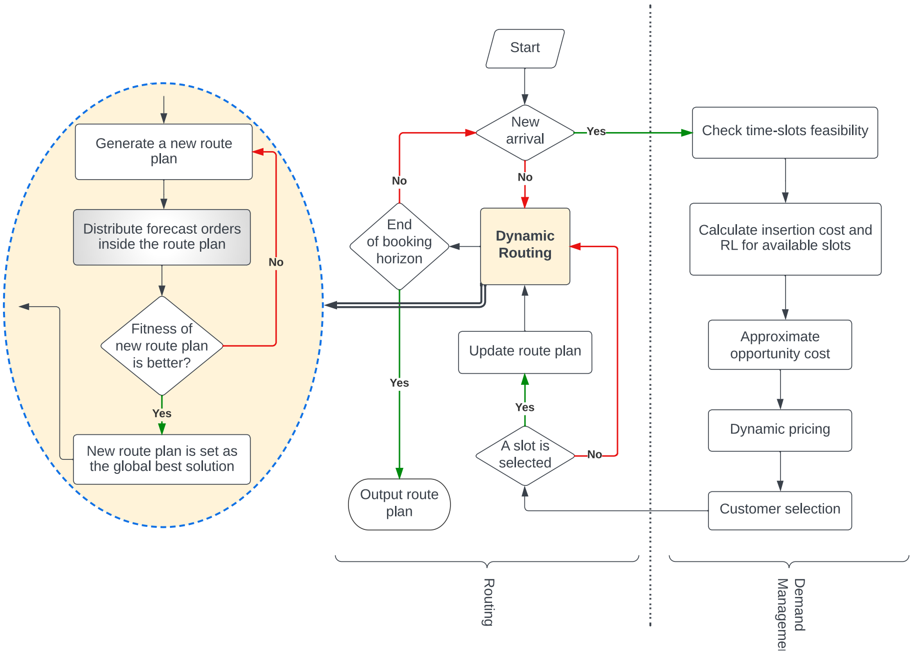

Fig. 2 Overview of the integrated dynamic routing and demand management problem. Incorporation of distribution mechanisms into the dynamic routing system is detailed in Section 4

Upon each new customer arrival, their order is temporarily inserted into the forecast route in each time slot, to test each time slot for feasibility, IC and RL calculation. The result informs the OC estimation for dynamic pricing, and the resulting slot availability, and prices are displayed to the customer for their choice. If the customer selects one time slot, the order becomes an actual order and is permanently inserted into the forecast route to replace one forecast order ("Update route plan" in Figure 2). The dynamic routing system is therefore re-optimised in response to new insertions/replacements ("Dynamic Routing" in Figure 2). This predictive route plan serves as a dynamic approximation of the final route, and is used to compute $OC = IC + RL$ for each potential order.

To keep the forecast route plan policy-agnostic, forecast orders are introduced without fixed time windows. Here, policy-agnostic means that the tentative route is not tied to the slot distribution implied by any particular pricing rule. This preserves the geographical structure needed for routing while allowing the heuristic to place forecast orders in whichever slots that best support an efficient provisional route. These forecast orders, therefore, act as placeholders that are progressively replaced by actual customer orders as bookings are made.

However, due to the ability to freely allocate forecast orders across time windows, the time-allocations given to those forecast orders along the generated route might significantly deviate from the original popularity of time slots. This deviation can render the forecast delivery plan unrealistic, especially when employing dynamic pricing strategies with predetermined price limits. For example, when forecast orders are modelled without time windows, the optimal VRP solution may allocate 20 deliveries to the 6–7 a.m. slot given the available fleet. While this allocation is operationally feasible, the 6–7 a.m. slot is typically unpopular; consequently, it is unlikely that pricing alone can induce 20 customers to select it over the booking horizon.

This creates a mismatch between the VRP solution obtained without time-window constraints and realistic customer choice behaviour. To address this, we estimate the maximum demand shift that any pricing policy can plausibly generate across slots and embed this limit into the routing problem. The resulting VRP still does not impose fixed time windows on individual orders (i.e., orders may be reassigned across slots), but each slot is subject to an upper bound on the number of deliveries–potentially below its purely logistical capacity. This mechanism incorporates realistic slot popularity into routing decisions while preserving flexibility by retaining forecast orders without predefined time windows.

4.1 Order upper bound for every time slot

Assuming we possess knowledge of the optimal dynamic pricing policy, predicting the slot choices made by customers under this policy becomes relatively straightforward, thus enhancing the accuracy of the final route predictions. However, a significant challenge arises from the fact that the optimal dynamic pricing policy is not known in advance and is inherently uncertain. In practical scenarios, retailers commonly impose specific limitations on the delivery prices they offer. These limitations can take the form of a defined continuous pricing range, like $[d_s^{\min}, d_s^{\max}]$, as observed by Yang and Strauss (2017). Alternatively, retailers might choose to offer delivery prices from a predetermined set of discrete levels, such as $\{d_s^1, d_s^2, \dots, d_s^l\}$, as explored by Koch and Klein (2020).

The objective of this section is to determine the upper bound on the number of orders that can feasibly be accepted for each time slot, considering the specified price limits. We refer to this as the "order capacity" for each time slot s, denoted by $\overrightarrow{CAP}_s$. Also, $\overrightarrow{CAP}$ denotes the vector of all per-slot bounds ($\overrightarrow{CAP} = \{\overrightarrow{CAP}_s \mid s \in S\}$). This determination allows us to evaluate the degree to which dynamic pricing, operating within specified upper and lower price limits ($[d_s^{\min}, d_s^{\max}]$), can influence customer demand. For each time slot s, computing $\overrightarrow{CAP}_s$ involves setting the price of slot s to its lowest possible value, $d_s^{\min}$, while assigning the maximum possible price, $d_s^{\max}$, to all other slots:

$$\overrightarrow{Price}(s') = \begin{cases} d_s^{\min} & \text{if } s' = s, \\ d_s^{\max} & \text{otherwise.} \end{cases} \tag{4.1}$$

This setup explores the highest potential share for slot s to be chosen under the choice model used in this study (MNL with selection probabilities in Eq. (3.1)).

Given the price setting of (4.1), a simulation study with historical data is carried out to calculate $\overrightarrow{CAP}$ following Algorithm 1, as a pre-processing step before receiving any real customer arrivals. Each Sim_i here denotes a full run following the procedures of Figure 2 except for the OC estimation and pricing parts. Within each simulation run (Sim_i), a complete booking period is emulated using a time-dependent Poisson process to model stochastic customer arrivals. An explicit dynamic routing and order replacement system (the left of Figure 2) is deployed to advise on routing feasibility. The fixed pricing in (4.1) is applied to the available slots, which replaces the three middle blocks on the "demand management" side of Figure 2. Customer choice is determined through Eq. (3.1). The number of accepted orders for each time slot s is logged to facilitate the subsequent calculation of $\overrightarrow{CAP}_s$ in Algorithm 1.

Algorithm 1: Calculate upper-bound vector ($\overrightarrow{CAP}$) for all slots

Input : Number of runs: $nRun$;
Available slots: S;
Slot price range: $[d_s^{\min},\ d_s^{\max}]$
Output: Set of upper bounds: $\overrightarrow{CAP}$.

$\overrightarrow{CAP} \leftarrow [0, 0, \ldots, 0]$; `// Initialise upper-bound vector of all slots`
for *each time slot* $s \in S$ **do**
  Initialise all slots' prices using (4.1)
  $\overrightarrow{\#Orders} \leftarrow [0, 0, \ldots, 0]$; `// Accepted orders in slot` s `per run`
  for $i \leftarrow 1$ *to* $nRun$ **do**
    Run simulation Sim_i for a full booking period ; `// Offer fixed prices from` $\overrightarrow{Price}$ `to all arrivals (any area` a`, time` t`).`
    $\overrightarrow{\#Orders}(i) \leftarrow$ final number of accepted orders in slot s during Sim_i
  $\overrightarrow{CAP}_s \leftarrow \left\lceil \text{avg}(\overrightarrow{\#Orders}) \right\rceil$; `// Ceiling of the mean`

For each slot s, we run $nRun = 100$ independent booking-horizon simulations. Prices are fixed per Eq. (4.1) with $d_s^{\min} = -10$ for the tested slot and $d_k = d_k^{\max}$ for all $k \in S \setminus \{s\}$. The slot set starts with $|S| = 17$ non-overlapping one-hour windows; availability may decrease during a run as slots fill. Arrivals follow the discretised time-dependent Poisson process with per-epoch arrival probabilities λ_t and area mix μ_a defined earlier; choices follow the MNL in Eq. (3.1) using the estimated β parameters (see Appendix D, Table 6). Order locations, sizes, and revenues are sampled from the historical dataset in Section 7.1. Routing feasibility and accept/reject outcomes are evaluated by the dynamic-routing module in Fig. 2. For each run we record the number of *accepted* orders in slot s; $\overrightarrow{CAP}_s = \lceil$mean over $nRun$ runs$\rceil$. We fix the random seed for each s to enable exact replication.

Note that while analytical calculations based on these equations allow us to predict the selection probability for each delivery slot and estimate the total number of arrivals, they do not fully account for the complexities of the routing problem, such as the physical capacity of delivery vans and the actual use of time slots. Therefore, we employ simulations to incorporate these critical factors, capturing the size and location of orders in a way that analytical methods cannot, thereby providing a more accurate estimation of $\overrightarrow{CAP}_s$ in the context of real-world routing constraints.

A series of such simulations, as outlined in Algorithm 1, is performed for every slot s. The value of $\overrightarrow{CAP}_s$ is then calculated as the ceiling of the average number of accepted orders in slot s by the conclusion of the booking horizon. This entire procedure is reiterated for all slots, with multiple simulations conducted to derive an empirical upper bound for each slot.

The resulting vector of upper bounds ($\overrightarrow{CAP}$) is then integrated into the CVRPTW to form a new variant named the CVRPTW–CAP problem. This problem aims to identify the optimal routes that serve all actual orders within their committed time slot and all forecast orders in any time slot, while ensuring the number of orders in every time slot does not exceed the prescribed upper bound $\overrightarrow{CAP}_s$. This is a novel modification to the standard CVRPTW to account for the expected spread of orders under different pricing policies, without directly controlling the time-slot choices of forecast orders. A MILP model is developed in this work for the CVRPTW–CAP (Appendix A). Due to its NP-hardness, this model cannot be solved for practical usage, especially in an online decision-making environment with dynamic orders.

For each slot s, $\overrightarrow{CAP}_s$ is a feasibility upper bound computed at the most slot-favourable admissible prices (set d_s to its lower bound and all other slots to their upper bounds) and evaluated with routing feasibility. Under our price-sensitive MNL, this choice maximises the attainable share for s within the admissible range; the resulting accepted-orders statistic therefore provides an upper bound (in expectation) on the volume that can be induced into slot s under any admissible pricing policy. We use the mean accepted count as a stable planning cap (rather than a high percentile), which avoids excessive conservatism and keeps the cap policy-agnostic. $\overrightarrow{CAP}$ regularises forecast-route construction only (hard cap in CVRPTW–CAP; soft penalty in SA); prices are always set from $OC = IC + RL$ on the current feasible plan, not from $\overrightarrow{CAP}$. We keep $\overrightarrow{CAP}$ policy-agnostic: while tighter per-slot forecasts could be obtained by simulating under a fixed pricing rule, doing so would couple capacity control to that rule and risk reinforcing it during search.

4.2 Simulated annealing (SA) for CVRPTW-CAP

To efficiently solve the CVRPTW-CAP, we employ Simulated Annealing (SA) as a heuristic. SA is used to find the initial (sub-)optimal routes with forecast orders, and is called upon the acceptance of every actual order to re-optimise the route. Figure 2 outlines the modifications we introduce to the dynamic routing block of Abdollahi et al. (2023)'s approach. The SA, as presented in Algorithm 2, is employed to solve the dynamic routing problem that is highlighted in Figure 2.

The parameter setup for the Simulated Annealing (SA) algorithm, characterised by an initial temperature ($T_0 = 1000$), a cooling rate ($\alpha = 0.99$), a minimum temperature threshold ($T_{\text{min}} = 0.1$), and iterations per temperature level ($N_{\text{iter}} = 10 \times \sqrt{\text{number of orders}}$), is selected to efficiently tackle the dynamic routing problem inherent in CVRPTW-CAP. While this configuration may not be universally optimal, given that the primary aim extends beyond merely optimising dynamic routing via SA, it constitutes a reasoned compromise. Resulting from extensive simulations and analytical assessments, this configuration has effectively navigated the complexities of dynamic routing.

Algorithm 2: Simulated Annealing for CVRPTW-CAP

Input : Current route plan with new order integration ($\text{Sol}_{\text{current}}$), $\overrightarrow{CAP}$, initial temperature ($T_0 = 1000$), cooling rate ($\alpha = 0.99$), minimum temperature threshold ($T_{\min} = 0.1$), iteration per temperature level ($N_{\text{iter}} = 10 \times \sqrt{\text{number of orders}}$).

Output: Optimised route with updated arrival times.

