Dynamic Time Slot Pricing Using Delivery Costs Approximations

Fabian Akkerman(✉), Martijn Mes, and Eduardo Lalla-Ruiz

Department of High-tech Business and Entrepreneurship, University of Twente, Enschede, The Netherlands
{f.r.akkerman,m.r.k.mes,e.a.lalla}@utwente.nl

Abstract. Attended home delivery (AHD) is a popular type of home delivery for which companies typically offer delivery time slots. The costs for offering time slots are often double compared to standard home delivery services (Yrjölä, 2001). To influence customers to choose a time slot that results in fewer travel costs, companies often give incentives (discounts) or penalties (delivery charges) depending on the costs of a time slot. The main focus of this paper is on determining the costs of a time slot and adjusting time slot pricing accordingly, i.e., dynamic pricing. We compare two time slot cost approximation methods, a cheapest insertion formula and a method employing random forests with a limited set of features. Our results show that time slot incentives have added value for practice. In a hypothetical situation where customers are infinitely sensitive to incentives, we can plan 6% more customers and decrease the per-customer travel costs by 11%. Furthermore, we show that our method works especially well when customer locations are heavily clustered or when the area of operation is sparsely populated. For a realistic case of a European e-grocery retailer, we show that we can save approximately 6% in per-customer travel costs, and plan approximately 1% more customers when using our time slot incentive policy.

Keywords: Time slot management · Dynamic pricing · Vehicle routing · Machine learning · Cost approximation

1 Introduction

During the last two decades, many e-commerce initiatives have driven the demand for package delivery services, resulting in several variations of business-to-consumer business models. One of the ultimate value-adding services is last-mile delivery, the delivery of packages to the customer's front door [10]. Home delivery services present great challenges for retailers, service providers, and logistics companies. Logistics must be organized in a way that is efficient, profitable, and satisfies the customers' wishes, while sometimes dealing with stochastic customer arrivals.

In this research, we focus on attended home delivery (AHD), for which it is necessary that the customer is at home at the delivery moment. AHD might

J. de Armas et al. (Eds.): ICCL 2022, LNCS 13557, pp. 214–230, 2022.
https://doi.org/10.1007/978-3-031-16579-5_15

be needed for security reasons (e.g., high-value goods), perishable goods (e.g., groceries), physically large goods (e.g., home appliances), or because services are performed (e.g., product installation) [2]. Many companies that offer AHD services provide their customers with time slots for choosing the delivery moment. Delivery time slots are offered to provide a high customer service and prevent costly delivery failure. When delivery has failed, the goods have to be offered for delivery at a different moment, which will result in additional storage, transportation and planning costs. In the case of perishable goods, the costs of a delivery failure are even higher, since the goods may be spoilt before the next delivery opportunity. An early study shows that AHD costs are often twice the cost of unattended delivery [26]. The AHD customer ordering process is mostly comprised of five steps: (i) the customer fills the online basket, (ii) the customer indicates the required delivery location, (iii) the customer is presented delivery time slots, (iv) the customer chooses a time slot and completes the order, and (v) the order is delivered within the required time window.

Time slots have different delivery charges as part of the company's pricing policy. Often, time slot pricing policies are intended to steer customer behavior towards time slots ("nudge") that are cheaper for the company, i.e., these time slots represent lower transportation costs. By using incentives or penalties, a company can influence customer behavior in choosing a time slot, hence, it is possible to reduce operational costs. The reduction of costs can be done by, e.g., smoothing the demand patterns or the geographical spread of customers over time to reduce demand peaks [2], reducing vehicle routing distance or time, and reducing the required fleet size.

There is limited time to perform many calculations before offering a time slot; recent research suggests that each 100-ms delay in the load time of websites can decrease sales conversion by 7% [3]. Nevertheless, we need to calculate the impact of the time slot offering in terms of, e.g., fuel, salary, vehicle rent, and emissions. In addition, the opportunity costs can be considered, which are the cost of offering a time slot now compared to saving it for potentially more profitable customers that arrive later [25]. The problem is further complicated by uncertain customer arrivals and customer behaviour. Although much research has been conducted on time slot allocation, i.e., the offering of only a subset of the feasible time slots, this study considers the situation in which always all feasible time slots are offered and we can reduce costs by nudging customers to time slots. The contributions of this paper are the application of regression models for approximating transportation costs, a novel parametric rank-based method for modelling customer behavior, and the application of our approach to a realistic time slotting case.

The remainder of this paper is structured as follows. In Sect. 2, we introduce the relevant scientific literature on attended home delivery and time slot management. In Sect. 3, we describe the problem and introduce our approximation and dynamic pricing method. Section 4 introduces the synthetic and European e-grocery retailer cases and in Sect. 5, we validate and illustrate our method using the two cases. Finally, we close with conclusions and future research directions in Sect. 6.

2 Literature

In this section, we give an overview of the state-of-the-art literature considering operational attended home delivery and time slot management. We discuss problem characteristics, solution methods, and cost approximation methods. We close with an overview of the contribution of this paper to the scientific literature.

