Incentive Schemes for Attended Home Delivery Services

Bachelor Thesis Econometrics and Operations Research

ERASMUS UNIVERSITEIT ROTTERDAM
ERASMUS SCHOOL OF ECONOMICS

Rogier H. Blokland
409043

Supervisor: Thomas R. Visser MSc
Second Assessor: Dr. Remy Spliet

Abstract

Many companies using attended home delivery services have found the delivery schedule to be a major logistic challenge. Customers choosing their time windows pose major restrictions on the delivery schedule, as the time windows are often small and customers near each other might choose different windows. In this paper, we therefore studied the use of incentives to influence customers to choose a time window more convenient for the planning of the delivery schedule. Because the future orders are unknown to a vendor, the time windows a customer might select are too. Therefore, we also analysed cases of incorrect estimates of customer behaviour. We used simulation to evaluate the proposed methods for distribution of the incentives. While we found flat incentives to influence the profit positively, incentives based on more sophisticated methods create a larger improvement. Furthermore, if the customers' behaviour is unknown, the extra profit which can be made, will decrease if the vendor is unable to estimate the behavior correctly. However, offering incentives will still be of use if the vendor makes a partially correct estimate.

July 2, 2017

Contents

1 Introduction

Home delivery of products to customers has a strongly rising share in today's commerce, as currently more customers order their goods online. While unattended deliveries already are a major logistic challenge for companies, attended deliveries are an even bigger one. To ensure customers are at home to receive the delivery, companies allow customers to choose a certain time window. This has consequences for the efficiency of truck delivery routes, as customers nearby each other might choose different time windows, causing very cost inefficient delivery schedules and the loss of potential customers. If customers can be influenced to choose more convenient time windows for the planning of the delivery schedule, profit could be improved. However, the problem gets even more complicated, as all customers choose their time window at different moments in time.

In this thesis, we consider the use of incentives to influence customers to choose convenient time windows, such that the costs of the delivery schedules will be decreased and more customers will fit into the schedule. At first, the methods proposed by Campbell and Savelsbergh (2006) will be implemented, following a reproduction of most of their results. The authors studied the influence of incentives in the form of discounts on the customers' purchase price. Multiple methods to decide on the distribution of the incentives are investigated, of which the results are compared and analysed. In their research, a simple customer behaviour model in which the probabilities of the choice on all time windows are known by the vendor is used. It is also assumed the change in the probabilities by offering incentives is known by the vendor. The authors argue that in the near future, quite accurate predictions on the preferred time windows can be made, because the possession of large amounts of data becomes more common. However, some customers might behave inconsistent or keep ordering rarely, such that the behaviour remains unpredictable in the future. Thus, offering incentives based on these estimates might lead to different results, since these might be incorrect. In this paper, we therefore investigate the influence of incorrect expected customer behaviour to measure the robustness of the methods proposed in Campbell and Savelsbergh (2006).

The thesis is structured as follows. First, a short review on relevant literature is given. Then, in Section 3, the problem is introduced and explained in more detail. In Section 4, we propose a method to optimise our delivery schedules and we introduce multiple manners for incentive distribution. Also, we look in more detail into the consequence of incorrect estimates. Afterwards, we present our results in Section 5 for both correct and incorrect estimates of customer behaviour. Furthermore, the incentive calculations might be quite costly in terms of computation time, while customers are not fond of long waiting times during their online purchase. Therefore, we will report the computation times of our methods. Lastly, in Section 6, we discuss our results and present our conclusions.

2 Literature review

In an earlier research, Campbell and Savelsbergh (2005) conduct research on the acceptance of orders in home grocery deliveries to maximise profit. The authors mainly evaluate whether to accept a new order or wait for other orders depending on information about possible coming orders. After the arrival of a new order, the value of the order is compared with potential future orders, taking into account the probability of actually having that order realised. Campbell and Savelsbergh found that not accepting all orders on a first come, first serve basis can significantly improve the profit. As orders could arrive quickly after each other, the running time of the algorithms should be as small as possible. The algorithms used in Campbell and Savelsbergh (2005, 2006) are based on insertion heuristics, of which the running time is investigated in Campbell and Savelsbergh (2004). The authors were able to provide a method to lower the running time of the insertion heuristic from $O(n^4)$ to $O(n^4)$.

Bent and Van Hentenryck (2004) study a partially dynamic vehicle routing problem with time windows. Part of the customers are known in advance, while it is assumed stochastic information about dynamic customers. The authors use this stochastic information to accept or exclude a customer to maximise the number of serviced customers. To achieve this, Bent and Van Hentenryck proposed a method to continuously generate multiple delivery schedules which include both known and stochastic requests. While the authors do not propose the use of incentives, the scheduling algorithm used by Bent and Van Hentenryck is based on a similar idea as in Campbell and Savelsbergh (2006), as many new schedules are also created from scratch.

At the time of the research of Campbell and Savelsbergh in 2006, the concepts of e-commerce were very young and not much similar work had been done before this paper. However, Madsen, Tosti, and Vælds (1995) introduced a problem very similar to the one considered in this paper. They considered the dispatchment of repairmen. Customers could choose restricted time windows where they wanted to have the repairmen do their job. Madsen et al. tried to find an optimal strategy for scheduling the repairmen, such that the travel time would be minimised. The difference however, is that the vendor itself chose the time window in which the visit to the customer would be made. They did find a working heuristic, based on clustering first, scheduling second. They also found that the selection of time windows is an important step in creating a good result, such that the routing can be done efficiently.