Initialise $T \leftarrow T_0$
Determine initial cost C_{current} of $\text{Sol}_{\text{current}}$
Set $\text{Sol}_{\text{best}} \leftarrow \text{Sol}_{\text{current}}$
while $T > T_{min}$ **do**
  for $i \leftarrow 1$ *to* N_{iter} **do**
    Perturb $\text{Sol}_{\text{current}}$ to generate new service sequence Sol_{new}
    Calculate earliest ($\vec{E}$) and latest ($\vec{L}$) arrival times according to the given sequence Sol_{new}
    Distribute forecast orders in Sol_{new} using Algorithm 3, obtaining balanced arrival time $\vec{E}_b$
    Evaluate C_{new}, incorporating a linearly scaled penalty for each slot where $\#Orders(s) > CAP_s$
    if $C_{new} < C_{current}$ **then**
      $\text{Sol}_{\text{current}} \leftarrow \text{Sol}_{\text{new}}$, adopting the new solution
      if $C_{new} < C_{best}$ **then**
        $\text{Sol}_{\text{best}} \leftarrow \text{Sol}_{\text{new}}$, updating the best solution
    else
      if $rand(0, 1) < e^{-(C_{new} - C_{current})/T}$ **then**
        $\text{Sol}_{\text{current}} \leftarrow \text{Sol}_{\text{new}}$ `// Adopt the new solution and continue with the next iteration of the loop`
      `// Retain` $\text{Sol}_{\text{current}}$ `and proceed with the next iteration of the loop`
  $T \leftarrow \alpha \cdot T$, cooling down the system
return Sol_{best}

To impose the order limit CAP_s as specified by constraint (A.12), we introduce a forecast-order distribution step using Algorithm 3 into SA. During the operation of SA, when generating a random tentative route plan, Algorithm 3 (re-)assigns arrival time to each order in the route considering the CAP_s values. In case when constraint (A.12) cannot be met after (re-)assigning the visiting times, penalty terms are added to the objective to push the SA search to feasible directions. In Section 4.2.1, we explain the employed forecast order distribution approach in details.

4.2.1 Forecast order distribution of a route plan

Recall that CAP_s represents the maximum number of orders expected in slot s. These orders are to be serviced by the whole fleet, which normally consists of multiple delivery vans. Nevertheless, during the distribution of orders across slots in the SA process, we work on the route of each delivery van sequentially. To achieve a more balanced distribution across all delivery vans, we introduce a refined CAP_s per van, denoted by $CAPpV_s(k)$, to reflect the slot-specific order limit in the route of delivery van k, based on the number of orders currently assigned to that particular van:

$$CAPpV_s(k) = \lceil CAP_s \cdot \frac{\text{length}(\overrightarrow{order}(k))}{\sum_s CAP_s} \rceil \tag{4.2}$$

where $\text{length}(\overrightarrow{order}(k))$ represents the number of orders for van k at time step t. Ceiling function $\lceil \cdot \rceil$ is applied to ensure that $CAPpV_s(k)$ takes integer values. $CAPpV_s(k)$ will not exceed CAP_s because the length$(\overrightarrow{order}(k))$ is never higher than $\sum_s CAP_s$. Indeed, $\sum_s CAP_s$ is considerably higher than the actual number of orders that will be received at the end of booking period, as CAP_s represents the upper-bound for the total orders that can be delivered in slot s. We introduce $CAPpV_s(k)$ to mirror the structure of CAP_s while tailoring it to van k. This approach captures the highest expected order count for slot s on van k, adhering to the same pattern as in CAP_s, contingent on the current order count.

Algorithm 3 employs Equation (4.2) to guide re-assignment of earliest arrival time of orders, denoted as $\vec{E}_b$, on a tentative route plan in SA. Note that the earliest arrival time ($\vec{E}$) signifies the earliest moment when a delivery can arrive at the customer node according to the current working route plan, ensuring that all routed orders before it (including itself) are not serviced before the start of their chosen slots. On the contrary, the latest arrival time ($\vec{L}$) denotes the latest allowable time by which the delivery van must arrive at the customer node, ensuring that all routed orders after it (including itself) are not serviced later than the end of their chosen slots. A route plan is feasible if $\vec{E} \leq \vec{L}$, i.e., the earliest arrival time is no greater than the latest arrival time for all orders on the route.

Delving into further details, when the current time slot approaches its capacity threshold ($CAPpV_s$), Algorithm 3 assesses the feasibility of shifting the next forecast order and all subsequent orders to later time slots (lines 10 to 19 in Algorithm 3). The feasibility of this shifting operation hinges on the flexibility of subsequent actual and/or forecast orders, which is determined by the slack between their earliest and latest arrival times (line 12). In line 14, the required shift size is calculated by determining the time difference between the beginning of the next time slot and the earliest arrival time of the forecast order under consideration. If the required shift size is no greater than the minimum slack of all subsequent orders, the current shifting of the forecast order is deemed valid and performed (line 15-19). However, if the shifting is infeasible (normally due to the presence of actual orders), the earliest arrival time remains unchanged. Note that this infeasibility only signals potential violation of the CAP_s but not necessarily leads to it, as the CAP_s is imposed collectively to the routes of all vans. Suppose one van has less than expected number of orders assigned to the same slot, the overall route plan may still be feasible.

Algorithm 3: Distribution of forecast orders in time slots based on $\overrightarrow{CAP}$

Input : For each van k: order sequence $\overrightarrow{order}(k)$;
Earliest and latest arrival times for each van k and order i: $\vec{E}(k,i)$ and $\vec{L}(k,i)$;
Slots upper-bound: $\overrightarrow{CAP}$;
Available slots: S;
Start and end time of all slots: $\vec{U}^B$, $\vec{U}^E$
Output: Balanced earliest arrival times $\vec{E}_b$.

Initialise $\vec{E}_b \leftarrow \vec{E}$
for $k \leftarrow 1$ *to number of vans* **do**
  Initialise $RS \leftarrow [0, 0, \ldots, 0]$; // Record number of orders in each slot
  Calculate refined $CAPpV_s(k) \leftarrow \lceil CAP_s \cdot \frac{\text{length}(\overrightarrow{order}(k))}{\sum_s CAP_s} \rceil$; // Refined CAP_s for van k
  for $i \leftarrow 1$ *to length*$(\overrightarrow{order}(k))$ **do**
    $s \leftarrow$ find_slot_number$(\vec{E}_b(k,i))$; // Find the corresponding slot number for $\vec{E}_b(k,i)$
    $RS(s) \leftarrow RS(s) + 1$; // Add order i to slot s
    if i *is a forecast order* **then**
      if $RS(s) > CAPpV_s(k)$ **then**
        if s *is not the last slot in* S **then**
          for $n \leftarrow i + 1$ *to length*$(\overrightarrow{order}(k))$ **do**
            $\text{Slack}_\text{n} \leftarrow \vec{L}(k,n) - \vec{E}_b(k,n)$; // Calculate slack of subsequent orders
          $\text{Min\_slack} \leftarrow \min_{i<n\leq \text{length}(\text{order}(k))} \text{Slack}_\text{n}$
          $\text{Shift} \leftarrow U^B_{s+1} - \vec{E}_b(k,i)$; // Calculate required shift size
          if Shift $\leq$ Min_slack **then**
            $RS(s) \leftarrow RS(s) - 1$; // Remove order i from slot s
            $RS(s+1) \leftarrow RS(s+1) + 1$; // Add order i to slot $s+1$
            for $h \leftarrow i$ *to length*$(\overrightarrow{order}(k))$ **do**
              $\vec{E}_b(k,h) \leftarrow \vec{E}_b(k,h) + \text{Shift}$; // Delay following orders by Shift

return $\vec{E}_b$

Note that although SA approach is used in this article to solve the CVRPTW-CAP, the Algorithms 1 and 3 as proposed here can be integrated seamlessly with any existing dynamic routing solvers/heuristics for CVRPTW used by a grocer. Algorithm 1 should be called offline before the start of booking horizon, using historical data to predict CAP_s through simulation. Algorithm 3, on the other hand, contributes to the online distribution of forecast orders, which is called upon obtaining a tentative route using any routing package and when evaluating the tentative route's fitness. Adopting this approach, an enhanced forecast route can be obtained which aligns with the slot popularity, taking into consideration the influences of the future pricing optimisation. Based on this enhance forecast route, more accurate opportunity cost estimations can be achieved with a hope to improve the optimisation of dynamic pricing.

At each acceptance we warm-start SA from the incumbent plan $\text{Sol}_{\text{current}}$, restrict moves to time-window–feasible relocate/shift/2-opt, and invoke the forecast-order rebalancing step only when a slot approaches its cap. We then recompute $OC = IC + RL$ on the updated best route. This incremental update keeps OC aligned with the live plan while achieving per-arrival runtimes < 0.014 s across areas (see Figure 9). For clarity, the pricing layer uses $OC = IC + RL$ recomputed on the updated feasible plan returned by Algorithm 2; the vector $\overrightarrow{CAP}$ only regularises forecast-route construction (hard cap; soft SA penalty) and is not used directly in the pricing equation.

5 Augmented Time Window

The order distribution approach as proposed in Section 4 aims to improve opportunity cost estimation when forecast orders without time windows are used to assist route planning. In this section, on the other hand, we aim to further preserve the flexibility provided by these forecast orders (not time-windowed) in the actual route, so as to improve the overall routing efficiency. Remember that the decision framework proposed by Abdollahi et al. (2023) invoke a replacement process when a new actual order is committed. With this replacement, the system transits from having a forecast order without a predetermined time window to having an actual order with a fixed, one-hour time window. However, it is important to recognise that an order without a designated time window offers more flexibility in routing optimisation, compared to an order with a fixed time window. To preserve routing flexibility when forecast orders are progressively replaced by realised customer orders, we introduce Augmented Time Windows (ATW). An ATW is an optional service window constructed by merging contiguous standard slots around the selected insertion position. To avoid implicitly making wider windows more attractive to customers, we apply a penalty parameter γ so that longer ATWs do not receive an unwarranted utility advantage. These extended time windows are offered in addition to the standard ones and incentivised through dynamic pricing, so as to encourage customers to select.

The existing literature on variable or extended-duration time windows in the context of the AHD problem has mainly explored two strategies: static short and long time slots (as shown by Klein et al. (2020)), and the combination of less popular slots with more popular ones to improve slot utility (as discussed by Strauss et al. (2020)). Differing from these methods, our study introduces a dynamic approach for slot combination, building upon the foundation laid by Abdollahi et al. (2023). Our approach leverages the flexibility inherent in forecast orders without fixed time windows.

Every time a customer places an order, we dynamically determine the optimal combinations of the existing set of Standard Time Windows (STW) to form Augmented Time Windows (ATW). Figure 3 provides an example of how ATWs are formed from adjacent STWs. Let us consider a scenario where a new actual order j is inserted into the tentative route during the dynamic routing process. In Figure 3, j_s^* represents the index of the most appropriate forecast order to remove when inserting order j into slot s. Here, i and $i+1$ represent the indices of the preceding and succeeding orders on the route, respectively, between which order j is being inserted. The insertion cost of replacing j_s^* with j is denoted as IC_s.

It is evident that if we aim to service new order j in slots 1, 2, or 3, the best forecast order to remove and the optimal insertion position on the route remain the same. This implies that even after this replacement and insertion, actual order j retains the flexibility to move across these three slots. However, if only STWs are available and the customer later selects slot 2, for instance, order j becomes constrained by the start and end time of slot 2 (i.e., [7am, 8am]). Consequently, this reduces the route's flexibility as any future new orders to be serviced in slot 3 cannot be inserted into gaps before order j, even if travel time allows both orders to be accommodated within a one-hour slot.

To maintain routing flexibility, we consolidate slots 1, 2, and 3 into an ATW that spans from 6am to 9am, as highlighted in green in Figure 3. If selected, order j can be fit into any

STW ATW

slot	1	2	3	4	5	6	7	8	9	10	11	12	13
j*	39	39	39	20	48	48	11	6	6	6	39	48	6
i	45	45	45	5	24	24	20	44	44	44	45	24	44
i+1	58	58	58	61	22	22	41	29	29	29	58	22	29
IC	13.42	13.42	13.42	12.49	6.80	6.80	4.51	-3.03	-3.03	-3.03	13.42	6.80	-3.03
begin_s	6 am	7 am	8 am	9 am	10 am	11 am	12 pm	1 pm	2 pm	3 pm	6 am	10 am	1 pm
end_s	7 am	8 am	9 am	10 am	11 am	12 pm	1 pm	2 pm	3 pm	4 pm	9 am	12 pm	4 pm

Fig. 3 Constructing augmented time windows from standard time windows

time between [6am, 9am], making the tentative route more adaptable for absorbing a higher number of future orders. This strategy helps enhance the overall efficiency and responsiveness of the routing system.