Since attended home delivery with time slots requires delivery to take place in a specified time interval, it relates to the well-known Vehicle Routing Problem with Time Windows (VRPTW). As part of the VRPTW, the field of AHD is typically divided into the following categories: (i) static time slot allocation, (ii) dynamic time slot allocation, (iii) differentiated pricing, and (iv) dynamic pricing [1,14,25]. Time slot allocation can be summarized by the question: "what time slots should we offer to a customer?" and time slot pricing can be stated as: "what time slots should we incentivize and what time slots should be penalized?". Static methods use forecast data or static rules and can be used to make strategic and tactical decisions, e.g., to decide on the number of time slots and the width of the time slots. For differentiated allocation, the goal is to find what time slots to offer to what delivery area, e.g., certain low-populated areas might be offered fewer time slots, which is a tactical decision. Differentiated pricing tries to find the best static price policy to influence customer behavior. When time slot allocation and pricing happen online, during the decision making, it is called dynamic. Dynamic decisions can consider real-time information about the customer and the current schedule to make better decisions [1,14,25].

We review the state-of-the-art scientific literature on operational decision making techniques for attended home delivery and time slotting, see Table 1 for an overview. We consider the following problem and solution elements: (i) the delivery horizon length, which indicates how many delivery days a customer can choose for delivery, (ii) the customer arrival process, which can be modelled using different probability distributions, (iii) the order generation, which is the way the orders (e.g., quantity, location or time slot) are generated, (iv) the time slot design, which indicates what width and possible overlap of time slots is considered, (v) the time slot allocation method, and (vi) if applicable, the time slot incentive method.

In [6], a model is presented that allows for a flexible horizon, but does not consider days of the week, nor seasonality. The customer arrival process is modelled using a non-homogeneous Poisson process, as inspired by scientific work in revenue management in the airline industry (see [15]). A Markov decision process model is proposed that dynamically adjusts the delivery charges per customer. Optimal prices are calculated based on an "equal profit" policy, which means that the retailer makes the same profit in the remaining booking horizon, regardless the customer choice. Delivery prices can change based on order size, depending on the time left in the booking horizon [6]. In [9], the models are tested on fictitious cases for which customers are uniformly scattered on a 60×60 grid. Their method dynamically determines the feasibility of a time slot insertion, using a combination of insertion heuristics and randomization to determine a feasible schedule. Next, the allocation and size of incentives are determined using a linear

Table 1. Problem and solution elements in AHD and time slotting literature.

Authors	Delivery horizon	Customer arrival process	Order generation	Time slot design	Slot allocation method	Slot incentive method
[6]	Flexible	N/A	Uniform	N/A	Feasibility check	Dynamic, Markov decision process model
[9]	Single-day	Uniform	-	Non-overlapping, 1 or 2-h width	Heuristic feasibility check	Dynamic LP-based model
[11]	-	N/A	Area-based, normal dist.	8 time slots	Dynamic, ESMR	N/A
[12]	Single-day	N/A	Uniform, Demand peaks	8 Non-overlapping, 1-h width time slots	Static/Dynamic, I1 insertion heuristic	N/A
[13]	Single-day	Random	General dist. of nr. of totes $i \in \{1,...,10\}$	4 Non-overlapping, 2-h width time slots	Feasibility check	Dynamic, MILP-model for opportunity costs
[24]	Single-day	Homogeneous Poisson arrivals	General dist. of nr. of totes $i \in \{1,...,10\}$	17 Non-overlapping, 1-hour width time slots	Feasibility check	Dynamic
[25]	Single-day	Time-dependent Poisson arrivals	Normal dist.	27 Partly-overlapping, 1-h width time slots	Heuristic feasibility check	Dynamic, opportunity costs, SDP

programming model, which maximizes the profits related to time slot offerings. The authors conclude the following from their research: (i) incentive schemes can substantially reduce costs, (ii) performance of incentive schemes can be improved using intelligent methods, (iii) incentives can reduce walkaways (lost sales), (iv) it is sufficient to provide incentives to only a few slots (≤ 3), (v) an increase in time slots triggers the need for more sophisticated incentive schemes, (vi) it is easier to persuade customers to choose a wider time window than to let them choose a specific time slot, and (vii) the use of incentives can be critical already in the early stages of making a routing schedule [9].

In [11], a computational study is conducted based on the metropolitan area of Stuttgart, which is divided into nine areas with varying population sizes. Customers can choose between eight time slots. Demand is drawn from the normal distribution and is dependent on the area and the average income in those areas. There is a fixed fleet of four vehicles and capacity is estimated with vehicle routing experiments. The offering of time slots to customers is dynamically determined using the order value. The used method is called "Estimated Marginal Seat Revenue heuristic" (EMSR), as described in [8]. EMSR determines buckets for order values and allocates time slots accordingly, i.e., customers with a high order value, falling in a high-value bucket, will receive more time slot offers than customers with low order value [11]. In a study that also considers metropolitan areas, different travel time patterns are considered to model congestion in the morning peak-hours [12]. Demand for the eight non-overlapping time slots is uniform, and for some experiments demand peaks for time slots are considered. The authors define both static and dynamic approaches to determine the time slot allocation to maximize the number of accepted time slot requests. The static method uses capacity restrictions and a static rule that considers the time windows in which a delivery must be feasible. The dynamic method uses expected, dynamically determined, travel times. The authors expand this method to also have a buffer for lateness and consider stochastic travel times. Their insertion heuristic is a time-dependent adaptation to the well-known I1 insertion heuristic by [21]. In [13], 12 areas are served by a single central depot. A set of 1000 customers can arrive randomly, one at a time, and the size of demand is defined using the number of order totes. The authors develop a mixed-integer linear programming model (MILP) which is integrated into a dynamic programming model for AHD by [25]. The MILP-model maximizes expected profits and is used as an approximation of opportunity costs. The availability of time slots is checked, but time slots are always offered when capacity allows it [13]. The dynamic programming model as described in [25], is the "de facto" framework for dynamic pricing. After doing a heuristic feasibility check, based on [9], the insertion costs are calculated. The pricing solution is dynamic, but for practical reasons it does not differentiate between customers that choose the same time slot and have the same location and order value. In [25], two policies are developed, one only considering the current insertion costs, the other also including the opportunity costs. Their method is tested on a realistic case, for which bookings on a single day arrive as early as 22 days in advance, with most bookings coming in the last three days before the cutoff time. Cancellation and re-scheduling is neglected. They show that dynamic pricing methods that do not consider future expected demands (i.e., opportunity costs) can produce worse results compared to static pricing methods [25]. In a follow-up study, [24] expand their method to use an area-specific cost estimation as input for an approximate dynamic programming approach. They show that the decomposition into smaller areas can successfully reduce computational efforts and estimate the costs [24].