Because the decision on the offered time windows seems to be of great importance, Agatz, Campbell, Fleischmann and Savelsbergh (2011) conducted a research closely related to Campbell and Savelsbergh (2006). Agatz et al. (2011) investigated the tactical allocation of available time windows per urban area. Because some areas have a lower demand than other ares, the authors want to cluster the deliveries in the same areas in a limited number of time windows. Therefore, the authors determined the number of time windows per zip code, after which specific time windows resulting a delivery schedule with low costs were determined. The results stress the importance of finding a balance between the concentration and spreading of deliveries in time windows. This means that it is important to have customers select time windows such that deliveries will be made in (almost) equal time windows, if their addresses are close to each other. Furthermore, the authors developed methods to make a selection of possible time windows for certain areas, such that costs can be increased even before scheduling.

3 Problem description

3.1 The HDPTI

The first problem described in Campbell and Savelsbergh (2006) is the Home Delivery Problem with Time Window Incentives (HDPTI). The problem uses attend home deliveries, such that customers choose a time window when they make their purchase online. Afterwards, a delivery schedule is made as efficiently as possible. The authors modified the problem as follows: when a customer orders its product, they determine the efficiency of including the delivery in all time windows. Now, an attempt to lure the customer into choosing a convenient time window can be made, in the form of discounts on the purchase price, so called incentives. Afterwards, the order is added into a preliminary schedule, after which a new customer makes a purchase. The objective of the problem is maximising the overall profit, thus revenue of executing the orders minus all given incentives and made delivery costs.

In this problem, delivery routes need to be created for a known day, where customers may make an order up to a certain time P, at least a day before the day the first delivery is made. For example, P will lie ultimately on Monday if all deliveries will be made on Tuesday. An order i has characteristics known to the vendor. These characteristics consist of the revenue, r_i, the customers' coordinates, x_i and y_i, a capacity d_i, a delivery handling time h_i and an estimate of the windows a customer might choose, including a preference for one window.

To solve the problem, the vendor uses m vehicles, all with a capacity Q. The planning horizon consists of $|T|$ non-overlapping time windows t, where T is the set of all time windows. All t have a starting time, $begin^t$, and ending time, end^t. Each t also has a probability to be chosen, which differ per customer i, denoted as p_i^t, which are known to the vendor. Furthermore, the travel time between order i and j is given by $t_{i,j}$, with a travel cost per unit c.

3.2 Incentives

After an order i comes in, the incentives are calculated per window, denoted as I^t. These consist of discounts on the purchase price of customer i. Only time windows where $p_i^t > 0$ are eligible for incentives, because time windows where $p_i^t = 0$ might be infeasible, or very inconvenient and will therefore never be chosen. The windows where $p_i^t > 0$ are represented in the set O. The set O is split in a set U in V, consisting of time windows which may and may not receive an incentive, respectively. In advance, it is also determined that at most l time windows may receive an incentive. How to divide the windows into U and V, depends on the methods, which are explained in Section 4.2. This also depends on the possibility to stop ordering when the time windows occur, such that a customer can walk way, of which both options are investigated.

In the case of not allowing customers to walk away, when the probability of the time windows in U increase, the windows in V must compensate for this with a decrease to keep the sum of all p_i^t at one. This total reduction in probability is equally divided among all time windows in V, and the reduction in probability per window to maximise expected profitability is denoted by z. Because the other time windows, where $p_i^t = 0$, can not decrease, these are left out of consideration. This reduction in the $p_i^t > 0$ causes the maximum incentive which can be offered to have a natural restriction, since windows can not decrease endlessly. However, the vendor might want to use smaller incentives, such that a limit B on a single incentives is used.

If the vendor allows customer to stop their purchase when selecting a time window, the set V is adjusted and a set F is added. All $t \in V$, for which $p_i^t > 0$, but no feasible insertion exists, will enter the set F. Then U consists of time windows eligible to receive incentives, V of windows where $p_i^t > 0$ and a feasible insertion exists, while F consists of windows where $p_i^t > 0$, but no feasible insertion exists.

3.3 Incorrect customer behaviour estimates

As discussed in Campbell and Savelsbergh (2006), the problem assumes very strict knowledge on customer behaviour. This assumption could be left unsatisfied, which is why inaccurate customer behaviour estimates will be considered. Even though methods to capture and predict the willingness of customers choosing certain time windows evolve, a customer might be inconsistent, have an irregular schedule or there just does not exist a record of earlier behaviour of the customer. Especially the latter is applicable for cases where a customer might only order once a year.

The HDPTI is adapted as follows: next to the estimated probability p_i^t, the actual probability for customer i to choose time window t is denoted as q_i^t, unknown to the vendor. In all incentive calculations, the estimate p_i^t will be used. However, when the time slot is actually chosen, after the offered incentives, q_i^t will be used in the simulation of the customer behaviour and the influence of incentives. We assume the actual customer behaviour is known, next to the estimated customer behaviour. The estimated time windows a customer will choose from will therefore differ.

The behaviour of customers is in this case captured as the willingness of a customer choosing a certain time window. Incorrect customer behaviour can therefore mean two things, as two directions are possible. The first is the number of windows with a positive estimated probability actually having a positive probability, or also the amount of windows for which $p_i^t > 0$ and $q_i^t > 0$. This will be referred to as the correctness of the estimates, in a percentage, COREST. The second direction is the number of customers of which the behaviour is predicted correctly, which will be referred to as NUMCOR. These two can also be mixed, as the correctness of the expectation can be varied per customer. This leaves the term therefore open to interpretation for a certain degree of incorrectness.

4 Solution methods

4.1 The HDPTI

To be able to evaluate the influence of incentives, Campbell and Savelsbergh introduced a simulation model which will be explained in this section. The simulation activates an algorithm to calculate incentives and preliminary schedules. It starts when a customer arrives. The vendor holds a set S of preliminary schedules, created when the previous customer decided on its time window. With these schedules, the lowest insertion costs for each time window in the new schedule are determined. Now, the optimal insertion place in a delivery route can be determined and incentives, determined with one of the incentive schemes, can be offered to lure a customer in accepting a convenient time window for scheduling. Afterwards, the customer makes its purchase final, and new preliminary schedules are created. In this section, we walk through the details of the algorithm chronologically.