In more generic terms, to generate ATW from STW, we examine the indices of the preceding and succeeding orders (i and $i+1$) during the order insertion and/or replacement process. The time slots in STW that have matching values for i and $i+1$ are considered the desired time slots to combine and form a new extended time window to add to ATW. Notably, since ATW are generated from adjacent time slots, having them in following routing (re-)optimisation does not burden the dynamic routing package with maintaining dis-connected aggregated time windows. Instead, an order selecting ATW simply becomes a standard order in the follow up routing (re-)optimisation with a longer time window. The utility parameterisation for ATWs (including the explicit formula and interpretation) is given in Section 7.4; pricing remains unchanged and follows the same MNL/pricing equations as for $STWs$ (see Section 6).

Figure 4 depicts the integration of dynamic routing and demand management, including both forecast order distribution into dynamic routing (Section 4) and ATW calculation (Section 5). This integration results in an improved route-based dynamic pricing process. Note that the process of creating ATWs are carried out after order distribution procedures of Section 4, on the already-revised earliest start time $\vec{E}_b$. So the resulting ATW will not include combinations that lead to CAP violations.

6 Calculate optimal delivery charges

After aligning the earliest arrival times of all orders ($\vec{E}_b$) using Algorithm 3 in Section 4, we proceed to calculate the opportunity cost $OC = IC + RL$ for all slots $S = STW \cup ATW$ as a result of formation of ATW as elaborated in Section 5.

To calculate the insertion cost (denoted as IC), we introduce the term $DC_t(\vec{x}_t, \vec{f}_t)$, which represents the total 'Delivery Cost' obtained from the dynamic routing system. Here, $\vec{x}_t$ denotes the vector of all accepted orders, whereas $\vec{f}_t$ is defined as a vector of all forecast orders that remain in the route at time t. If it is not feasible to fulfil all orders $(\vec{x}_t, \vec{f}_t)$ due to capacity and time slot constraints, we define $DC_t(\vec{x}_t, \vec{f}_t) = \infty$. The insertion cost of accepting the order (a, s) at time t is denoted by $IC_t(\vec{x}_t, \vec{f}_t, \mathbf{1}_{as})$, which is calculated as

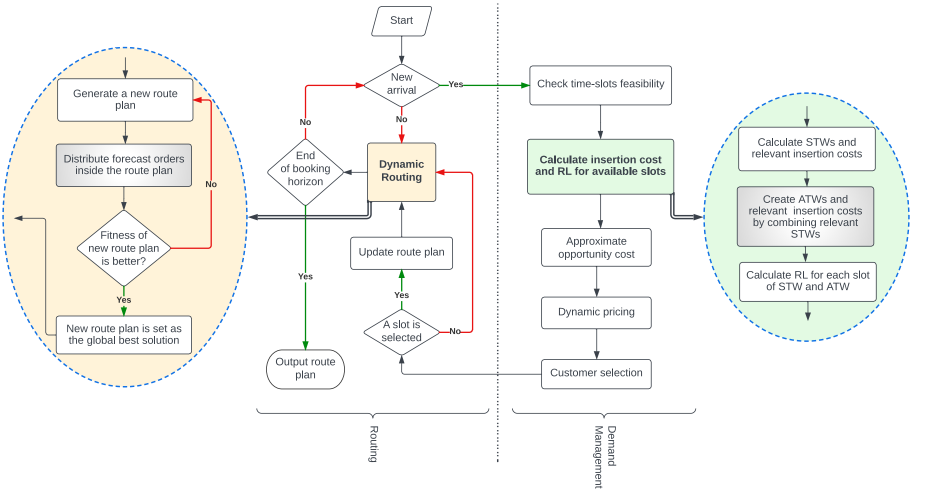

Fig. 4 Creation of ATW using the existing STW to enhance the efficiency of the routing process as shown in the right oval and discussed in Section 5

follows:

$$IC_t(\vec{x}_t, \vec{f}_t, \mathbf{1}_{as}) = \begin{cases} \min\limits_{j_s \in \vec{f}_{rad}} \left[DC_{t+1}(\vec{x}_t + \mathbf{1}_{as}, \vec{f}_t - \mathbf{1}_{a^*}) - DC_{t+1}(\vec{x}_t, \vec{f}_t) \right] & \text{if } \vec{f}_t \neq NULL \\ DC_{t+1}(\vec{x}_t + \mathbf{1}_{as}, \mathbf{0}) - DC_{t+1}(\vec{x}_t, \mathbf{0}) & \text{otherwise} \end{cases} \tag{6.1}$$

where $\vec{f}_{rad}$ is a vector derived from $\vec{f}_t$ containing only those forecast orders that are within a radius of rad miles from the new order. j_s indicates the forecast order that is to be removed while inserting the new order (a, s). j_s^* represents the best forecast order that has been identified for removal, with its area denoted by a^*.

Let $RL_t(\vec{x}_t, \vec{f}_t, \mathbf{1}_{as})$ denote the expected revenue loss of accepting the order (a, s) at time t. Let $w_{s'}(\vec{x}_t, \vec{f}_t)$ denote the idle time in slot s' before accepting the new order (a, s). Suppose the forecast order to be replaced by the new order (a, s) is from area a^*, the idle time after replacing the forecast order by the new order is represented by $w_{s'}(\vec{x}_t + \mathbf{1}_{as}, \vec{f}_t - \mathbf{1}_{a^*})$. According to Abdollahi et al. (2023), the revenue loss for accepting the new order (a, s) is given by:

$$RL_t(\vec{x}_t, \vec{f}_t, \mathbf{1}_{as}) = \sum_{s' \in S} \theta_{t,s'} (w_{s'}(\vec{x}_t, \vec{f}_t) - w_{s'}(\vec{x}_t + \mathbf{1}_{as}, \vec{f}_t - \mathbf{1}_{a^*})) \tag{6.2}$$

where $\theta_{t,s'} \in \mathbb{R}$ represents the monetary value of the consumed time by all orders in time slot s'. To calculate $\theta_{t,s'}$ we use Equation (6.3) which is based on the best route at time t generated by the routing package:

$$\theta_{t,s'} = \frac{\sum\limits_{o \in \{(\vec{x}_t, \vec{f}_t) : U_{s'}^B \leq \vec{E}_b(o,k) \leq U_{s'}^E\}} r(o)}{U_{s'}^E - U_{s'}^B} \tag{6.3}$$

where $U_{s'}^B$ and $U_{s'}^E$ denote the start and end times, respectively, of time slot s'. Additionally, $\vec{E}_b(i,k)$ signifies the balanced earliest arrival time for order i in van k, calculated as the delivery time output by Algorithm 3. This output is determined based on the current solution provided by the dynamic vehicle routing system. The concept of calculating revenue loss was originally introduced by Abdollahi et al. (2023). However, in this study, we extend the approach by considering both actual and forecast orders that remain in the system at time t to calculate $\theta_{t,s'}$. This inclusion allows us to account for the contribution of both actual and forecast orders towards revenue loss until the end of the booking period.

Note that formulas (6.1), (6.2) and (6.3) are valid for both STWs and ATWs. With (6.1), the IC of the ATW is identical to that of the STWs which are combined together, as they are combined when their insertion costs are the same. With (6.2) and (6.3), the RL of the ATW is similar to an average of all combined STWs. ATWs therefore use the same MNL and pricing equations as STWs (Eqs. (6.5)–(6.6)); no ATW-specific pricing rule is introduced. Assembling the insertion cost Equation (6.1) and revenue loss Equation (6.2), we obtain the opportunity cost estimate:

$$OC_t(\vec{x}_t, \vec{f}_t, \mathbf{1}_{as}) = IC_t(\vec{x}_t, \vec{f}_t, \mathbf{1}_{as}) + RL_t(\vec{x}_t, \vec{f}_t, \mathbf{1}_{as}) \tag{6.4}$$

Therefore, the term $(V_{t+1}(\vec{x}_t) - V_{t+1}(\vec{x}_t + \mathbf{1}_{as}))$ in Equation (3.2) will be approximated by the estimated opportunity cost as given in Equation (6.4). To compute the optimal slot price d_s^*, we use the following formula suggested by Dong et al. (2009):

$$d_s^* = OC_t(\vec{x}_t, \vec{f}_t, \mathbf{1}_{as}) - r_t - \frac{h}{\beta_d}. \tag{6.5}$$

where h shows the unique solution to:

$$(h-1)\exp(h) = \sum_{s \in F_a(\vec{x}_t, \vec{f}_t)} \exp\big(\beta_0 + \beta_s + \beta_d\,[\,OC_t(\vec{x}_t, \vec{f}_t, \mathbf{1}_{as}) - r_t\,]\big) \tag{6.6}$$

where h value in Equation (6.6) is calculated using approximation methods like Bisection or Newton.

7 Experimental Results

In the upcoming section, we will begin by outlining the areas of study and the methodologies employed to compare our method with several benchmarks. We will present the results obtained from diverse experiments, evaluated against various measures. Moreover, we will provide a detailed explanation of the implementation aspect pertaining to the distribution of forecast orders. Furthermore, we will explore the potential of augmented time windows to further enhance the effectiveness of our methods. We will illustrate the performance improvements achieved through this approach and compare them to scenarios where standard time windows are used exclusively.

7.1 Data Specification

In order to examine the impact of forecast order distribution and augmented time windows, we conducted experiments on seven distinct delivery areas. These areas were chosen to represent different customer densities and distribution patterns. The experiments were carried out using actual customer locations and historical booking data, while ensuring the protection of commercial information and customer privacy through appropriate measures. Table 1 provides a comprehensive overview of the selected areas, including their specific characteristics and the number of vans deployed to fulfil the final accepted delivery orders.
Customer time-slot choice is modelled using the multinomial logit (MNL) formulation in Eq. (3.1). The coefficient set $(\beta_0, \beta_d, \beta_s)$ is held constant across areas to isolate the effects of routing feasibility and pricing policy, and is reported in Appendix D (Table 6).
We benchmark the proposed approach against several comparison pricing methods, described next. These methods include:

1. Static Pricing method (SP): This method applies a fixed delivery charge of £3 throughout the booking horizon.
2. Dynamic Pricing with Insertion Cost without employing forecast order (DP-IC): In this method, the pricing is dynamically adjusted based on the insertion cost, without considering the forecast orders. This method is a benchmark method which is used in different works such as Yang and Strauss (2017).
3. Dynamic Pricing with Fixed Routing of Forecast order with time window (DP-FR-F): This method involves dynamically adjusting the pricing based on fixed, time-windowed forecast orders obtained from historical route; forecast route is not updated with new order acceptance. The approach introduced by Yang et al. (2016) is referred to as the "foresight policy" in this context.
4. Dynamic Pricing with Dynamic Routing of Forecast orders without time windows (DP-DR-F): It is introduced by Abdollahi et al. (2023) which involves dynamically adjusting pricing by considering a forecast route generated from forecast orders without time windows as well as actual orders. The forecast route is continually updated as new orders are accepted.
5. Dynamic Pricing with Dynamic Routing of Distributed Forecast orders without time window (DP-DR-DF): This method considers dynamically adjusting the pricing based on forecast route plan entailing distributed forecast orders; the forecast route is updated upon new order acceptance. This is the approach proposed in Section 4.
6. Dynamic Pricing with Dynamic Routing of Distributed Forecast orders without time windows entailing Augmented Time Windows (DP-DR-DF-ATW): This approach is same as DP-DR-F, but with the inclusion of augmented time windows as proposed in Section 5.

For a detailed breakdown of the components of each method, please refer to Table 2.

Lastly, the experiments were conducted using MATLAB on a high-performance computing system equipped with an Intel Core i9-7940X 3.1GHz processor. To ensure reliable results, we performed a total of 30 independent runs of the experiments and reported the average.