As [20] indicated, attended home delivery literature can also be categorised on the method for including routing costs. Most literature uses the costs resulting from explicit routing decisions, often obtained from a heuristic, since the VRPTW is NP-hard [9,11,12,25]. Alternatively, an approximation of the routing costs, without making explicit routing decisions, can be used, e.g., with Daganzo-approximation [19] or a seed-based approximation method [13,14]. Another option for routing costs approximation, not used before in time slot management research, is the use of regression models, as shown in [16] or [4].

In summary, we observe that the literature considers exclusively time slot allocation or time slot incentives. Those focusing on incentives often state that the closing of time slots for certain customers (i.e., time slot allocation) is a method that results in lost sales and customer dissatisfaction [6]. Hence, dynamic pricing is perceived as the best method, since it can balance the trade-off between lost sales and profits. Also, we see that the topic of cost approximation, being opportunity costs or transportation costs, is much studied. We recognise two different options for dynamic pricing: (i) approximate the costs of a time slot and use this as basis for setting time slot prices [9], or (ii) optimize the time slot prices, such that the behavior of customers is nudged optimally, like is done in the approximate dynamic programming model of [25]. Aside from the previous problem and solution elements, the time slotting literature also differentiates the modelling of customer choice. Most literature either use a probabilistic model, a rank-based model, or the multinomial logit (MNL) model. The latter seems to be the dominant method for the most recent literature. For more information about the MNL model we refer to [22].

The contribution of this paper to the existing scientific literature is threefold. First, we show the application of regression models for approximating transportation costs instead of the currently in use heuristic methods. Second, we present a novel parametric rank-based method for modelling customer behavior that, compared to the currently in use multinomial logit choice model, does not require behavioral data and requires fewer computations. Finally, we apply our solution approach and customer choice model to realistic time slotting case studies together with commercial vehicle routing and time slot allocation software.

3 Problem Formulation

In this section, we give the problem formulation in Sect. 3.1, describe the customer choice model in Sect. 3.2, and show how we attribute transportation costs to customers in Sect. 3.3.

3.1 Problem Characteristics

In this section, the notation of all variables, parameters, and sets is introduced, based on the formulation in [23]. We adhere to the order process as perceived by a customer. This process consists of three steps: (i) customer arrival, (ii) time slot offering, and (iii) time slot selection and confirmation. During a certain period, customers place orders at a retailer, after which the customers are offered a time slot for delivery. As common in these types of problems, we specify this period as $[0, T]$, for which 0 is the first time a customer can place an order and T is the "cutoff time", which is the last moment a customer can place an order. After T, the final delivery schedule is made for a single day, by solving a VRPTW. The customer arrival times are unknown upfront. A customer $i \in \mathcal{C}$ can arrive at any time t_i within the horizon $[0, T]$. Customer orders have a certain size, for example, indicated by weight or volume, q_i. The order quantity q_i is also unknown upfront. Each customer has a delivery service duration, i.e., the time it takes for the deliverer, after arrival at the address, to hand over the package. The service duration is indicated with l_i. The expected delivery duration can be estimated with a fixed time component and a variable time component that is dependent on the order quantity q_i.

After the customer arrival, the customer must be offered a set of time slots for delivery. We consider a single day of delivery time slots, these are all part of the set $\mathcal{T}$, with the earliest time slot beginning after T. The set of offered time slots is denoted $\mathcal{S}_i$, so that $\mathcal{S}_i \subseteq \mathcal{T}$. Time slot offering depends on feasibility and the offering policy. Each time slot $s \in \mathcal{T}$ can be of different length and can be overlapping or non-overlapping. The individual time slot duration is denoted with $[a_s, b_s]$. s is a single element in this set, i.e., $s \in \mathcal{S}_i$. Each time slot that is offered gets a certain incentive to steer customer behaviour. We consider incentives on a continuous scale, part of the incentive set $\mathcal{G}$. The incentives can be dynamically determined and differ per time slot. The incentive given to a certain time slot is $\mathcal{G}_s$. The incentives $\mathcal{G}$ are decimal numbers on the domain $[-1, 1]$, with a negative number indicating a penalty, and a positive number indicating an incentive. During the calculation time z_i, we need to determine (i) what time slots are feasible to offer to customer i, concerning both vehicle capacities and time window constraints, and (ii) the costs of offering a certain time slot. We denote the set of customers that accepted a time slot and need to be planned by $\mathcal{C}'$. The directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ models the system where nodes $\mathcal{V} = \mathcal{C}' \cup \mathcal{D}$ consist of the set of customers $\mathcal{C}'$ and the set of depots $\mathcal{D}$. Each customer $i \in \mathcal{C}'$ can be served from every depot in the set $\mathcal{D}$. The travel time on edge $(i, j) \in \mathcal{E}$ can be expressed with $\tau_{i,j}$. A single depot $d \in \mathcal{D}$ has a fixed number of vehicles L_d available for delivery. The fleet is homogeneous, where every vehicle has a capacity of H. To make a delivery, a vehicle has to visit the nodes along its route. A vehicle route always starts and ends at the same depot. For the planning of vehicle routes, we consider three constraining factors: (i) the vehicle capacity H cannot be exceeded, (ii) the vehicle routes must start and finish in the interval $[a_d, b_d]$, dependent on depot d, and (iii) the delivery of customers must be done within their selected time slot. A vehicle can leave from a customer i only after the full service duration l_i.