4.1.1 Determining insertion costs

When a new customer, suppose i, arrives, it needs to be determined where the customer should be placed in the schedule. The vendor holds a set S consisting of at most n different feasible schedules, including all orders up to the arrival of the new customer. Each of those schedules has at most m delivery routes. The optimal schedule so far is denoted as $C(*) = \min_{s \in S} C(s)$. Not all schedules are optimal considering all current orders, but a non-optimal schedule can become optimal when including a new order. To speed up the insertion heuristic, we use earliest (e_i) and latest (l_i) delivery times for each order, as recommended in Campbell and Savelsbergh (2004). These times are given by the following relationships, would order i be between $i-1$ and $i+1$ in time window t:

$$e_i = \max(e_{i-1} + h_{i-1} + t_{i-1,i}, begin^t) \text{ and} \tag{1}$$

$$l_i = \min(l_{i+1} - t_{i,i+1} - h_i, end^t). \tag{2}$$

Here, e_1 will equal the beginning time of its time window, while this will mean the end of its time window for the last delivery.

We can start the calculation of the insertion costs immediately upon arrival of customer i. The feasibility of inserting i in each time window t in all schedules $s \in S$ is evaluated by using the calculations for the earliest and latest times (Equations (1) and (2)). If the insertion is feasible, the actual cost of adding order i between $i-1$ and $i+1$, can be calculated as

$$\text{cost} = c(t_{i-1,i} + t_{i,i+1} - t_{i-1,i+1}) + C(s) - C(*). \tag{3}$$

Recall c to be the travel cost per unit. The terms $C(s)$ and $C(*)$ are necessary, because insertion in a schedule which is very expensive, could be cheaper than including it in the least cost schedule. Would we not take these costs into account, we could advance with a schedule more expensive than the least cost schedule, while including the order in the least cost schedule is still cheaper than the schedule with the cheapest insertion. Only the lowest insertion costs for each time window t and for all schedules in S is stored, denoted as C^t. Also, the point of insertion in the associated schedule s should be maintained. We use these values to calculate the incentives.

4.1.2 Effects of incentives

The incentives, I^t, are calculated based on these C^t. We will go in further detail about the incentive methods in Section 4.2. When we have the I^t, we first need to adjust all p_i^t for each customer. In Campbell and Savelsbergh (2006), the influence of an incentive is assumed to be linear. The probability of selecting a window increases by a constant rate x, the incentive impact parameter, per euro of given incentive. Thus, the new $\bar{p}_i^t = p_i^t + xI^t$. The gain in probability by the time windows receiving an incentive, needs to be compensated for in a reduction of probabilities for time windows not receiving an incentive. This total reduction in probability is equally divided among all time windows not receiving an incentive. Now, the new customer i chooses its time window depending on the new probabilities.

We must note, that if window t is infeasible due to earlier accepted orders, a problem arises would $p_i^t > 0$ for the customer. As mentioned in Section 3.2, we use two options, one where walking away is not allowed and one where it is. The first distributes the p_i^t of the infeasible window equally among all other feasible time windows where $p_i^t > 0$. The latter gives a customer the possibility to cancel the order, such that a probability to walk away will equal the sum of all p_i^t of time windows which are infeasible. However, it is possible to offer incentives to other feasible time windows to reduce the probabilities for the infeasible time windows. This way, the probability a customer will walk away will be lower.

4.1.3 Creating new schedules

When customer i chose a time window, the schedule belonging to the cheapest insertion in that time window will be changed such that customer i will fit into the new schedule. The earliest and latest delivery times of the new order are evaluated as in Equations (1) and (2). All other delivery times need to be updated dependent on the new order. A method to keep the computation time low is proposed in Section 4.4. After the updating of the schedule, this schedule will be seen as the new least cost schedule and all other $s \in S$ are removed. New schedules are constructed from scratch using the following repeated five-step approach:

i Evaluate all unrouted orders j by determining their feasibility in all time windows in all partially built schedules. Use the relationships of e_i and l_i to determine the feasibility quickly.

ii For all insertions which are feasible, e.g. $e_i \leq l_i$, evaluate the costs of the insertion as $c(t_{i-1,i} + t_{i,i+1} - t_{i-1,i+1})$. Would there be no feasible insertions, creation of this schedule is interrupted and a new schedule is created from scratch. The unfinished schedule is not compensated for.

iii From the top-k cheapest insertions, one is chosen randomly and executed.

iv Update the e_i and l_i accordingly for all orders i in the changed route. See Section 4.4 for a more detailed description.

v If all orders are inserted, add the resulting feasible schedule s to the set S coupled with its total cost $C(s)$.

Repeat this process until $n - 1$ attempts for a new schedule were made. Now, let $C(*) = \min_{s \in S} C(s)$. Now, the vendor is ready to accept a new customer.

4.2 Incentive distribution methods

The most important jobs of an incentive scheme is the decision on which time windows will receive an incentive and what the size then should be. The first decision can be made by comparing the cost of inserting a new order to each other. Then, for example the time windows which are most desirable for the vendor can be given an incentive. However, this is not as straightforward as it might seem. As shown in Campbell and Savelsbergh (2006), selecting the time windows to give incentives for a maximisation of expected profit is nontrivial. For example, one of the observations is the following: "If a single time window is considered for an incentive ($|U| = 1$) and the optimal incentive is zero, it is possible that it will receive a positive incentive when considered in conjunction with another time window." (Campbell and Savelsbergh, 2006, p. 332). Campbell and Savelsbergh argue that this might be caused by the expected profit function or the incentive limit B, as those have a quadratic nature.