7.2 Comparative analysis of the methods based on various metrics

To evaluate the effectiveness of the new approaches introduced earlier, namely the employment of distributed forecast order and dynamically extending feasible time windows through

Table 1 Features of seven areas under study

Area	1	2	3	4	5	6	7
Geographical spread	Rural	Semi-Rural	Suburb	Suburb	Suburb	City	City
Average number of customers	63	130	206	308	398	502	611
Average customer density(mile)	11.78	5.87	13.43	11.98	8.81	2.67	2.32
Number of delivery vehicles	2	4	6	9	11	14	16
Type of vehicle	Van	Van	Van	Van	Van	Van	Van
Traffic condition	Low	Low	Medium	Medium	Medium	High	High

Table 2 Comparison of different methods based on key features

Method	Pricing policy	OC	Route planning	Forecast order type	Time window type
SP	Static	-	-	-	STW
DP-IC	Dynamic	IC	-	-	STW
DP-FR-F	Dynamic	IC	Fixed forecast orders	With TW	STW
DP-DR-F	Dynamic	IC + RL	Dynamic forecast orders	Without TW	STW
DP-DR-DF	Dynamic	IC + RL	Distributed dynamic forecast orders	Without TW	STW
DP-DR-DF-ATW	Dynamic	IC + RL	Distributed dynamic forecast orders	Without TW	STW & ATW

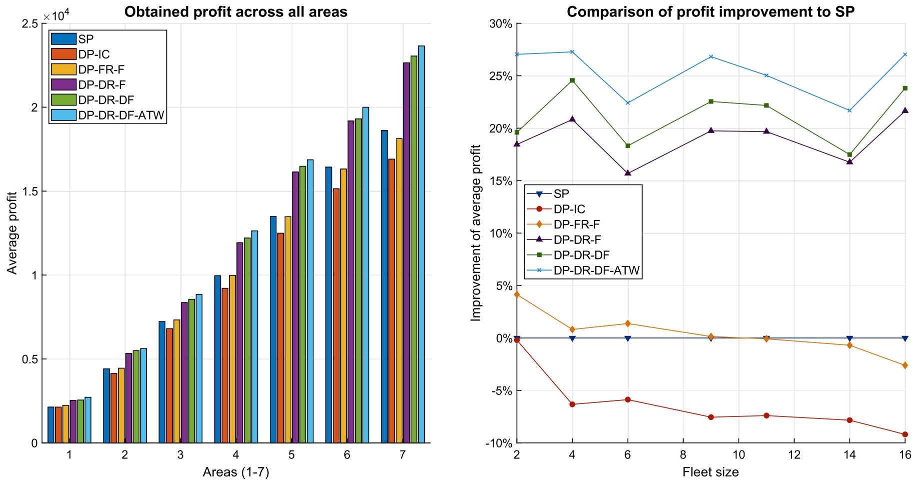

Fig. 5 Comparison of average profit across the studied methods over 30 runs for the seven study areas. Areas 1–7 correspond to instances with 2, 4, 6, 9, 11, 14, and 16 vehicles, respectively. Percentage labels report improvement relative to the static-pricing baseline

the implementation of augmented time windows, various measures have been considered. Among these metrics, profit growth stands out as a crucial benchmark to assess the performance of the methods. In Figure 5, we present the profit achieved for all the methods across different areas, along with the percentage improvement in profit compared to the static pricing method used as the baseline. This visualisation provides a comprehensive overview of how each method performs in terms of generating profit and highlights the extent to which they outperform the static pricing approach.

Analysing Figure 5, it is evident that DP-DR-DF outperforms DP-DR-F, highlighting the significance of distributing forecast orders. This observation emphasises the importance of accurately estimating revenue loss, as it enables a more realistic calculation of opportunity costs. By considering and accurately estimating revenue loss, businesses can make more informed decisions and achieve better financial outcomes by optimising their pricing policies. For a more detailed investigation into the distribution of forecast orders and the estimation of time budget as a component of revenue loss calculation, please refer to Section 7.3.

The DP-DR-DF-ATW model, enhanced by integrating augmented time windows, showcases a profit boost of 22% to 27% over static pricing. This uplift highlights augmented time windows' role in steering customers to more profitable time slots across the booking period. Customers are offered broader time window choices and lower delivery charges, enabling more efficient order management and countering the preference for popular time slots. This approach ensures higher slot availability, elevating booking success rates. In Figure 5, the assumed equal utility of combined slots may not always hold; Section 7.4 delves into utility decrement variations and their effects on DP-DR-DF-ATW's performance.

Another advantage of the methodology proposed in Section 5 for augmented time windows (ATW) is its compatibility with various methods and approaches in dynamic pricing. This approach can seamlessly function as an add-on to any insertion-based dynamic routing system, facilitating profit improvement without requiring modifications to the specific method for dynamic pricing. This flexibility allows for easy integration into different methods with diverse approaches. Furthermore, the inclusion of ATW does not impose significant

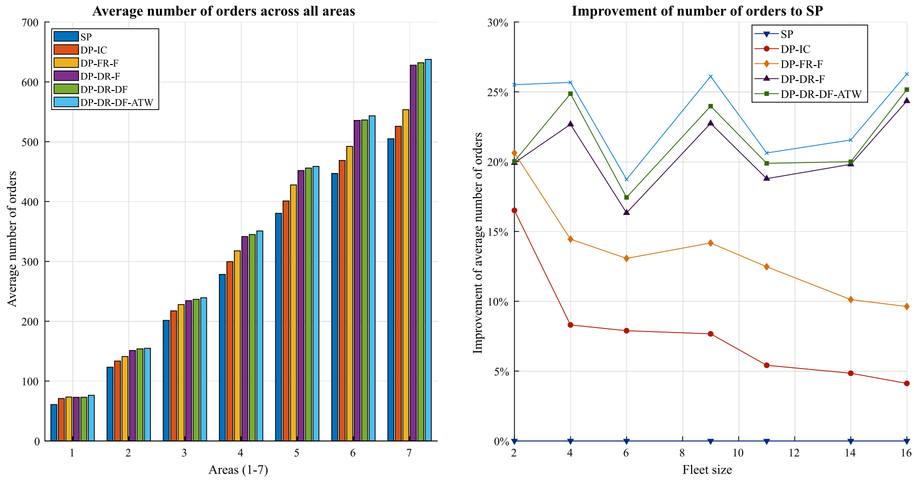

Fig. 6 Comparison of the average number of accepted orders across the studied methods over 30 runs for the seven study areas. Areas 1–7 correspond to instances with 2, 4, 6, 9, 11, 14, and 16 vehicles, respectively

computational burdens on the methods. Instead, it simply redefines the standard time window based on the inherent characteristics of each method. Consequently, the integration of augmented time windows contributes to improved performance without introducing excessive computational complexities.

Figures 6 and 7 highlight the improved efficiency the method brings to the dynamic routing system. Figure 6 showcases the average number of accepted orders across all areas. This figure highlights the effectiveness of our approaches in accommodating a greater number of orders within the same problem setting. By optimising the routing process, the DP-DR-DF-ATW excels in incorporating more orders into the routes, thereby maximising resource employment and meeting customer demand effectively. In addition, the DP-DR-DF approach achieves a slight increase in accepted orders while generating more profit through enhanced pricing of time slots and more accurate RL estimation. Complete dispersion statistics (mean $\pm$ sd and [min, max]) and percentage improvements versus the SP baseline for all methods and areas are provided in Appendix B.

In Figure 7, a graphical comparison is presented, focusing on the average travelling cost required to serve an order and the average profit gain per mile in all areas. These metrics directly reflect the efficiency of the final routes generated by the methods and their impact on the overall performance of the dynamic routing system. The figure underscores the effectiveness of our approaches in achieving cost-minimisation objectives while maximising profit gains per mile. This emphasises the importance of an efficient forecast route plan, enabling intelligent and strategic travel decisions, ultimately contributing to enhanced profitability. It is important to note that the x–axis in Figure 7 does not represent increasing the fleet size within a single fixed region, but rather a sequence of seven distinct areas, each with its own fleet size, demand volume and spatial structure (see Table 1 and Appendix C). The local increase in travelling cost per order around fleet sizes 9 and 11 (Areas 4 and 5) arises when moving to larger, more geographically dispersed suburban regions in which the total number of customers and the covered area grow faster than the fleet size. In these settings, routes must still cover substantial line-haul distance to connect multiple customer clusters, which increases the miles travelled per order. By contrast, the jump in profit per mile at fleet sizes

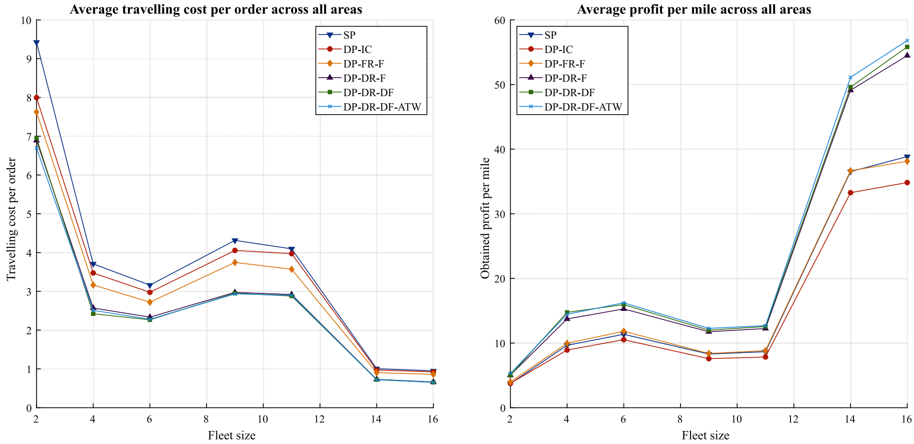

Fig. 7 Comparison of average travelling cost per accepted order and average profit per mile over 30 runs for the seven study areas. Areas 1–7 correspond to instances with 2, 4, 6, 9, 11, 14, and 16 vehicles, respectively

14 and 16 (Areas 6 and 7) is associated with very dense urban layouts, where many closely located customers allow each mile of travel to generate more revenue and the fixed and time-related vehicle costs are spread over a larger number of delivered orders. These structural differences across areas, rather than fleet size alone, therefore explain the non-monotonic patterns observed in Figure 7.

The availability of time slots and their corresponding prices over the booking period significantly impact the total profit. Figure 8 provides insights into the number of available time slots and the average advised price for representative areas (1 and 7). When a new order arrives, there exists a trade-off between immediate gain and long-term profit. This trade-off explains why the prices offered by DP-IC and DP-FR-F approaches are consistently lower than those of DP-DR-F, DP-DR-DF, and DP-DR-DF-ATW. Note that the DP-IC and DP-FR-F approaches, lacking consideration for expected revenue loss in the opportunity-cost estimation, primarily focus on the immediate gain associated with the order under consideration. To entice the customer to make a purchase, these approaches lower the price offered significantly. However, in the case of DP-DR-F, DP-DR-DF, and DP-DR-DF-ATW approaches, which anticipate future orders, there is less pressure to secure an immediate order. Consequently, the prices offered tend to be higher on average. The variation in prices across different approaches underscores the significance of incorporating an accurate assessment of revenue loss in the opportunity-cost estimation. This consideration enables approaches to strike a balance between immediate gains and long-term profitability, resulting in more strategic pricing decisions and ultimately maximising the total profit.

When comparing DP-DR-DF and DP-DR-DF-ATW with DP-DR-F, it is evident that the prices charged by DP-DR-F are generally lower. This is because in DP-DR-F, the best time slot is determined solely based on the dynamic routing system without considering time windows or the original popularity of time slots. Consequently, there is a higher pressure to lower the price in order to persuade a customer to book an undesirable time slot if their location is deemed the best fit for that particular slot. On the other hand, DP-DR-DF and DP-DR-DF-ATW offer more stable delivery prices compared to DP-DR-F by leveraging the benefits of forecast order distribution. Ideally, the pricing policy towards the end of the booking horizon should aim to treat customers fairly by providing consistent prices across

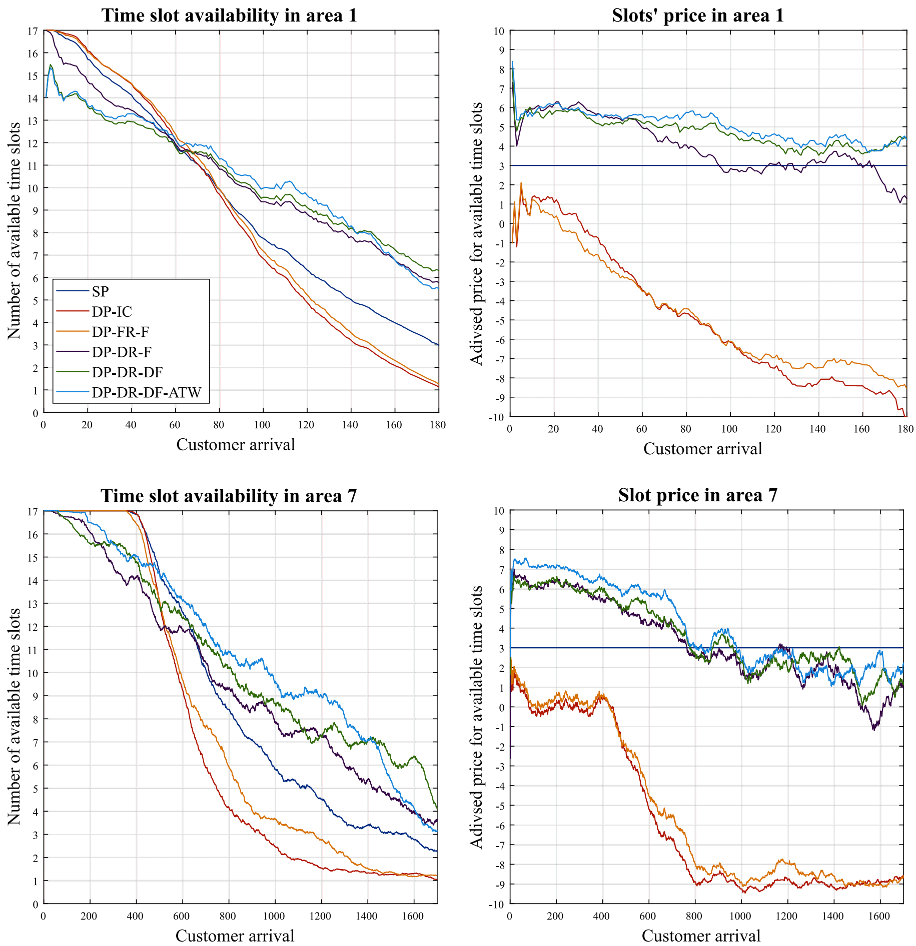

Fig. 8 Evolution of average slot availability and average offered delivery price during the booking horizon for two representative instances: Area 1 (2 vehicles) and Area 7 (16 vehicles), averaged over 30 runs

all time slots. This equitable approach is particularly evident in low-density areas with lower complexity, where the demand system is less intricate. However, in areas with higher demand, this advantage may be somewhat diminished due to factors such as increased order volume, more delivery vans, and greater pressure on the dynamic routing system. Nevertheless, in real-world scenarios, it is common for prices in dynamic pricing schemes to decrease over time as slot availability dwindles. Typically, popular slots tend to be reserved earlier than less popular ones, resulting in lower prices being offered towards the end of the booking period to stimulate bookings for the less popular slots.