3.2 Customer Choice Model

To model the way customers react to time slot incentives, we develop a new rank-based choice model with a utility theory scoring component. Our approach combines two common methods found in literature, namely, a rank-based model and a parametric utility theory model, see [14] and [22] for recent examples of both modelling types. We model customer preference as follows. A customer has a ranking for all time slots, i.e., the first preferred time slot is ranked highest and the least preferred time slot is ranked lowest, as is normal for rank-based models. The ranking of time slots is based on scores and, therefore, the ranking can be influenced by incentives, similar to models based on utility theory, e.g., the multinomial logit model.

Each customer gives "base scores" to all time slots, expressed with $K_i \subseteq \mathcal{T}$. For our experiments, we use a preference list that includes all time slots, i.e., $|K_i| = |\mathcal{T}|$. We model different types of customers. Some customers can be seen as "rigid", and others are perceived as more "sensitive" to incentives. The level of sensitivity is expressed with f_i, which is a continuous parameter on the scale $[0, 1]$, with 0 being rigid and 1 sensitive. The incentive effectiveness is directly related to the sensitivity parameter f_i of a customer. We do not know the customer sensitivity upfront.

We define the number $\beta_{i,s}$ as the base score of a time slot s, with $\beta_{i,s}$ on the domain $[\frac{1}{|K_i|}, 1]$, with $|K_i|$ being the number of time slots in the base preference list of customer i. The assignment of scores to time slots is done in a decreasing fashion, i.e., the first preference gets the highest score (1), and the last preference gets the lowest score ($\frac{1}{|K_i|}$). The lowest possible score is $\frac{1}{|K_i|}$ instead of 0 because this prevents problems when there are only few time slots and the difference in base score is too large for incentives to have any effect. The equation to determine base scores $\beta_{i,s}$ is given by:

$$\beta_{i,s} = \frac{|K_i| - k_{i,s} + 1}{|K_i|}, \tag{1}$$

where $\beta_{i,s}$ is the base score for time slot s of customer i, $|K_i|$ is the number of time slots in the preference list of customer i, and $k_{i,s}$ is the randomly drawn ranking of time slot s for customer i, where the ranking is an integer number $k_s \in \{1, 2, \ldots, |K_i|\}$. We can influence the ranking of the base preferences using incentives. The incentive decision must be made for all feasible time slots. The incentives we can give are continuous numbers with $\mathcal{G}_s$ on the domain $[-1, 1]$. A negative incentive can be interpreted as a penalty. The incentives are multiplied by the customer sensitivity f_i, and then added to the base preference scores. Next, the list is re-ordered from high to low and the customer chooses the highest ranking time slot that is offered, as common for utility theory models. The total score of a time slot for a customer is expressed with $u_{i,s}$ and is calculated using Eq. 2, as is common for utility theory models [22].

$$u_{i,s} = \beta_{i,s} + f_i \cdot \mathcal{G}_s. \tag{2}$$

3.3 Determining Transportation Costs

To obtain the routing costs per customer, we need to do some transformations with routing data. These transformations are necessary since we need to find the costs of adding a customer, but we only have the total routing costs of the VRPTW, i.e., the total costs need to be divided over the customers. We use a method we call "half-edge partitioning" (HEP), which can be applied to most VRP and VRPTW solutions. HEP is a straightforward, but slightly simplistic method that allocates half of the costs (time or distance) needed to travel an edge to the customer from which the edge departs, and the other half to the customer at which the edge arrives. The edges that depart from and arrive at the depot are partially allocated to their arriving and departing customers, respectively. The other half of these depot edges are equally divided over all customers. The routing costs, expressed in travel time or distance, of a single customer c served by a vehicle that serves a set of customers $\mathcal{C}'$, can be expressed as:

$$\text{Travel costs of customer } c = \frac{1}{|\mathcal{C}'|}(0.5t_{d,f} + 0.5t_{l,d}) + 0.5t_{i,c} + 0.5t_{c,j}, \tag{3}$$

with $t_{i,j}$ being the travel time or distance in the final routing schedule on edge (i, j), where i and j are the locations visited before and after customer location c, respectively. The depot is indicated with d, and customer f and customer l are the first and last customer of a vehicle route, respectively. For our experiments, we use travel time as the cost factor.

4 Solution Approach

In this section, we first describe the cheapest insertion method as cost approximation in Sect. 4.1. Next, we describe the engineered features used in our cost approximation regression model in Sect. 4.2. We show how we obtain training data in Sect. 4.3 and finally, we describe our time slot incentive policy in Sect. 4.4.