Another option might be to incorporate the selection of the time windows into the optimisation model. However, this would result in the creation of a mixed integer program, which is very time consuming and thus undesirable. Therefore, the selection of time windows and the actual incentive decision is split into two parts. In Campbell and Savelsbergh (2006), five methods to determine this division are investigated, such that it can be computationally determined how well the different methods perform, both in time and expected profit, namely:

i *NOINC* (no incentives): no incentives are given, such that a benchmark exists. The selection probabilities are thus based on the initial p_i^t-values.

ii *CHPFLT* (cheapest windows get flat incentives): this method uses flat incentives, as equal incentives to l time windows with the cheapest insertion costs. These incentives are as large as possible for all time windows, without violating the probability axioms, which says probabilities need to be between zero and one and the sum of all probabilities is maintained to one. No incentives are given if all time slots have equal insertion costs, or if the number of time slots that have the possibility of receiving an incentive is not larger than the maximum number of time slots which may receive an incentive.

iii *CHPLP* (cheapest windows get LP-based incentive): the optimal incentives are chosen with a linear optimisation model, where the l time windows with the cheapest insertion costs are allowed to receive incentives. If one of the incentives is 0, the LP is solved again without that specific time window being able to receive an incentive. In our case, the earliest time window not receiving an incentive, if possible, would be put in the set V.

iv *BSTLP* (incentive for best set U based on LP): all possible combinations of time windows which may receive an incentive are enumerated, of size 1 up to l, after which the same LP as in the CHPLP is solved for each of these combinations. The set with the maximum expected profit is chosen. One must note that when considering all possible sets U, windows with a nonpositive probability to be chosen can not be part of U. Would multiple sets have the same expected profit, the first found one is chosen.

v *BST* (best case): this artificial method is used as an upper bound, as this makes sure customers always select the time window preferred by the vendor, which is the time window with the cheapest insertion costs. The wishes of the customer regarding the time slots that are available as a choice should be respected. However, no incentives are paid.

Also, the maximum number of time windows receiving an incentive, l, can be altered such that more options exist.

First, the determination of finding the cheapest time windows will be explained in more detail. In Campbell and Savelsbergh (2006), an opening is left considering this explanation, such that our interpretation might not correspond with the interpretation of Campbell and Savelsbergh. At most l windows can be added to the set of cheapest windows W. It speaks for itself that these are the windows with the lowest insertion costs C^t. If for example $l = 3$, and the third and fourth cheapest windows have the same costs, one of them will be chosen randomly. However, would we only have four windows where $p_i^t > 0$, both are not added to W. If l equals the amount of windows where $p_i^t > 0$, the CHPFLT will not offer incentives, because all possible time windows will then receive an equal incentive, making the incentives useless. However, an LP can time windows receive different incentives, such that it can be made sure a time window cheaper than the other window will be chosen. Therefore, for the CHPLP, we will include all windows.

The methods CHPLP and BSTLP base the size of the incentives on a linear program. Because the insertion costs represent the costs which will be made of inserting an order in time slot t, Campbell and Savelsbergh represent the incentive decision, for customer i, by the following quadratic optimisation problem (2006):

$$\begin{aligned} \max \quad & \sum_{t \in U}(r_i - C^t - I^t)(p_i^t + xI^t) + \sum_{t \in V}(r_i - C^t)(p_i^t - z) && && (4) \\ \text{subject to} \quad & z \leq p_i^t && \forall t \in V, && (5) \\ & \sum_{t \in U} xI^t = z|V|, && && (6) \\ & 0 \leq I^t \leq B && \forall t \in U. && (7) \end{aligned}$$

The objective function maximises the expected profit, considering the probabilities with which a time window will be chosen. The first part sums the expected profitability, including the revenue, insertion costs and given incentive of all time slots given an incentive, where the probability is adjusted according to the incentive. The second part does the same for time windows not receiving an incentive and therefore also takes a decreased probability into account. Constraints (5) make sure that the adjusted probability of a time window to be chosen will always be nonnegative, while Constraints (7) set the given incentive below the maximum possible incentive (and nonnegative). In Constraint (6), it is made sure the decrease in probability is properly divided over all time slots in V, such that the sum of the probabilities will remain one.

This is a quadratic optimisation model, which typically is time consuming and undesirable for an algorithm such as this. To make it less time consuming, the quadratic term is excluded from the summation and approximated with a piecewise linear function over $f - 1$ intervals between $I^t = 0$ and $I^t = u$, where

$$u = \min\Big(B, \frac{\min_{t \in V} p_i^t}{x}|V|\Big). \tag{8}$$

This approach of calculating u makes sure the exact maximum possible incentive will be taken as the end of the interval. Exact details on how to make an appropriate approximation of I^t can be found in Nemhauser and Wolsey (1988). However, it must be noted that f extra variables $y_1^t, \dots, y_f^t$ are needed per time window. An introduction of integer variables is unnecessary, because we are interested in the maximum, while $-(I^t)^2$ is non-increasing and convex on the interval. Therefore, the optimisation problem to be solved in the algorithm will result in the following LP relaxation:

$$\begin{aligned} \max \quad & \sum_{t \in U}(x(r_i - C^t) - p_i^t)I^t \; - \\ & \sum_{t \in U} x\Big(\Big(\frac{u}{f-1}\Big)^2 y_2^t + \Big(\frac{2u}{f-1}\Big)^2 y_3^t + \ldots + (u)^2 y_f^t\Big) - \sum_{t \in V}(r_i - C^t)z \end{aligned} \tag{9}$$

$$\text{subject to} \quad \sum_{i=1}^{f} y_i^t = 1 \qquad \forall t \in U, \tag{10}$$

$$I^t = \frac{u}{f-1} y_2^t + \frac{2u}{f-1} y_3^t + \ldots + u y_f^t \qquad \forall t \in U, \tag{11}$$

$$z \leq p_i^t \qquad \forall t \in V, \tag{12}$$

$$\sum_{t \in U} x I^t = z|V|, \tag{13}$$

$$0 \leq I^t \leq B \qquad \forall t \in U. \tag{14}$$

Here, Constraints (10) and (11) are added to approximate I^t.