Figure 9 presents a run-time assessment of all methods across all seven areas. It quantifies the time required by each method to evaluate and display feasible time slots and their corresponding prices for a new order.

As shown in Fig. 9, the maximum observed per-arrival processing time is 0.014 s across all areas–even under fine time discretisation. Note that the algorithm is event-driven: computation is triggered only on arrivals (not on every epoch T). The computational complexity,

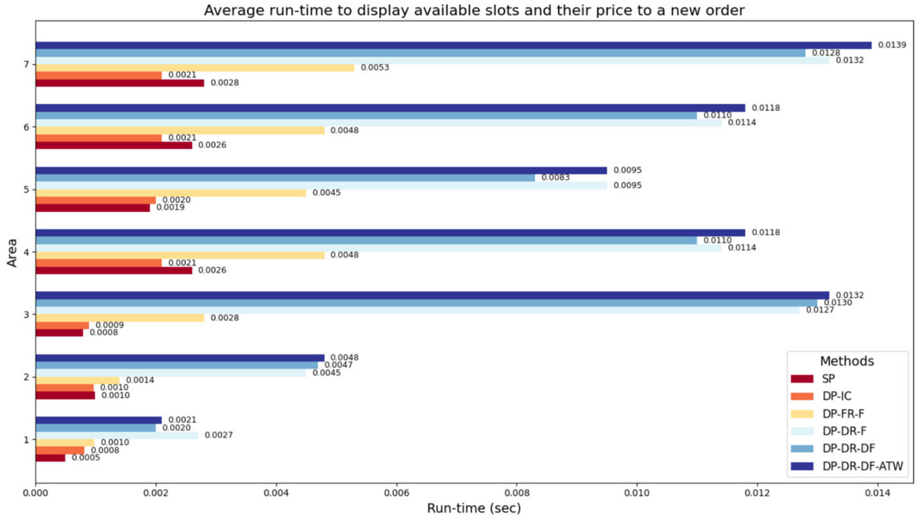

Fig. 9 Per-arrival run-time required to evaluate feasible slots and compute the corresponding delivery prices across the seven study areas. Areas 1–7 correspond to instances with 2, 4, 6, 9, 11, 14, and 16 vehicles, respectively

therefore, scales with the number of arrivals rather than the number of discretised epochs; in our test set, the average number of customers increases from 63 in Area 1 to 611 in Area 7 (Table 1). All methods are therefore suitable for online use; moreover, DP–DR–DF is marginally faster than DP–DR–F in many areas, because distributing forecast orders smooths the route plan and shrinks the set of neighbouring forecast orders that must be considered for replacement when a new order arrives.

7.3 The Impact of Forecast Order Distribution on Time Budget

In this section, we aim to demonstrate the impact of distributing forecast orders based on the CAP. Figure 10 serves as a visual representation, highlighting how the distribution of forecast orders contributes to aligning the time budget of each slot with the slot popularity predicted by the MNL model. This comparison provides valuable insights into how effectively the CAP-based distribution approach can optimise time slot employment and improve the overall match between customer demand and available time slots.

The left heatmap corresponds to the DP-DR-F method, where forecast order distribution is not considered. It reveals that the time budgets for the first half of the time slots exhibit higher values across all customer arrivals. In contrast, the right heatmap representing the DP-DR-DF method, which integrates forecast order distribution, demonstrates two distinct peaks in the time slot values. This pattern aligns with the slot popularity observed in the Multinomial Logit (MNL) model. Relative to DP-DR-F, DP-DR-DF shifts time budget away from early slots (e.g., 1–8) and restores later slots (e.g., 9–17), consistent with per-slot caps moderating early-slot concentration (Figure 10).

The line graph positioned below each heatmap illustrates the average time budget across all customer arrivals, juxtaposed with the scaled slot popularity. Notably, the incorporation of forecast order distribution in the DP-DR-DF method leads to a more pronounced alignment

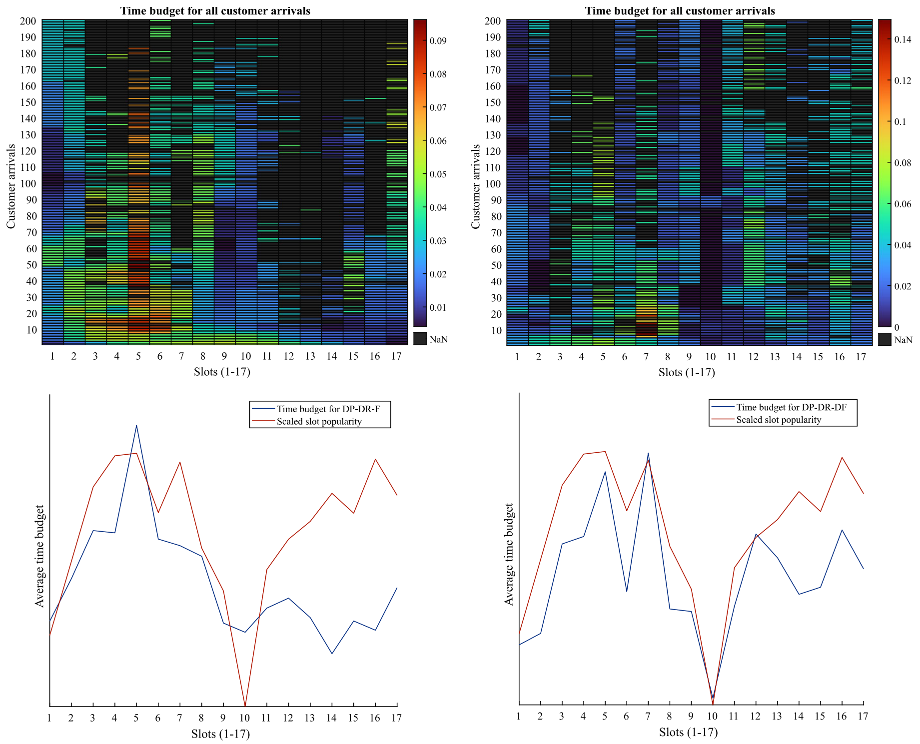

Fig. 10 Comparison of estimated slot time budgets and scaled MNL slot popularity during the booking horizon for Area 1. The left panel corresponds to DP-DR-F and the right panel to DP-DR-DF; the heatmaps show slot-wise time budget over arrivals, and the line plots show the corresponding average profiles

between the slot popularity and the time budget, highlighting the impact of this approach on achieving a closer match between predicted popularity and actual time allocations.

7.4 Incentives on the ATW by calculation of new MNL parameters corresponding to for ATW

This section elaborates on the modification of the MNL choice model, incorporating ATW parameters alongside the known STW parameters. As outlined in Section 3, Equation (3.1) dictates the probabilities of selecting and not selecting a time slot s, respectively, based on the delivery charge vector $\vec{d}_{a,t}$. The β_s parameter, indicative of slot utility, is pivotal in these equations.

Since this is the first implementation of augmented time windows, no historical data is available to estimate β_s values for ATW. Thus, we need to adapt the β_s parameter, initially defined for STW, to suit ATW, using Equation (7.1). This equation averages the β_s values of constituent slots forming s' and reduces it by a factor correlating with s' length.

For an augmented time window s_i', comprising standard time windows $\{stw_1, stw_2, \ldots, stw_{n_i}\}$, the β value is calculated accordingly.

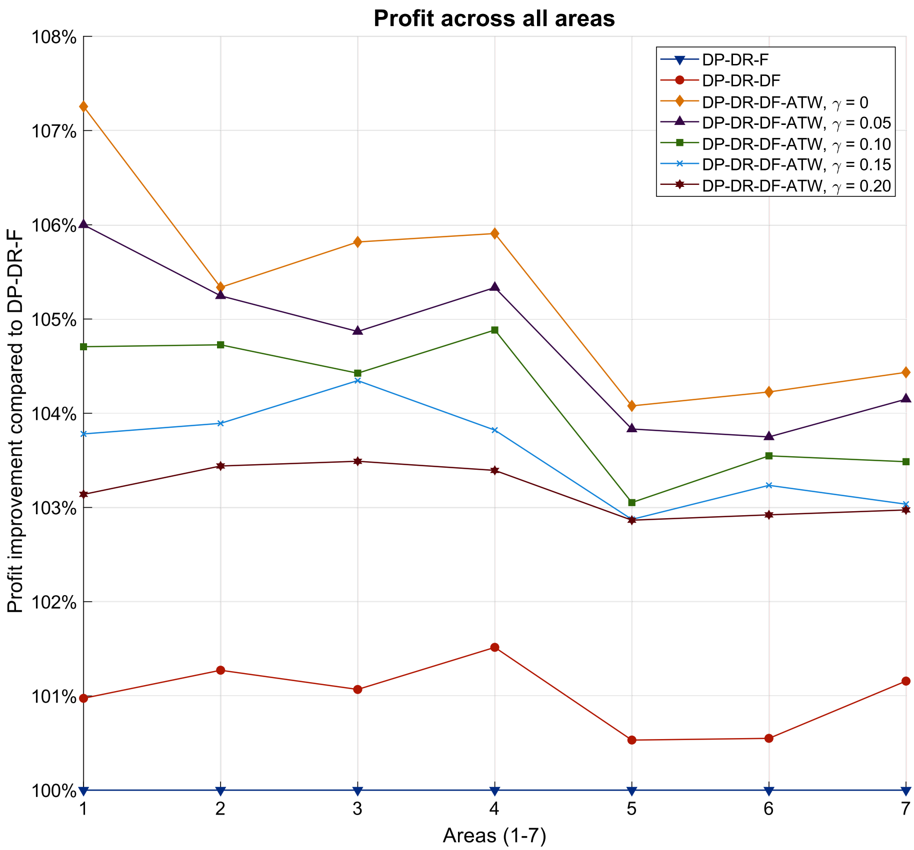

Fig. 11 Impact of γ values on profit in all areas

$$\beta_{s_i'} = \frac{\sum_{j=1}^{n_i} \beta_{stw_j}}{n_i} - ((n_i - 1) \times \gamma) \tag{7.1}$$

Since an augmented time window has a broader time range and requires customers to wait longer at the delivery location compared to a standard time window, its popularity among customers is expected to decline. Equation (7.1) is constructed based on the assumption that the utility of an augmented slot decreases linearly with its length, and the rate of reduction is captured by the parameter γ. During experiments, we test different values of γ to examine the influence of combining time slots.

To determine suitable scales for γ values, we can interpret γ in terms of monetary units. Recall that β_d represents the utility parameter associated with delivery price. Consequently, as the price per unit increases, the utility will decrease by an amount proportional to $|\beta_d|$ (where β_d is a negative value). For example, if we assume that extending a time slot by one hour leads to a utility reduction equivalent to increasing its price by £Z, then γ can be set as $\gamma = Z|\beta_d|$ (equivalently, $\gamma = -Z\beta_d$ since $\beta_d < 0$).

In our research, we consider a range of γ values spanning from 0 to 0.2, approximately corresponding to a price increment of £0 to £2.6. The summary of profit gains is visualised in Figure 11, which highlights the performance of the DP-DR-DF-ATW method across various γ bands in comparison to other approaches.

Analysing Figure 11, we observe that areas with lower customer density exhibit higher profit gains when using the DP-DR-DF-ATW method; areas with a higher volume of customer requests show a lower profit gain. Nevertheless, the improvement in profit remains substantial. This indicates the efficacy of applying ATW in exploring and exploiting customers' flexibility in time slots. A more comprehensive display of the impact of γ values is presented in Table 3, which focuses on representative areas 1, 3, and 6.

In high-volume (dense) areas, larger fleets and tighter overlaps between delivery windows leave limited routing slack, meaning DP–DR–DF typically operates close to the feasibility boundary. Augmented time windows (ATWs) therefore provide fewer opportunities to unlock additional feasible rearrangements of stops, resulting in a smaller incremental profit than in lower-volume (sparser) areas, where latent slack can be converted into additional accepted orders and improved route efficiency. This mechanism is reflected in Fig. 11: as γ increases (customers value ATWs less), both acceptance and profit decrease across all areas, while DP–DR–DF–ATW consistently remains above DP–DR–DF. Per-area levels and dispersion are reported in Appendix B (Table 4).