4.1 Cheapest Insertion Transportation Cost Approximation

The idea of the cheapest insertion cost approximation is relatively simple: during the booking horizon, we keep track of a preliminary routing schedule that contains all booked customer orders up to the respective moment. This preliminary routing schedule is sequentially constructed using cheapest insertion, and periodically re-optimized after every 20^{th} customer arrival. This re-optimization interval strikes a balance between computational effort and performance for our experiments. We use a commercial vehicle routing solver [17] for re-optimization. When a new customer arrives, the cheapest insertion algorithm calculates how much it would cost, in terms of travel time, to add the new customer to a vehicle route. The cheapest insertion algorithm returns the costs of insertion for every feasible time slot. These costs differ per time slot, since vehicles that serve customers in the same time window may be close by, or alternatively have to make a detour. Cheapest insertion is simple, fast and dynamic, since it uses all current customer information for estimating costs. Nevertheless, it has the disadvantage of being myopic, i.e., it makes the best decision at a point in time, but cannot make a forecast about future customers.

4.2 Regression-Based Transportation Costs Approximation

As discussed in [4], we use a regression model to approximate transportation costs. For this paper, we show the results of random forests regression, since this method is able to fit complex functions without too much computational time. To make transportation costs predictions, we need to supply features to the model. Therefore, we aggregate customer and routing information using area-time slot clusters (ATC), as common in the literature [13,25]. The following information of an ATC is stored: customer locations expressed in latitude and longitude, customer order volume expressed in kilograms, and the routing costs per customer. Aggregation-based features give a synopsis of the characteristics of an *area* and *time slot* cluster (ATC) a customer is in. For every feasible time slot option, we calculate the feature values *before* and *after* the potential insertion of the new customer, to obtain the expected increase in routing costs. The engineered features are based on the features proposed in [4]. Examples of these features are: the number of customers in an ATC, the number of days between customer arrival and the end of the horizon, the distance between the depot and the ATC-centroid, the variance of the angles between customers in an ATC and the depot and the average distance between customers in an ATC. A complete overview of features, including a short description of each feature, and the data partition over which each feature is calculated, can be found in Table 2.

Table 2. Summary of features used for the regression model.

Feature	Feature description	Data partition
Days until the cutoff time (F1)	The number of days left at the arrival of the customer until the cutoff time	N/A
Number of customers in ATC (F2)	The number of customers accepted in the ATC	ATC
Haversine distance from ATC centroid to depot (F3)	The distance from the centroid of all accepted customers in ATC to the depot	ATC
Average distance between customers in an ATC (F4)	The average distance between all accepted customers in ATC	ATC
Variance customer-depot bearing (F5)	The variance of the bearings between the customers in ATC and the depot	ATC
Average customer-depot bearing (F6)	The mean of the bearings between the customers in ATC and the depot	ATC
Area ID (F7)	Binary vector indicating the area	$\mathcal{A}$
Time slot ID (F8)	Binary vector indicating the time slot	$\mathcal{S}$
Variance of time slot population (F9)	The variance of the number of accepted customers per time slot in area $a \in \mathcal{A}$	$a \in \mathcal{A}$
Time slot distance (F10)	The distance measured in time slots between the first and last populated time slot in area $a \in \mathcal{A}$	$a \in \mathcal{A}$
Number of time slots (F11)	The number of booked time slots in $a \in \mathcal{A}$	$a \in \mathcal{A}$

4.3 Obtaining Training Data

We use a simulation model to test different methods and policies. To train our methods, we need to obtain data. We do this by generating a separate set of instances and running full simulations on these. For these training instances, we do not use any nudging policy, i.e., customers choose the offered time slot that has the highest base score $\beta_{i,s}$. We obtain the following data after a simulation run: (i) a final VRPTW-schedule, (ii) all customer locations, and (iii) the time slots chosen by customers.

4.4 Simple Incentive Policy

To test the quality of the cost approximations and the effect they can have on a dynamic pricing policy, we present a simple dynamic pricing policy that uses the approximated costs per time slot, and subsequently, returns time slot prices. The time slot prices are always on the domain $[-1, 1]$ and can be tuned using a parameter. After obtaining a cost approximation for all feasible time slots, given by the set $\mathcal{S} \subseteq \mathcal{T}$, we first calculate the mean $\overline{C}_{\mathcal{S}}$ and standard deviation $\sigma_{C_{\mathcal{S}}}$ of the predicted costs over all feasible time slots. Next, we calculate the difference between the predicted costs for time slot s and the mean estimated costs over all time slots $\mathcal{S}$:

$$\widehat{c}_s = -1 \cdot \left(c_s - \overline{C}_{\mathcal{S}}\right). \tag{4}$$

We multiply with -1 to give higher incentives to the time slots with low costs and vice versa. Next, we use a tunable parameter W multiplied by the standard deviation $\sigma_{C_\mathcal{S}}$ to control how much standard deviations distance from the mean $\widehat{c}_s$ is considered large, and adjust the magnitude of incentives accordingly. In our experiments, we tuned W and found that $W = 1.0$ gives best results. In case that $W\sigma_{C_\mathcal{S}} \ll \widehat{c}_s$, we cap the incentives to remain in the domain $[-1, 1]$. When the costs for all the time slots are the same, i.e., $\sigma_{C_\mathcal{S}} = 0$, no incentives are given:

$$\text{Incentive for time slot s} = \begin{cases} 0, & \text{if } \sigma_{C_\mathcal{S}} = 0, \\ -1, & \text{if } \frac{\widehat{c}_s}{W\sigma_{C_\mathcal{S}}} \leq -1, \\ \frac{\widehat{c}_s}{W\sigma_{C_\mathcal{S}}}, & \text{if } -1 < \frac{\widehat{c}_s}{W\sigma_{C_\mathcal{S}}} < 1, \\ 1, & \text{if } \frac{\widehat{c}_s}{W\sigma_{C_\mathcal{S}}} \geq 1. \end{cases} \tag{5}$$

5 Case Studies

In this section, we explain the two cases on which we test our transportation cost approximation approach. First, we describe the synthetic case in Sect. 5.1 and then a realistic European e-grocery retailer case in Sect. 5.2.