We also consider the option of customers walking away. When considering this variant the sets U and V are different, while the set F is introduced. Now, F consists of all time windows with positive probability to be chosen, where no feasible insertion exists, thus the schedule is full timewise. U and V are split up as before. The objective function will remain equal, however, with the different set V. Constraints (12) and (13) will differ, as those restrictions still hold for all time windows, and will be changed into the following constraints:

$$z \leq p_i^t \qquad \forall t \in V,\, F, \tag{15}$$

$$\sum_{t \in U} x I^t = z(|V| + |F|). \tag{16}$$

Furthermore, the calculation of variable u changes. Because elements of V now need to be put in F, the size of the set V changes, while the maximum possible incentive must remain equal to before allowing customers to walk away. Thus, u will be calculated as

$$u = \min\Big(B, \frac{\min_{t \in V} p_i^t}{x}(|V| + |F|)\Big). \tag{17}$$

4.3 Incorrect customer behaviour estimates

To be able to compare the results when varying the correctness of the estimates, it is useful to make sure the real behaviour of each customer is kept constant. That way, the decisions of each customer could be equal and the NOINC and BST will remain equal throughout all runs. Now, the results can be compared truly.

4.3.1 Different handling of probability changes

When incentives are given, the probabilities of choosing a time window need to be adjusted. However, in the case of incorrect estimates of those probabilities, the real probabilities need to be adjusted. In this paper, the influence of incentives is tried to be kept as close as possible to the linear setup defined by Campbell and Savelsbergh (2006), such that the results can be compared well. This is straightforward. First, incentives could be offered to time windows which will never be chosen. Second, incentives could cause (the sum of all) real probabilities to end up larger than one. The first one is easy to solve by checking whether the time window actually is possible according to the actual preferences of the customer. The latter is more complicated.

This problem is solved as follows. First, increase the probability of the time windows receiving an incentive for which holds $q_i^t > 0$. Afterwards, let $s = \sum_{t \in T: I^t > 0} q_i^t$ be the sum of the just created new probabilities. Three different scenarios arise; $s < 1$, $s = 1$, $s > 1$.

The easiest case is $s = 1$. Then, all q_i^t for which $I^t = 0$ are set to 0. As the idea of the linear set up is to increase each q_i^t with xI^t, it would not be in line with the linear influence of incentives to still consider windows for which $I^t = 0$. In the case of $s > 1$, the influence of the incentives was too large. We also set q_i^t to 0 for all t for which $I^t = 0$, but we need to decrease the increased q_i^t. We do this proportionally to s, thus divide each q_i^t by s. Then, s will become one.

The last case is $s > 1$. Then, the increase in probability needs to be compensated for with windows for which $q_i^t > 0$ for which $I^t = 0$. The decrease in those windows is denoted as $d_{\text{total}} = \sum_{t \in T: I^t > 0, q_i^t > 0} xI^t$. We distribute d_{total} equally among these t. Because the incentives are calculated with the incorrect p_i^t, it could be that the decrease causes one of the q_i^t end up negative. To prevent this, we decrease all q_i^t in steps, where each step consists of decreasing with the lowest remaining q_i^t which needs to be decreased, resulting in a total decrease d_{step}. After each step, d_{total} is decreased with this d_{step}. We keep doing this as long as $d_{\text{total}} > 0$ and there exists q_i^t which can be decreased. At the end, all probabilities q_i^t are scaled proportionally to one, if necessary.

For this approach is chosen because, it does not conflict with the linear setup of the impact of incentives. Including the difference to zero in the probability to walk away (if this is considered) is also not desirable, as giving incentives does not suddenly result in someone having a stronger desire to walk away.

4.4 Methods to speed up insertion heuristic

Evaluating all insertion possibilities of a new order in a certain time window in all possible places can be quite an exhaustive search for an instance with thousands of customers. As this is done many times during the algorithm, also when creating the new schedules, speeding up this part can result in a significant smaller running time, especially for instances with many customers. The process of the feasibility of inserting order i in time window t, between $i-1$ (with window t_{i-1}) and $i+1$ (with window t_{i+1}), at place p can be checked efficiently. Do this by iterating over all windows t, but only evaluate the places where that time window could be inserted, thus where $t_{i-1} \leq t_i$ and $t_i \leq t_{i+1}$. Start with $p = 1$, after which p should be incremented if $t_i = t_{i+1}$, while t_i is increment if $t_i < t_{i+1}$. Continue until the insertion of the last time window right before returning to the depot is evaluated.

Furthermore, every time a customer is added in a route, the earliest and latest delivery times need to be updated. This can be quite time consuming. Therefore, as explained as Campbell and Savelsbergh (2004), considering the latest delivery times, one only needs to update those preceding the added order, while only the succeeding orders need to get a new earliest delivery time. Thus, when adding order i in a route with size n,

$$e_{k,\text{new}} = \max(e_{k,\text{old}}, e_{k-1} + h_{k-1} + t_{k-1,k}) \text{ for } k = i+1 \text{ up to } n \text{ and} \tag{18}$$

$$l_{k,\text{new}} = \min(l_{k,\text{old}}, l_{k+1} - t_{k,k+1} - h_k) \text{ for } k = i-1 \text{ down to } 0. \tag{19}$$

We can stop this process prematurely if $e_{k,\text{new}} = e_{k,\text{old}}$, as all times of orders succeeding k will not change. This also holds for the latest delivery times.