Table 3 provides a comparison of the performance metrics between DP-DR-DF and DP-DR-DF-ATW variants using different γ values. As explained before, higher γ values mean lower preference of ATWs and therefore lower selection rates of these slots. As a result, this leads to increased mileage per order and a decrease in the number of orders, which justifies ATWs' ability of maintaining routing flexibility and conveying more orders. As γ increases, the total profit decreases. However, even with the largest γ value, the DP-DR-DF-ATW variants outperform DP-DR-DF. This is particularly evident when considering the profit improvement, which shows that even at the largest γ, DP-DR-DF-ATW still outperforms DP-DR-DF.

In our final analysis, we examine the selection rate comparison between ATW and STW. Figure 12 clearly demonstrates that a considerable proportion of customers, ranging from 8.5% to 13.2%, prioritise ATW over STW. This finding strongly suggests that a notable segment of customers offer their flexibility and potential cost savings offered by choosing from ATW, which typically have a lower delivery charge, as opposed to selecting specific predefined time slots with a higher delivery charge associated with STW.

7.5 Impact of forecast accuracy on DP-DR-DF-ATW efficacy

In Figure 13, we explore the impact of varying forecast levels on the efficacy of DP-DR-DF-ATW. This figure demonstrates how underestimating or overestimating order numbers influences the performance of our proposed method. By intentionally varying the forecast levels to be 20% lower or higher than the expected average, we seek to simulate real-world scenarios where prediction inaccuracies are inevitable.

The DP-DR-DF-ATW method, characterised by its distinct brown bars, showcases notable resilience against forecast inaccuracies. Under scenarios of both underestimation and over-estimation, DP-DR-DF-ATW consistently outperforms, maintaining a significant advantage over other evaluated methods, such as SP, DP-IC, DP-FR-F and DP-DR-F. This evidence of resilience underscores DP-DR-DF-ATW's superior adaptability and robustness in handling forecast inaccuracies, highlighting its effectiveness across various scenarios.

Table 3 Sensitivity analysis on γ for DP-DR-Df-ATW with comparison to DP-DR-DF

	γ **values**	Total mileage	Mileage/Order	Number of orders	Total order size	Accepted price	Total profit	profit improvement(%)
Area 1	0	510.74	6.62	76.27	284.63	1.80	2712.81	6.22
	0.05	517.50	6.62	75.7	285.79	1.67	2681.05	4.98
	0.10	507.21	6.70	75.1	282.067	1.51	2648.33	3.70
	0.15	516.42	6.88	74.26	276.80	1.37	2624.93	2.78
	0.20	513.90	6.86	74.03	276.43	1.39	2608.71	2.74
	DP-DR-DF	507.32	6.89	72.93	274.46	0.95	2553.96	
	γ **values**	Total mileage	Mileage/Order	Number of orders	Total order size	Accepted price	Total profit	Profit improvement(%)
Area 3	0	545.32	2.23	239.20	914.05	2.39	8847.95	4.70
	0.05	553.63	2.33	237.20	909.17	2.26	8768.50	3.76
	0.10	549.51	2.30	237.93	907.25	2.18	8731.50	3.32
	0.15	552.19	2.33	236.13	906.33	2.20	8724.89	3.24
	0.20	540.03	2.28	239.39	900.57	2.12	8653.31	2.40
	DP-DR-DF	537.05	2.28	236.60	899.25	1.32	8450.85	
	γ **values**	Total mileage	Mileage/Order	Number of orders	Total order size	Accepted price	Total profit	Profit improvement(%)
Area 6	0	391.22	0.72	543.30	2141.70	1.11	19996.41	3.66
	0.05	391.92	0.72	539.63	2135.04	1.06	19904.82	3.18
	0.10	390.73	0.72	540.67	2140.20	0.90	19866.18	2.98
	0.15	391.18	0.73	538.23	2134.21	0.89	19806.22	2.67
	0.20	392.70	0.73	538.73	2139.41	0.69	19746.09	2.36
	DP-DR-DF	392.58	0.73	535.37	2122.49	0.13	19291.04	

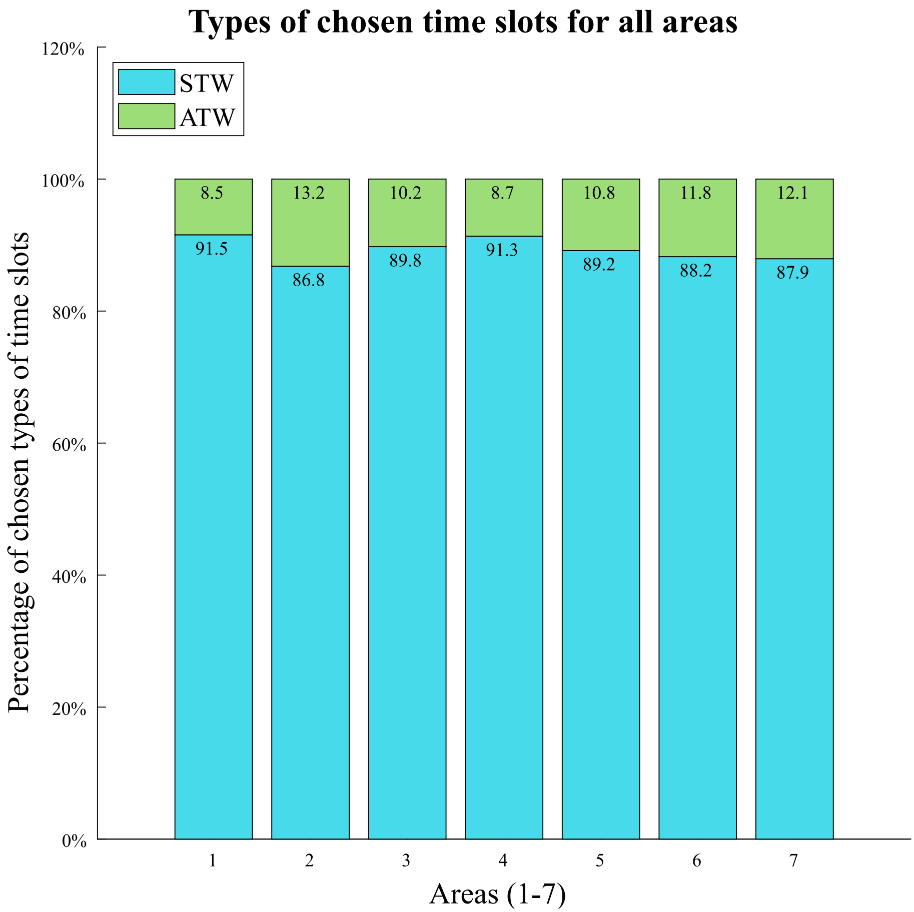

Fig. 12 Percentage distribution: selection of STW versus ATW in all areas

8 Discussion and Conclusion

This section synthesises our methodological contributions and empirical results, and draws out implications for practice and research. We first recap the setting and the two proposed enhancements–slot-wise feasibility caps ($\overrightarrow{CAP}$) and Augmented Time Windows (ATW)–and how they integrate with pricing. We then summarise the main empirical findings (acceptance and prices, time-budget rebalancing, and online runtime) and close with limitations and directions for future work.

Setting and contributions

We study integrated demand management and dynamic vehicle routing for attended home delivery with price-sensitive slot choice. Two practical enhancements are proposed. First, we regularise forecast-route construction using per-slot feasibility upper bounds ($\overrightarrow{CAP}$), computed under the most slot-favourable admissible prices and verified for routing feasibility;

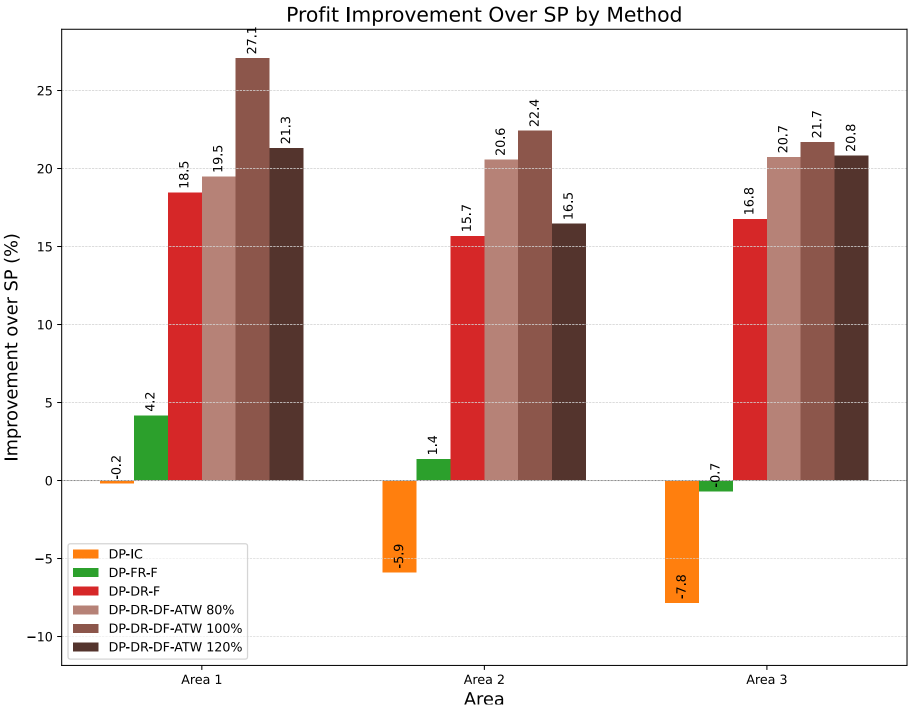

Fig. 13 The effect of inaccuracy of forecast level on the performance of DP-DR-DF-ATW

$\overrightarrow{CAP}$ acts only as a hard/soft cap in CVRPTW–CAP/SA and does not directly enter the pricing equation. Second, we introduce Augmented Time Windows (ATW) formed by merging contiguous STWs that share the same insertion context, thereby preserving routing flexibility when forecast orders are replaced by actual orders. ATWs enter the same MNL/pricing as STWs.

Empirical findings

(i) *Acceptance and prices.* Relative to the no-cap baseline (DP–DR–F), introducing $\overrightarrow{CAP}$ (DP–DR–DF) and augmented time windows (DP–DR–DF–ATW) increases order acceptance, while the mean accepted delivery fee also rises (Figs. 6 and 8). This pattern is consistent with reduced discount pressure on less-preferred early slots once feasibility is regularised. Importantly, delivery fees remain determined by the opportunity-cost structure $OC = IC + RL$; $\overrightarrow{CAP}$ acts only as a feasibility regulariser that mitigates early-slot crowding rather than as a direct pricing driver.

(ii) *Time-budget rebalancing.* $\overrightarrow{CAP}$ attenuates over-concentration in early slots and preserves time budgets in later slots, yielding allocations that more closely track MNL-implied slot popularity (Fig. 10). Augmented time windows can be deployed selectively to exploit routing slack when it exists. Consistent with Fig. 11 and Appendix B, the incremental benefit of ATWs is larger in sparser regions (where slack is available) and smaller in denser regions operating near capacity.

(iii) *Online runtime.* Warm-started simulated annealing with a time-window-feasible neighbourhood delivers per-arrival decision times below 0.014 s across areas (Fig. 9), while maintaining alignment between $OC = IC + RL$ and the current feasible plan. If required operationally, non-negative delivery fees can be imposed by setting $d_s^{\min} \geq 0$ without altering the pricing logic. Reporting both mean outcomes and dispersion remains important for assessing policy stability and robustness.

Limitations and future work

The bound $\overrightarrow{CAP}$ is intentionally policy-agnostic. While tighter, slot-specific forecasts could be simulated, doing so would couple capacity control to a particular policy and may risk reinforcing that policy during search. Future work includes retailer-specific or adaptive bounds, richer choice models, alternative constructions of $\overrightarrow{CAP}$, integration with additional fast CVRPTW heuristics, and extending ATWs to non-adjacent slot combinations jointly with incentive design.

Appendix A Formal statement of CVRPTW with order caps

We will now provide a formal formulation of the mathematical model that is the focus of our study. As the formal mathematical model is not computationally tractable, it subsequently will be addressed through an approximate solution using a Simulated Annealing (SA) algorithm. Within the booking horizon, we define $\mathcal{A}_t$ as the set of actual orders and $\mathcal{F}_t$ as the set of forecast orders at time step t. The cost of travel from order i to j is represented by C_{ij}. The binary variable X_{ijk} takes a value of 1 when there exists a direct connection between orders i and j using van k on the proposed delivery route plan. l_{ij} denotes the distance between orders i and j. Each van in the problem has a specified capacity denoted by Q, and each order i has a specific demand represented by q_i. Additionally, Φ_{ik} denotes the calculated time at which van k either begins service at order location i or departs from/arrives at the depot (for $i = 0$ and $i = D$ respectively). The problem setting provides a list of pre-defined time windows, each of which has a designated start time denoted by U_s^B and an end time denoted by U_s^E. $[TW_i^B, TW_i^E]$ signifies the agreed-upon time window of order i; for forecast orders, this constraint is relaxed. We refer to this modified problem as CVRPTW-CAP, and its formulation is as follows.