5.1 Synthetic Case

We generate instances with a single depot from which a fleet of 20 vehicles serves an area of 50 kilometer radius from the depot. The customers are generated in a randomly clustered (RC) pattern, i.e., 80% of the customer belongs to one of the eight customer clusters, while 20% of the generated customers have a random location. During a booking horizon of 21 days, 750 customers can request a time slot. We offer six non-overlapping time slots of 2-hour width. Customers have a base preference list that entails all six time slots, i.e., customers can be nudged to every time slot that is feasible. In practice, VRPTWs have both a vehicle capacity restriction and a time window restriction. For these experiments, we first study the effect of only having a time window (RC-T) restriction, and next, the effect of adding a capacity restriction of 25 customers per vehicle (RC-TC). We use two different customer price sensitivity settings: one for which customers are infinitely flexible (Flex), i.e., customers will always choose the time slot that we nudge; and a second one that uses a sensitivity of $f_i = 1$ (see Sect. 3.2), i.e., customers are sensitive but the time slot with the highest incentive is not necessarily always chosen, since the base scores, before incentives, have influence on the eventual time slot choice.

5.2 European E-grocery Retailer

One of the main contemporary application areas of time slotting is e-grocery retailing, i.e., offering the possibility to order groceries online and delivering them at home. The reason that grocery retailers use time slots for delivery is that the goods are often perishable, so a failed delivery can be costly. Compared to the synthetic case, customers are dispersed over a larger region containing cities and rural areas. The grocery retailer has a heterogeneous fleet, with smaller vehicles used for cities and larger vehicles for rural areas. The retailer offers seven overlapping time slots, and serves from multiple depots (4) with different fleet sizes. For the individual instances, we use order data obtained from the same day of the week, to prevent seasonality differences. The fleet is heterogeneous in terms of vehicle capacity and driving speed. The retailer offers five overlapping time slots of 2-h width, and two time slots of 4 and 5 h width. Customers arrive on a booking horizon of 9 days, and on average instances have ∼2000 customer arrivals. Since some of our features are calculated based on the depot location, but we do not know upfront which depot serves a customer area, we always use the main depot for feature value calculations.

6 Computational Experiments

We use a simulation model that mimics customer behavior and integrates commercial time slot allocation and vehicle routing services. The simulation model is built in C# and maintained by the Math Innovation Team from the software development company ORTEC. All cost approximation methods have been trained using the Python Scikit-learn library [18] and are loaded in C# using the ONNX standard artificial intelligence format [7]. The general event structure of the simulator follows the following events: (i) a customer arrives and requests a time slot offering, (ii) a feasibility check for every time slot is done using cheapest insertion and the feasible time slots are offered to the customer, (iii) the customer chooses a time slot, (iv) the customer choice is recorded in the system. A commercial VRP solver is called after every 20^{th} customer arrival to update the intermediate routing schedule, and after the final customer arrival to obtain the final routing schedule. For a full description of the simulation model, we refer to [5,23].

We report six different statistics: (i) the percentage of all customers that could be planned and served, (ii) the average number of time slots that were feasible to offer to a customer, (iii), the percentage of customers that were nudged to a different time slot than their first preference, (iv) the average travel time per customer in minutes, (v) the average waiting time per customer in minutes, and (vi) the average travel distance per customer in kilometers. For both cases, the travel time and the travel distance are calculated with the actual road network costs, using the commercial VRP solver. Traffic congestion has not been accounted when calculating travel times. Waiting time is reported because it is an essential element of VRPTWs: potentially, driving times can be low, however, early arrivals at customer locations result in drivers having to wait.

In the remainder of this section, we first show the results for a synthetic case study in Sect. 6.1, using generated instances based on real data. For these instances, we alter problem attributes to test our approach in different settings. Next, in Sect. 6.2, we use a real case from an European e-grocery retailer to validate our approach in a realistic setting.

6.1 Results for the Synthetic Case

Table 3 summarizes the experimental results. First, we show results for the RC type without time window restriction, that is, RC and RC-C, respectively. Next, we add the time window restriction and show results for the case without a time slot incentive policy. The results show that the addition of time slots, disregarding incentives, causes a significant decrease (19.9%) in the number of served customers for the uncapacitated instance. For the capacitated instances, the difference in served customers is insignificant. However, for both the uncapacitated and the capacitated instances, the addition of time slots causes a large increase in travel time, waiting time, and travel distance, e.g., the travel distance increases by 110.7% and 68.5% for the RC and RC-C instances, respectively.

Table 3. Simulation run statistics on the randomly clustered instances with a time restriction (RC-T) or capacity restriction (RC-TC), using 5 replications.