5 Computational experiments

To compute the results, 25 data instances are simulated. Since the original data instances used in Campbell and Savelsbergh (2006) were unavailable, new ones were generated. These will make sure it is possible to compare the incentive methods, customer behaviour patterns and the maximum number of time windows which may receive an incentive. Unless otherwise stated, each instance contains 30 customers with a service time of 20 minutes. Each customers has coordinates generated uniformly in an artificial area of 60 by 60 units, a revenue of €100 and an estimation of the time windows that each customer might choose, including a preferred window. Each customer can get a maximum incentive of €5, while the influence per euro of incentive is 0.2.

Regarding the preferences, three customer behaviour patterns (PROBPAT) are used. They all assume customers have a preference for eight consecutive windows, where the vendor delivers in twelve one hour time windows. A combination of the beginning and the end of the day is also possible. As a basis, all time windows have an equal probability, which is also the first pattern. In the second, the probability of choosing one of the time windows is twice as large as the others, while this is thrice as large for the third pattern. The window with the larger probability is chosen randomly. We use an Euclidian distance for calculating the travel times, where one unit takes one minute. Furthermore, a travel cost of €1 per unit is used. Only one vehicle is used for the simulation, where it is assumed that the capacity needed for all orders will not exceed the vehicle capacity, for computation purposes.

Lastly, when creating new schedules, a list of cheapest insertions is use. Because all schedules are created with some randomisation, the schedules in S could be very different. However, the costs are quite equal if k is kept small, approximately three. Therefore, we use $k = 3$. Next to this, five linear pieces are used for the approximation in the LP is done, such that $f = 6$.

All results were run on a laptop using Windows 10, an Intel(R) Core(TM) i5-4210U 1.7GHz CPU with a 2.4GHz Turbo Boost and 4GB RAM. The algorithm was created in Java version 7.0.1 in combination with Gurobi 7.0.2.

5.1 The HDPTI

First, a base scenario is evaluated. The maximum number of windows which may receive an incentive, referred to as $|U|$, is alternated from one to four to evaluate the differences in the amount of windows a customer might receive an incentive. Also, all incentive methods introduced in Section 4.2 are considered. Information containing the average over all $|U|$ for each method is added to compare the different methods, including the improvement made relative to the basic method where no incentives are offered. Below the improvements, the results in Campbell and Savelsbergh (2006) are displayed for comparison of our average results. Furthermore, the base scenario is not only evaluated for the situation of not allowing walkways, but also when walkaway is allowed. These results can be found in Table 1.

Table 1: Base results

		Walkaway **is not** allowed					Walkaway **is** allowed				
PROBPAT	|U|	NOINC	CHPFLT	CHPLP	BSTLP	BST	NOINC	CHPFLT	CHPLP	BSTLP	BST
1	1	1212.43	1361.69	1347.24	1338.39	1427.93	1069.25	1331.47	1331.60	1296.21	1427.93
	2	1212.43	1357.85	1360.75	1354.96	1427.93	1069.25	1287.17	1364.06	1345.29	1427.93
	3	1212.43	1263.91	1376.18	1367.55	1427.93	1069.25	1238.03	1386.92	1348.41	1427.93
	4	1212.43	1274.75	1332.92	1377.87	1427.93	1069.25	1194.56	1333.99	1351.31	1427.93
Average		1212.43	1314.55	1354.27	1359.69	1427.93	1069.25	1262.81	1354.14	1335.31	1427.93
% impr.		0.00	8.42	11.70	12.15	17.77	0.00	18.10	26.64	24.88	33.55
(C&S 2006)		(0.00)	(6.11)	(13.66)	(14.42)	(10.77)	(0.00)	(21.44)	(36.31)	(35.13)	(31.30)
2	1	1187.80	1327.51	1350.57	1345.00	1427.93	1094.57	1341.97	1347.00	1335.29	1427.93
	2	1187.80	1301.51	1373.98	1376.53	1427.93	1094.57	1280.86	1320.42	1354.02	1427.93
	3	1187.80	1258.62	1385.67	1393.53	1427.93	1094.57	1219.40	1365.79	1351.07	1427.93
	4	1187.80	1250.62	1361.20	1435.40	1427.93	1094.57	1152.15	1346.65	1341.91	1427.93
Average		1187.80	1284.56	1367.85	1387.61	1427.93	1094.57	1248.60	1344.96	1345.57	1427.93
% impr.		0.00	8.15	15.16	16.82	20.22	0.00	14.07	22.88	22.93	30.46
(C&S 2006)		(0.00)	(6.21)	(12.07)	(11.74)	(9.20)	(0.00)	(14.52)	(27.66)	(27.15)	(24.77)
3	1	1177.15	1311.24	1309.23	1299.39	1427.93	1033.40	1344.75	1374.87	1317.01	1427.93
	2	1177.15	1232.67	1292.47	1318.93	1427.93	1033.40	1283.00	1318.64	1327.46	1427.93
	3	1177.15	1205.90	1304.72	1299.60	1427.93	1033.40	1208.19	1337.74	1326.94	1427.93
	4	1177.15	1229.00	1299.98	1312.18	1427.93	1033.40	1151.41	1356.40	1356.81	1427.93
Average		1177.15	1244.70	1301.60	1307.52	1427.93	1033.40	1246.84	1346.91	1332.05	1427.93
% impr.		0.00	5.74	10.57	12.77	21.30	0.00	20.65	30.34	28.90	38.18
(C&S 2006)		(0.00)	(7.90)	(11.24)	(11.74)	(9.79)	(0.00)	(14.50)	(23.68)	(25.28)	(24.23)

The improvements as in Campbell and Savelsbergh (2006) are displayed below our results.