Constraints (A.2) and (A.6) enforce the requirement that each order is visited only once and that the van's capacity is not exceeded. The van departs from the depot at location 0, visits the orders, and returns to the depot at location D, as indicated by constraints (A.3)-(A.5). Inequality (A.7) is designed to calculate the arrival times of orders, ensuring that if van k is travelling from location i to location j, it cannot arrive at j before a certain time. This time is calculated as $\Phi_{ik} + b_i + \frac{l_{ij}}{v} - \mathcal{M}(1 - X_{ijk})$. Here, $\mathcal{M}$ represents a large number. The term b_i denotes the service time required at the order location i, and v is the travel speed of the delivery vehicle. Constraint (A.7) also eliminates possible sub-tours. The binary variables α_{is} and δ_{is} ensure order allocation within a time slot's start and end limits, respectively. ξ_{is} acts as an overall indicator confirming correct order placement within these time boundaries. Constraint (A.8) guarantees that time windows are respected. Furthermore, constraints (A.9), (A.10), and (A.11) verify whether order i is serviced within time slot s. Constraint (A.12) counts the number of orders in slot s and ensures that it does not exceed the CAP_s.

Based on the provided formulation, it becomes apparent that this does not strictly fit the definition of a CVRPTW problem. This is due to the fact that the study introduces constraints solely on the maximum order count within each time slot, without directly assigning time slots to specific forecast orders. Since forecast orders lack specific time windows, they have the flexibility to be placed in any available slot based on their respective locations.

$$\min \quad \sum_{i\in\mathcal{A}_t\cup\mathcal{F}_t\cup\{0\}} \sum_{j\in\mathcal{A}_t\cup\mathcal{F}_t\cup\{D\}} \sum_{k} C_{ij}X_{ijk} \tag{A.1}$$

$$s.t. \quad \sum_{j\in\mathcal{A}_t,\mathcal{F}_t,\{D\}} \sum_{k\in V} X_{ijk} = 1, \qquad \forall i \in \mathcal{A}_t \cup \mathcal{F}_t \tag{A.2}$$

$$\sum_{j\in\mathcal{A}_t\cup\mathcal{F}_t\cup\{D\}} X_{0jk} = 1, \qquad \forall k \in V \tag{A.3}$$

$$\sum_{i\in\mathcal{A}_t\cup\mathcal{F}_t\cup\{0\}} X_{iDk} = 1, \qquad \forall k \in V \tag{A.4}$$

$$\sum_{i\in\mathcal{A}_t\cup\mathcal{F}_t\cup\{0\}} X_{igk} = \sum_{j\in\mathcal{A}_t\cup\mathcal{F}_t\cup\{D\}} X_{gjk}, \qquad \forall g \in \mathcal{A}_t \cup \mathcal{F}_t, \forall k \in V \tag{A.5}$$

$$\sum_{i\in\mathcal{A}_t\cup\mathcal{F}_t} q_i \sum_{j\in\mathcal{A}_t\cup\mathcal{F}_t\cup\{D\}} X_{ijk} \leq Q, \qquad \forall k \in V \tag{A.6}$$

$$\Phi_{ik} + b_i + \frac{l_{ij}}{v} - \mathcal{M}(1 - X_{ijk}) \leq \Phi_{jk}, \qquad \forall i \in \mathcal{A}_t \cup \mathcal{F}_t \cup \{0\}, \forall j \in \mathcal{A}_t \cup \mathcal{F}_t \cup \{D\}, \forall k \in V \tag{A.7}$$

$$TW_i^B \leq \Phi_{ik} \leq TW_i^E, \qquad \forall i \in \mathcal{A}_t, \forall k \in V \tag{A.8}$$

$$U_s^B \geq \Phi_{ik} - \mathcal{M}\delta_{is}, \qquad \forall i \in \mathcal{A}_t \cup \mathcal{F}_t, \forall k \in V, \forall s \in S \tag{A.9}$$

$$U_s^E \leq \Phi_{ik} + \mathcal{M}\alpha_{is}, \qquad \forall i \in \mathcal{A}_t \cup \mathcal{F}_t, \forall k \in V, \forall s \in S \tag{A.10}$$

$$\xi_{is} \geq \alpha_{is} + \delta_{is} - 1, \qquad \forall i \in \mathcal{A}_t \cup \mathcal{F}_t, \forall s \in S \tag{A.11}$$

$$\sum_{i\in\mathcal{A}_t\cup\mathcal{F}_t} \xi_{is} \leq CAP_s, \qquad \forall s \in S \tag{A.12}$$

$$X_{ijk} \in \{0, 1\}, \qquad \forall i \in \mathcal{A}_t \cup \mathcal{F}_t \cup \{0\}, \forall j \in \mathcal{A}_t \cup \mathcal{F}_t \cup \{D\}, \forall k \in V \tag{A.13}$$

$$\alpha_{is} \in \{0, 1\}, \qquad \forall i \in \mathcal{A}_t \cup \mathcal{F}_t, \forall s \in S \tag{A.14}$$

$$\delta_{is} \in \{0, 1\}, \qquad \forall i \in \mathcal{A}_t \cup \mathcal{F}_t, \forall s \in S \tag{A.15}$$

$$\xi_{is} \in \{0, 1\}, \qquad \forall i \in \mathcal{A}_t \cup \mathcal{F}_t, \forall s \in S \tag{A.16}$$

$$\Phi_{ik} \geq 0, \qquad \forall i \in \mathcal{A}_t \cup \mathcal{F}_t \cup \{0\} \cup \{D\}, \forall k \in V. \tag{A.17}$$

It is evident that the proposed CVRPTW-CAP is NP-hard, however it has to be solved iteratively, upon every update of sets $\mathcal{A}_t$ and $\mathcal{F}_t$ with the acceptance of new orders. As a consequence of its computational complexity, heuristic methods will be employed to find a near-optimal solution for practical purposes.

Appendix B Summary statistics over 30 runs (Areas 1–7)

Each entry shows mean ± sd [min, max] across n=30 independent runs. The last column reports percentage improvement in Total profit relative to the SP baseline for that area.

Appendix C Spatial distribution of the test areas

To help interpret the area-wise performance results reported in Section 7, Figure 14 illustrates the spatial distribution of forecast orders and depot locations for the seven areas summarised in Table 1. Each panel shows the customer locations in local coordinates, expressed as Miles (East) and Miles (North) relative to the corresponding depot and scaled into a common $[0, 100] \times [0, 100]$ frame. This normalisation enables a direct visual comparison of the relative spread and density of demand across areas.

The figure highlights the diversity of geographical patterns considered in our experiments: Area 1 is a sparse rural region with a small number of customers spread over a wide area; Areas 2–5 correspond to increasingly larger and more populated semi-rural and suburban regions; and Areas 6–7 are dense urban areas where many customers are concentrated within a relatively small spatial extent. These differences in spatial structure, together with the varying number of orders and vehicles reported in Table 1, explain why several performance curves in the main text are non-monotonic with respect to "fleet size".

Table 4 Appendix B – Summary statistics over 30 runs (Areas 1–7): mean±sd [min, max] and improvement vs. SP baseline

Area	Method	Total profit	Mileage / order	Accepted orders	% vs. SP (profit)
1	SP	2030.73 ± 28.4 [1980, 2080]	9.46 ± 0.18 [9.20, 9.98]	65.5 ± 1.3 [63, 68]	0.00%
1	DP–IC	2027.07 ± 28.4 [1975, 2075]	8.02 ± 0.14 [7.82, 8.48]	76.3 ± 1.5 [74, 79]	−0.18%
1	DP–FR	2126.99 ± 29.8 [2075, 2175]	7.75 ± 0.13 [7.53, 8.02]	79.1 ± 1.6 [76, 82]	+4.74%
1	DP–DR–F	2406.97 ± 33.7 [2345, 2465]	6.93 ± 0.12 [6.65, 7.10]	78.7 ± 1.6 [76, 82]	+18.53%
1	DP–DR–DF	2434.17 ± 34.1 [2375, 2495]	6.98 ± 0.11 [6.77, 7.18]	78.5 ± 1.6 [76, 81]	+19.87%
1	DP–DR–DF–ATW	2584.94 ± 36.2 [2520, 2650]	6.85 ± 0.10 [6.66, 7.02]	82.3 ± 1.6 [79, 85]	+27.29%
2	SP	4412.49 ± 66.0 [4310, 4520]	3.80 ± 0.09 [3.42, 3.78]	123.2 ± 2.9 [118, 128]	0.00%
2	DP–IC	4131.39 ± 61.8 [4025, 4240]	3.51 ± 0.04 [3.41, 3.73]	134.1 ± 3.2 [128, 140]	−6.37%
2	DP–FR	4431.91 ± 66.3 [4315, 4550]	3.15 ± 0.08 [3.00, 3.31]	141.3 ± 3.3 [135, 147]	+0.44%
2	DP–DR–F	5311.92 ± 79.4 [5170, 5450]	2.57 ± 0.04 [2.43, 2.70]	151.9 ± 3.6 [145, 158]	+20.38%
2	DP–DR–DF	5512.69 ± 82.5 [5370, 5660]	2.51 ± 0.07 [2.37, 2.64]	153.9 ± 3.6 [147, 160]	+24.93%
2	DP–DR–DF–ATW	5623.46 ± 84.2 [5480, 5770]	2.44 ± 0.03 [2.31, 2.57]	154.6 ± 3.6 [148, 161]	+27.44%
3	SP	7031.88 ± 100.0 [6850, 7200]	3.18 ± 0.08 [3.02, 3.34]	201.3 ± 3.7 [195, 208]	0.00%
3	DP–IC	6653.56 ± 94.6 [6490, 6820]	2.91 ± 0.06 [2.78, 3.02]	217.8 ± 4.0 [211, 225]	−5.38%
3	DP–FR	7166.19 ± 101.9 [6990, 7340]	2.35 ± 0.02 [2.22, 2.46]	228.4 ± 4.2 [221, 236]	+1.91%
3	DP–DR–F	8100.73 ± 115.2 [7900, 8300]	2.32 ± 0.05 [2.21, 2.42]	234.1 ± 4.3 [226, 242]	+15.20%
3	DP–DR–DF	8391.14 ± 119.3 [8180, 8600]	2.31 ± 0.05 [2.21, 2.41]	235.9 ± 4.3 [228, 244]	+19.33%
3	DP–DR–DF–ATW	8656.95 ± 123.1 [8440, 8870]	2.31 ± 0.04 [2.20, 2.41]	237.7 ± 4.4 [230, 246]	+23.11%
4	SP	9961.44 ± 160.0 [9680, 10250]	4.31 ± 0.10 [4.10, 4.52]	278.2 ± 4.9 [269, 287]	0.00%
4	DP–IC	9198.07 ± 147.8 [8940, 9460]	4.05 ± 0.03 [3.93, 4.29]	300.2 ± 5.3 [291, 310]	−7.66%
4	DP–FR	9974.39 ± 160.4 [9690, 10260]	3.79 ± 0.09 [3.61, 3.96]	318.3 ± 5.6 [308, 328]	+0.13%

Table 4 continued

Area	Method	Total profit	Mileage / order	Accepted orders	% vs. SP (profit)
4	DP–DR–F	11930.74 ± 191.8 [11590, 12270]	2.96 ± 0.08 [2.80, 3.11]	343.6 ± 6.0 [333, 354]	+19.77%
4	DP–DR–DF	12216.61 ± 196.4 [11870, 12560]	2.94 ± 0.01 [2.79, 3.01]	346.4 ± 6.1 [335, 357]	+22.64%
4	DP–DR–DF–ATW	12669.42 ± 203.6 [12310, 13030]	2.93 ± 0.07 [2.78, 3.07]	349.7 ± 6.2 [339, 361]	+27.18%
5	SP	13388.34 ± 225.0 [12960, 13810]	4.14 ± 0.09 [4.06, 4.32]	388.1 ± 5.8 [378, 399]	0.00%
5	DP–IC	12352.08 ± 207.6 [11990, 12720]	3.96 ± 0.09 [3.77, 4.14]	408.3 ± 6.1 [397, 419]	−7.74%
5	DP–FR	13374.95 ± 224.8 [12980, 13770]	3.72 ± 0.09 [3.54, 3.90]	440.1 ± 6.6 [428, 452]	−0.10%
5	DP–DR–F	15952.98 ± 268.0 [15480, 16420]	2.94 ± 0.04 [2.80, 3.08]	458.8 ± 6.8 [447, 471]	+19.16%
5	DP–DR–DF	16350.46 ± 274.7 [15870, 16830]	2.93 ± 0.07 [2.79, 3.06]	465.3 ± 6.9 [453, 478]	+22.13%
5	DP–DR–DF–ATW	16753.96 ± 281.5 [16260, 17250]	2.93 ± 0.02 [2.79, 3.06]	468.8 ± 7.0 [456, 481]	+25.14%
6	SP	16189.31 ± 210.0 [15810, 16570]	1.00 ± 0.06 [0.92, 1.18]	442.4 ± 7.8 [428, 456]	0.00%
6	DP–IC	14850.45 ± 192.6 [14510, 15190]	0.98 ± 0.09 [0.87, 1.03]	464.3 ± 8.1 [450, 479]	−8.27%
6	DP–FR	16127.80 ± 209.2 [15750, 16500]	0.96 ± 0.02 [0.91, 0.99]	486.9 ± 8.5 [472, 502]	−0.38%
6	DP–DR–F	19219.12 ± 249.3 [18770, 19670]	0.73 ± 0.06 [0.69, 0.77]	530.4 ± 9.3 [514, 547]	+18.71%
6	DP–DR–DF	19371.27 ± 251.3 [18920, 19820]	0.74 ± 0.02 [0.70, 0.78]	531.0 ± 9.3 [514, 548]	+19.65%
6	DP–DR–DF–ATW	20186.41 ± 261.9 [19720, 20660]	0.74 ± 0.03 [0.70, 0.78]	536.2 ± 9.4 [519, 553]	+24.69%
7	SP	18616.89 ± 275.0 [18150, 19050]	0.95 ± 0.02 [0.93, 0.99]	505.3 ± 8.5 [490, 521]	0.00%
7	DP–IC	16803.12 ± 248.3 [16370, 17240]	0.89 ± 0.08 [0.78, 0.95]	529.8 ± 8.9 [514, 546]	−9.74%
7	DP–FR	17954.90 ± 265.3 [17490, 18420]	0.85 ± 0.06 [0.80, 0.91]	555.5 ± 9.4 [539, 572]	−3.56%
7	DP–DR–F	22867.61 ± 336.9 [22280, 23460]	0.73 ± 0.02 [0.68, 0.76]	629.6 ± 10.6 [611, 649]	+22.83%
7	DP–DR–DF	23254.82 ± 342.6 [22650, 23850]	0.72 ± 0.04 [0.65, 0.77]	631.8 ± 10.6 [613, 651]	+24.91%
7	DP–DR–DF–ATW	23751.69 ± 349.9 [23140, 24360]	0.73 ± 0.01 [0.69, 0.75]	635.2 ± 10.7 [616, 655]	+27.58%