Offer strategy	Instance	Planned customers (%)	Avg. no. of feasible TS	Nudged customers (%)	Travel time/ customer (min.)	Waiting time/ customer (min.)	Distance/ customer (km)
No time slots	RC	100%	N/A	N/A	9.02	0.38	6.94
No time slots	RC-C	66.7%	N/A	N/A	12.0	0.56	9.66
No incentive	RC-T	80.1%	4.3	N/A	17.46	0.72	14.62
No incentive	RC-TC	66.6%	3.8	N/A	21.74	3.31	16.28
IC	RC-T Flex	84.9%	4.8	80.1%	15.70	0.65	13.15
RFR	RC-T Flex	84.2%	4.8	78.4%	15.68	0.64	14.25
IC	RC-TC Flex	65.9%	3.9	81.3%	21.81	1.81	15.36
RFR	RC-TC Flex	66.6%	3.2	79.4%	24.78	1.01	15.37
IC	RC-T	85.5%	4.7	58.2%	15.54	0.91	12.49
RFR	RC-T	81.9%	4.0	66.5%	16.89	0.86	14.03
IC	RC-TC	66.6%	3.9	58.6%	21.93	4.77	14.98
RFR	RC-TC	66.7%	3.7	66.3%	22.78	4.81	15.96

When we add our incentive policy, either based on the insertion costs (IC) or the random forests regression (RFR) model, we see that we can significantly reduce operational costs and increase the number of served customers for the case with infinitely flexible customers (Flex). For the uncapacitated case, we can plan on average 4.5% more customers and decrease travel distance by 5.5%. For the capacitated case, the incentive policy can reduce travel distance by 5.6%. The IC method most often outperforms the RFR method. Possibly this is caused by the frequent updates of the vehicle routing plan (after every 20^{th} customer arrival), which makes IC more reliable. Finally, we show results for the case with more realistic customer sensitivity, i.e., when time slot incentives do not always have an effect. For most cases, this causes a drop in performance, although we are still able to significantly reduce costs compared to the situation without incentives.

6.2 Results for the European E-grocery Retailer

Table 4 shows the results for the real case study of an European e-grocery retailer. We observe from the "No time slots" experiment that we cannot plan more than 81.1% of the customers due to vehicle capacity restrictions. The addition of time slots causes an increase in travel time and travel distance of 34.5% and 33.9%, respectively. We observe that IC and RFR both can plan more customers when nudging to infinitely flexible customers, compared to the situation without incentives. IC saves 15.7% in travel time and 15.0% in distance per customer, compared with the situation without incentives. RFR improves slightly less compared with the situation without incentives; it saves 7.3% in travel time and 11.2% in distance per customer. Comparing the situation without incentives with the best performing incentive policy setting, we see 0.7% more planned customers, 6.2% less travel time, and 5.3% less traveled distance per customer. Waiting times are low, and the differences between waiting times are insignificant. Again, IC shows somewhat better performance in most statistics, but RFR seems to be the more "active" policy with more nudging.

Table 4. Simulation run statistics for the European e-grocery retailer case, using 2 replications.

Offer strategy	Planned customers (%)	Avg. no. of feasible TS	Nudged customers (%)	Travel time/ customer (min.)	Waiting time/ customer (min.)	Distance/ customer (km)
No time slots	81.1%	N/A	N/A	4.18	0	2.53
No incentive	80.4%	5.6	N/A	5.62	0.03	3.39
IC (Flex)	81.1%	5.7	85.6%	4.74	0	2.88
RFR (Flex)	81.0%	5.6	86.8%	5.21	0.02	3.01
IC	80.5%	5.6	45.4%	5.25	0.02	3.15
RFR	81.0%	5.5	75.1%	5.84	0.08	3.96

7 Conclusions

We explored the possibilities for improving time slot solutions by approximating the costs of adding a customer to a time slot, and subsequently, we studied the effects of dynamic pricing based on these cost approximations. To model customer behavior without the need for a behavioral study, we developed a parametric rank-based customer choice model for which we can influence the ranking of time slots by giving incentives or penalties. We developed a simple incentive policy to test our customer choice model and time slot cost approximation. Our solution approach for approximating the costs of time slots is centered around the prediction of transportation costs using regression models. To improve prediction and reduce noise, we aggregated customers in area-time slot combination (ATC) clusters, after which we trained regression models to predict the travel times to serve an ATC-cluster. We tested our proposed solution on two different cases. First, we ran several experiments with a synthetic case, i.e., a case with generated data using both a time-constrained variant and a capacity-constrained variant. For the second case, we used data from an European e-grocery retailer.

When we compared the situation without time slots, i.e., customers can be planned the whole day, with the situation with time slots, we saw a decrease of the percentage of customers that can be served (∼20%), and a significant increase in travel time, waiting time, and distance per customer. When giving incentives, we can plan 6% more customers and decrease travel time, waiting time and distance per customer by 11% compared to the situation without incentives. Our random forests method often performed similar or slightly worse compared to the insertion costs (IC) method. For the Europen e-grocery retailer case, IC could save in travel times (−15.7%) and distance (−15.0%) per customer, while planning slightly more customers compared to the case without incentives. Our random forests method planned a similar number of customers, and saved 7.3% in travel time and 11.2% in distance per customer, respectively.

Further research can be done on the aggregation structure used for aggregating customers in spatial areas, e.g., using adaptive grids that automatically identify customer clusters. Our rudimentary incentive policy could be improved by improving the cost approximation, e.g., by considering more features or using other supervised learning approaches, e.g., neural networks. The definition of transportation costs is another interesting aspect that requires more research since the half-edge partitioning method we used could be improved to consider more than only travel time or distance. Although we studied the correlation between customer time slots and costs, there is a lacking *causality* between giving incentives and total transportation costs. Hence, the dynamic nature and complexity of the time slotting cause a disconnect between our time slot cost approximation, time slot incentive policy and the final costs. Potentially, a (deep) reinforcement learning model could be valuable for learning this implicit relationship.

Acknowledgements. This paper is based on the master thesis of Fabian Akkerman supervised by Martijn Mes and Eduardo Lalla-Ruiz. We would like to thank Thomas Visser of ORTEC for his advice and assistance during the graduation process.