Immediately, one can see the benefit of offering incentives. In all situations, the profit has improved compared to not giving incentives. When walkaway is not allowed, the average extra profit is approximately 6 to 8% for the simple incentive scheme, while the improvement made from solving the incentives as a linear optimisation model lies between 11 and 17% for both the CHPLP and BSTLP, depending on the PROBPAT. It shows us we that using an LP can improve our profit up to 2 times as much as a flat incentive method. Furthermore, the best case scenario shows us that an average improvement up to 21% can be made. When walkaway is allowed, the improvements are much higher, because the NOINC method performs much worse, while the offering of incentives almost makes up for the loss in customers because we allow walkaways. This already points out the importance of preventing customers to walk away.

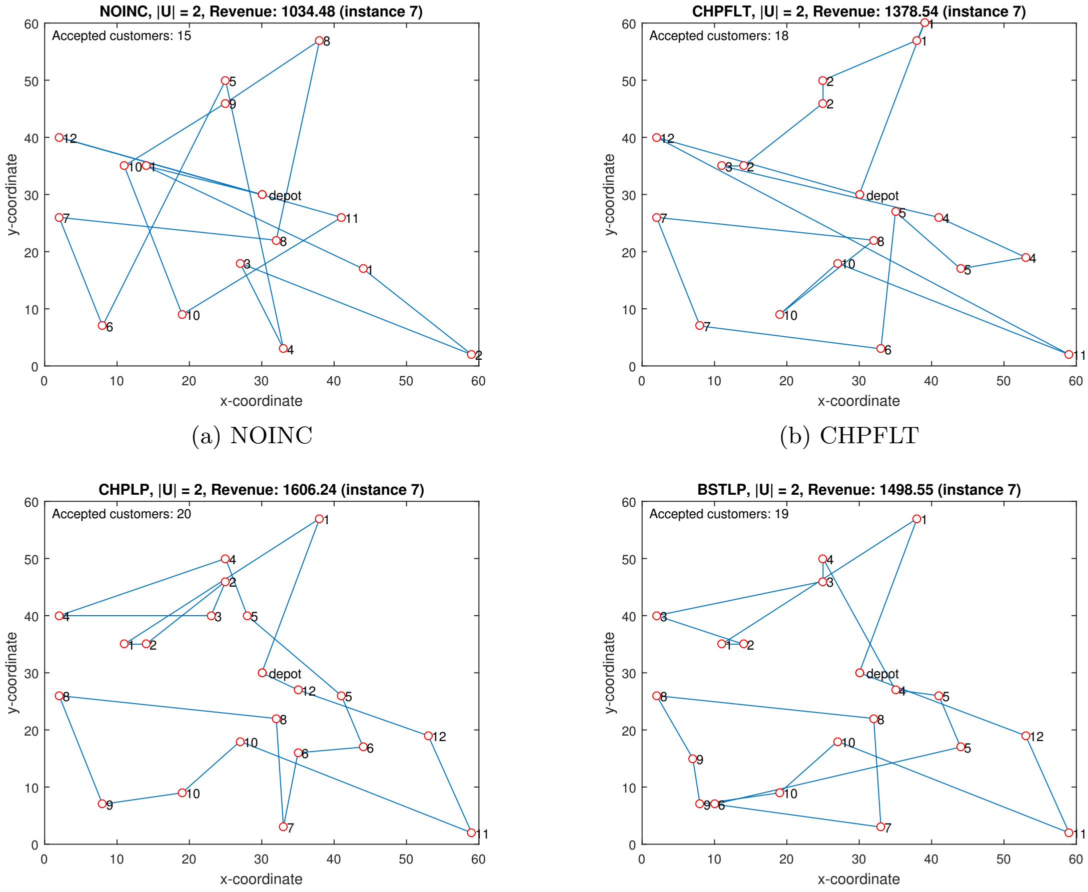
The pattern that more complicated incentive methods result in a higher improvement corresponds to the results found in Campbell and Savelsbergh (2006), while the improvements found differ from those results. For PROBPAT 1, the methods perform worse, while for PROBPAT 3, we do better in most cases. The improvement are influenced largely by the NOINC, being a cause of the different results. As the 25 instances are (which are not the same as in Campbell and Savelsbergh) completely randomised, while we also use randomisation in the algorithms, other seeds resulted in different results. This is also partly because including an extra order results in a major extra revenue. Therefore, these differences are unmeasurable. will thus result in different results. Lastly, our best case scenario outperforms all incentive method, which one would expect, in contrast to the results in Campbell and Savelsbergh. Again, this could be caused by randomisation, but it is impossible to draw a conclusion because of the different data instances.

The amount of windows eligible for incentives, differ per PROBPAT and whether walking away is allowed or not. For the CHPFLT, we see in all patterns that offering incentives to less windows is rewarding. Compared to Campbell and Savelsbergh (2006), this was only the case for PROBPAT 1 and 2, as PROBPAT 3 showed the better performance would $|U|$ be 2 or 3. For both allowing and not allowing walkaways, PROBPAT 3 has a different result in Campbell and Savelsbergh. Since this seems to be consistent, this difference might be caused because of the undefined size of the flat incentives given in the methods by Campbell and Savelsbergh.

When we look at the LP-based methods, we see a consistent rise of the profit when incentives are offered to more different time slots. Overall, the CHPLP gives us an optimal $|U|$ of 3, while the BSTLP performs better when $|U| = 4$. As the latter method checks all possible combinations of eligible time windows and thus results in more options, this makes sense. We do see, for both allowing and not allowing walkaways, a difference for the third probability pattern. It seems as $|U|$ is less important, as the results are quite mixed, such that all $|U|$ perform well at least once in the four evaluated options. Next to this, the profit drops. As this was also the case for the CHPFLT, this might be caused by the probability pattern, because it is harder to exclude an expensive window if a customer has a strong preference for that window, which is the case for PROBPAT 3.

To take a closer look at the actual result of the selection of time windows by offering incentives, the resulting schedule of one instance is given for all methods, excluding the best case scenario. As the incentives not always have the desired influence, a more extreme instance is chosen. In Figure 1, the different resulting schedules are displayed, while the revenue and number of accepted customers are also included.