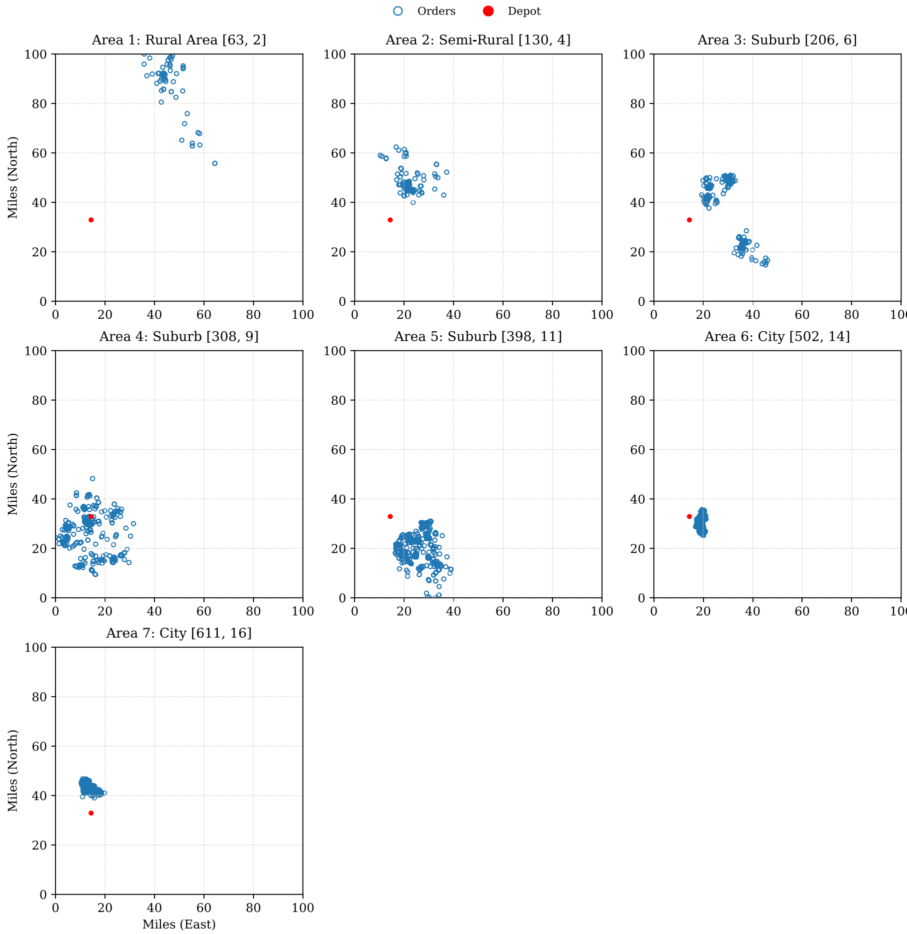

Fig. 14 Spatial distribution of forecast orders (blue hollow circles) and depot locations (red filled circles) for the seven test areas. Coordinates are shown in normalised local miles, Miles (East) on the horizontal axis and Miles (North) on the vertical axis, to make densities and spread patterns comparable across areas

Appendix D MNL choice model parameters

We simulate customer slot choice with a Multinomial Logit (MNL) model

$$U_{s'} = \beta_0 + \beta_{s'} + \beta_d \, d_{s'},$$

where $d_{s'}$ is the delivery charge for slot s'. Table 5 lists the global parameters, and Table 6 reports the slot-specific utilities β_s used for standard time windows (STWs). For Augmented Time Windows (ATWs), $\beta_{s'}$ is computed from constituent STWs via Eq. (7.1) in the main text.

Table 5 Global MNL parameters used in slot-choice simulation

Parameter	Symbol	Value
Intercept	β_0	−2.5087
Price coefficient	β_d	−0.0766

Table 6 Slot-specific utilities β_s for the 17 STWs

s	β_s	s	β_s	s	β_s	s	β_s	s	β_s
1	−1.0305	2	−0.3591	3	+0.3107	4	+0.5922	5	+0.6154
6	+0.0796	7	+0.5356	8	−0.2415	9	−0.6286	10	−1.6736
11	−0.4351	12	−0.1610	13	0	14	+0.2533	15	+0.0736
16	+0.5620	17	+0.2346						

Note. For an ATW s' formed by n_i contiguous STWs $\{stw_1, \ldots, stw_{n_i}\}$, we set as in Eq. (7.1). Experiments vary $\gamma \in [0, 0.2]$ as discussed in Section 7.4

References

Abdollahi, M., Yang, X., Nasri, M. I., & Fairbank, M. (2023). Demand management in time-slotted last-mile delivery via dynamic routing with forecast orders. *European Journal of Operational Research*, *309*, 704–718. https://doi.org/10.1016/j.ejor.2023.01.023. https://www.sciencedirect.com/science/article/pii/S0377221723000425

Agatz, N., Campbell, A. M., Fleischmann, M., Nunen, J. V., Savelsbergh, M., van Nunen, J., & Savelsbergh, M. (2013). Revenue management opportunities for internet retailers. *Journal of Revenue and Pricing Management,12*, 1–19. https://doi.org/10.1057/rpm.2012.51

Agatz, N. A. H., Fleischmann, M., & van Nunen, J. A. E. E. (2008). E-fulfillment and multi-channel distribution: A review. *European Journal of Operational Research, 187*, 339–356. https://doi.org/10.1016/j.ejor.2007.04.024

Asdemir, K., Jacob, V. S., & Krishnan, R. (2009). Dynamic pricing of multiple home delivery options. *European Journal of Operational Research, 196*, 246–257.

Campbell, A. M., & Savelsbergh, M. (2006). Incentive schemes for attended home delivery services. *Transportation Science, 40*, 327–341.

Campbell, A. M., & Savelsbergh, M. W. P. (2005). Decision support for consumer direct grocery initiatives. *Transportation Science, 39*, 313–327.

Cordeau, J., Iori, M., & Vezzali, D. (2023). A survey of attended home delivery and service problems with a focus on applications. *4OR-Q Journal of Operational Research,21*, 547–583. https://doi.org/10.1007/s10288-023-00556-2

Dong, L., Kouvelis, P., & Tian, Z. (2009). Dynamic pricing and inventory control of substitute products. *Manufacturing & Service Operations Management,11*, 317–339.

Gevaers, R., Voorde, E., & Vanelslander, T. (2014). Cost modelling and simulation of last-mile characteristics in an innovative b2c supply chain environment with implications on urban areas and cities. *Procedia - Social and Behavioral Sciences,125*,. https://doi.org/10.1016/j.sbspro.2014.01.1483

Keskin, M., Branke, J., Deineko, V., & Strauss, A. K. (2023). Dynamic multi-period vehicle routing with touting. *European Journal of Operational Research, 310*, 168–184. https://doi.org/10.1016/j.ejor.2023.02.037

Klein, R., Koch, S., Steinhardt, C., & Strauss, A. K. (2020). A review of revenue management: Recent generalizations and advances in industry applications. *European Journal of Operational Research,284*, 397–412. https://doi.org/10.1016/j.ejor.2019.06.034. https://www.sciencedirect.com/science/article/pii/S0377221719305272

Klein, R., Mackert, J., Neugebauer, M., & Steinhardt, C. (2018). A model-based approximation of opportunity cost for dynamic pricing in attended home delivery. *OR Spectrum: Quantitative Approaches in Management,40*, 969–996. https://EconPapers.repec.org/RePEc:spr:orspec:v:40:y:2018:i:4:d:10.1007_s00291-017-0501-3.

Koch, S., & Klein, R. (2020). Route-based approximate dynamic programming for dynamic pricing in attended home delivery. *European Journal of Operational Research,287*, 633–652. https://doi.org/10.1016/j.ejor.2020.04.002. https://www.sciencedirect.com/science/article/pii/S0377221720303209

Köhler, C., Ehmke, J. F., & Campbell, A. M. (2020). Flexible time window management for attended home deliveries. *Omega, 91*, Article 102023.

Lin, I., & Mahmassani, H. (2002). Can online grocers deliver?: Some logistics considerations. *Transportation Research Record Journal of the Transportation Research Board, 1817*, 17–24. https://doi.org/10.3141/1817-03

Mackert, J., Steinhardt, C., & Klein, R. (2019). Integrating customer choice in differentiated slotting for last-mile logistics. *Logistics Research,12*. https://doi.org/10.23773/2019_5. https://www.bvl.de/lore/all-volumes--issues/volume-12/issue-1/integrating-customer-choice-in-differentiated-slotting-for-last-mile-logistics

McFadden, D., et al. (1973). Conditional logit analysis of qualitative choice behavior. *Frontier in Econometrics,*1, 2.

Mintel. (2022). *Online Grocery Retailing - UK - 2022*. London: Technical Report Mintel Group Ltd.

Ötken, Ç. N., Yıldız, B., Arslan, O., & Laporte, G. (2023). Making opportunity sales in attended home delivery. *Computers & Operations Research,160*, Article 106362. https://doi.org/10.1016/j.cor.2023.106362. https://www.sciencedirect.com/science/article/pii/S0305054823002265

Punakivi, M., & Saranen, J. (2001). Identifying the success factors in e-grocery home delivery. *International Journal of Retail & Distribution Management, 29*, 156–163. https://doi.org/10.1108/09590550110387953

Snoeck, A., Merchán, D., & Winkenbach, M. (2020). Revenue management in last-mile delivery: state-of-the-art and future research directions. *Transportation Research Procedia,46*, 109–116. https://doi.org/10.1016/j.trpro.2020.03.170. https://www.sciencedirect.com/science/article/pii/S2352146520303707. The 11th International Conference on City Logistics, Dubrovnik, Croatia, 12th - 14th June 2019.

Strauss, A., Gulpinab, N., & Zheng, Y. (2020). Dynamic pricing of flexible time slots for attended home delivery. *European Journal of Operational Research, 294*, 1022–1041. https://doi.org/10.1016/j.ejor.2020.03.007

Wang, X., Zhan, L., Ruan, J., & Zhang, J. (2014). How to choose "last mile" delivery modes for e-fulfillment. *Mathematical Problems in Engineering,2014*,. https://doi.org/10.1155/2014/417129

Waßmuth, K., Köhler, C., Agatz, N., & Fleischmann, M. (2023). Demand management for attended home delivery-a literature review. *European Journal of Operational Research,311*, 801–815. https://doi.org/10.1016/j.ejor.2023.01.056. https://www.sciencedirect.com/science/article/pii/S0377221723000887

Yang, X., & Strauss, A. (2017). An approximate dynamic programming approach to attended home delivery management. *European Journal of Operational Research, 263*, 935–945. https://doi.org/10.1016/j.ejor.2017.06.034

Yang, X., Strauss, A. K., Currie, C. S. M., & Eglese, R. (2016). Choice-based demand management and vehicle routing in e-fulfillment. *Transportation Science, 50*, 473–488. https://doi.org/10.1287/trsc.2014.0549

Yildiz, B., & Savelsbergh, M. (2020). Pricing for delivery time flexibility. *Transportation Research Part B: Methodological, 133*, 230–256. https://doi.org/10.1016/j.trb.2020.01.004