References

1. Agatz, N., Campbell, A., Fleischmann, M., Nunen, J., Savelsbergh, M.: Revenue management opportunities for internet retailers. J. Revenue Pricing Manage. **12**, 128–138 (2013). https://doi.org/10.1057/rpm.2012.51
2. Agatz, N., Campbell, A.M., Fleischmann, M., Savelsbergh, M.: Challenges and opportunities in attended home delivery. In: Golden, B., Raghavan, S., Wasil, E. (eds.) The Vehicle Routing Problem: Latest Advances and New Challenges. Operations Research/Computer Science Interfaces, vol. 43, pp. 379–396. Springer, Boston (2008). https://doi.org/10.1007/978-0-387-77778-8_17
3. Akamai: Akamai online retail performance report: milliseconds are critical (2017). https://www.akamai.com/uk/en/about/news/press/2017-press/akamai-releases-spring-2017-state-of-online-retail-performance-report.jsp
4. Akkerman, F., Mes, M.: Distance approximation to support customer selection in vehicle routing problems. Ann. Oper. Res., April 2022. https://doi.org/10.1007/s10479-022-04674-8
5. Akkerman, F.: Delivery cost approximations for dynamic time slot pricing, April 2021. http://essay.utwente.nl/86079/
6. Asdemir, K., Jacob, V.S., Krishnan, R.: Dynamic pricing of multiple home delivery options. Eur. J. Oper. Res. **196**(1), 246–257 (2009). https://doi.org/10.1016/j.ejor.2008.03.005
7. Bai, J., Lu, F., Zhang, K., et al.: ONNX: Open neural network exchange (2019). https://github.com/onnx/onnx
8. Belobaba, P.: Air travel demand and airline seat inventory management. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge (1987)
9. Campbell, A., Savelsbergh, M.: Incentive schemes for attended home delivery services. Transp. Sci. **40**, 327–341 (2006). https://doi.org/10.1287/trsc.1050.0136
10. Campbell, A.M., Savelsbergh, M.W.P.: Decision support for consumer direct grocery initiatives. Transp. Sci. **39**(3), 313–327 (2005). https://doi.org/10.1287/trsc.1040.0105
11. Cleophas, C., Ehmke, J.: When are deliveries profitable? Bus. Inf. Syst. Eng. **6**, 153–163 (2014)
12. Ehmke, J.F., Campbell, A.M.: Customer acceptance mechanisms for home deliveries in metropolitan areas. Eur. J. Oper. Res. **233**(1), 193–207 (2014). https://doi.org/10.1016/j.ejor.2013.08.028
13. Klein, R., Mackert, J., Neugebauer, M., Steinhardt, C.: A model-based approximation of opportunity cost for dynamic pricing in attended home delivery. OR Spectr. **40**(4), 969–996 (2017). https://doi.org/10.1007/s00291-017-0501-3
14. Klein, R., Neugebauer, M., Ratkovitch, D., Steinhardt, C.: Differentiated time slot pricing under routing considerations in attended home delivery. Transp. Sci. **53**(1), 236–255 (2019). https://doi.org/10.1287/trsc.2017.0738
15. Lee, T.C., Hersh, M.: A model for dynamic airline seat inventory control with multiple seat bookings. Transp. Sci. **27**(3), 252–265 (1993)
16. Nicola, D., Vetschera, R., Dragomir, A.: Total distance approximations for routing solutions. Comput. Oper. Res. **102**, 67–74 (2019)
17. ORTEC: Vehicle routing solutions. https://ortec.com/en/solutions/vehicle-routing. Accessed 14 July 2022
18. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)

19. Robuste, F., Daganzo, C.F., Souleyrette, R.R.: Implementing vehicle routing models. Transp. Res. Part B Methodol. **24**(4), 263–286 (1990)
20. Snoeck, A., Merchán, D., Winkenbach, M.: Revenue management in last-mile delivery: state-of-the-art and future research directions. Transp. Res. Procedia **46**, 109–116 (2020). https://doi.org/10.1016/j.trpro.2020.03.170. The 11th International Conference on City Logistics, Dubrovnik, Croatia, 12–14 June 2019
21. Solomon, M.M.: Algorithms for the vehicle routing and scheduling problems with time window constraints. Oper. Res. **35**(2), 254–265 (1987). https://doi.org/10.1287/opre.35.2.254
22. Strauss, A.K., Klein, R., Steinhardt, C.: A review of choice-based revenue management: theory and methods. Eur. J. Oper. Res. **271**(2), 375–387 (2018). https://doi.org/10.1016/j.ejor.2018.01.011
23. Visser, T., Agatz, N., Spliet, R.: Simultaneous customer interaction in online booking systems for attended home delivery. Technical report, ERS-2019-011-LIS, ERIM Report Series Research in Management, Erasmus Research Institute of Management, October 2019. http://hdl.handle.net/1765/120585
24. Yang, X., Strauss, A.: An approximate dynamic programming approach to attended home delivery management. Eur. J. Oper. Res. **263**, 935–945 (2017). https://doi.org/10.1016/j.ejor.2017.06.034
25. Yang, X., Strauss, A.K., Currie, C.S.M., Eglese, R.: Choice-based demand management and vehicle routing in e-fulfillment. Transp. Sci. **50**(2), 473–488 (2016). https://doi.org/10.1287/trsc.2014.0549
26. Yrjölä, H.: Physical distribution considerations for electronic grocery shopping. Int. J. Phys. Distrib. Logist. Manag. **31**(10), 746–761 (2001)