Notice the difference in chaotic and clean division of time windows over the coordinate grid. While in Figure 1a, the NOINC has a delivery route going from one side to another, the schedules become more advanced with every method, as the incentive offering methods have more customers close to each other with equal or consecutive time windows. Also, visible is the better division for the CHPLP and BSTLP in comparison with the CHPFLT. Not only will this reduce the delivery costs, more customers are accepted, as the CHPLP accepts 20 customers, while the NOINC can only handle 15 customers.



(a) NOINC (b) CHPFLT (c) CHPLP (d) BSTLP

Figure 1: Example of the resulting schedules of an instance, split by method

As it is likely the NOINC has to decline customers because of the schedule already in the early stages, before the schedule is full, it is also interesting to see what happens if the number of customers is downgraded to, in this case 15. At that moment, every customer has a significant impact on the resulting profit, as there are not enough customers left to compensate for any unserved customers. These results can be found in Table 2. Campbell and Savelsbergh did not include the option of allowing walkaway, such that no improvement is noted.

When walkaway is not allowed, notice the lower profit in all cases compared to the profit found in Table 1, where the BST-method even drops to a profit improvement of approximately 13%. While the profit of the three incentive methods also drops, the earlier build up of the more sophisticated incentive schemes to work better is still present. One result stands out, as the CHPFLT in PROBPAT 3 has an average approximately equal to the NOINC, while this is not the case in Campbell and Savelsbergh (2006). This might be caused by the instances, as the LP-based methods also show a lower improvement in the profit. As changing the size of the incentives, the undefined size in Campbell and Savelsbergh is not the cause.

In the case where walkaway is allowed, the profit does improve. This is due to the very poor performance of the NOINC method, which shows how well incentives are able to prevent customers from abandoning their order. The difference in scheduling costs not only points out an improvement in profit, but also the acceptance of approximately 3 or 4 customers extra for the CHPFLT. When taking a look at the LP-based methods, we can even conclude that they are able to let on average only one out of fifteen customers run away, as the best case scenario consists of schedules including all customers. Therefore, it can be said that even if it is not possible to completely fill a schedule, incentives will have a significant impact on the revenue.

Table 2: Base results with less orders

		Walkaway **is not** allowed					Walkaway **is** allowed				
PROBPAT	|U|	NOINC	CHPFLT	CHPLP	BSTLP	BST	NOINC	CHPFLT	CHPLP	BSTLP	BST
1	1	1029.21	1091.52	1093.20	1086.17	1156.49	800.43	1077.76	1076.09	1043.28	1156.49
	2	1029.21	1084.97	1111.53	1114.12	1156.49	800.43	1074.39	1099.02	1101.26	1156.49
	3	1029.21	1061.01	1105.22	1124.23	1156.49	800.43	1036.21	1095.56	1089.46	1156.49
	4	1029.21	1076.34	1088.68	1115.49	1156.49	800.43	951.35	1083.16	1089.13	1156.49
Average		1029.21	1078.46	1099.66	1110.00	1156.49	800.43	1034.93	1088.46	1080.78	1156.49
% impr.		0.00	4.79	6.85	7.85	12.37	0.00	29.30	35.98	35.02	44.48
(C&S 2006)		(0.00)	(3.98)	(7.97)	(8.21)	(10.43)	(–)	(–)	(–)	(–)	(–)
2	1	1019.37	1081.03	1092.03	1100.80	1156.49	801.04	1064.65	1065.02	1073.22	1156.49
	2	1019.37	1083.48	1102.31	1100.36	1156.49	801.04	1050.89	1081.76	1081.13	1156.49
	3	1019.37	1067.80	1108.82	1118.94	1156.49	801.04	988.57	1086.83	1075.99	1156.49
	4	1019.37	1050.40	1110.91	1123.60	1156.49	801.04	930.55	1090.42	1079.24	1156.49
Average		1019.37	1070.68	1103.52	1110.92	1156.49	801.04	1008.67	1081.01	1077.40	1156.49
% impr.		0.00	5.03	8.26	8.98	13.45	0.00	25.92	34.95	34.50	44.37
(C&S 2006)		(0.00)	(4.02)	(6.34)	(6.34)	(9.18)	(–)	(–)	(–)	(–)	(–)
3	1	1032.44	1058.77	1073.99	1081.54	1156.49	764.78	1033,47	1030,14	1024,61	1156,49
	2	1032.44	1045.02	1086.39	1093.97	1156.49	764.78	1019,99	1012,89	1001,45	1156,49
	3	1032.44	1017.19	1085.56	1094.83	1156.49	764.78	969,55	1023,15	1016,36	1156,49
	4	1032.44	1023.41	1066.53	1082.89	1156.49	764.78	924,80	1033,77	1024,90	1156,49
Average		1032.44	1036.10	1078.12	1088.31	1156.49	764.78	986,95	1024,99	1016,83	1156,49
% impr.		0.00	0.35	4.42	5.41	12.02	0.00	29.05	34.02	32.96	51.22
(C&S 2006)		(0.00)	(4.19)	(5.93)	(7.75)	(9.24)	(–)	(–)	(–)	(–)	(–)

The improvements as in Campbell and Savelsbergh (2006) are displayed below our results. If no improvement is referred to, that experiment was not executed.

So far, we have only looked at the base model, while the results also depend on the maximum size of the incentives and the exact impact of the incentives. Therefore, in Tables 3 and 4, we vary the upper limit on the incentives, B, and the actual impact of the incentives on the probabilities, x, respectively. Also, instead of considering all $|U|$, only the results of $|U| = 2$ are shown, as those show a large improvement in profit already. This makes the analysis of the results easier. Lastly, 30 customers are used again.

Table 3: Base results for varying incentive upper limit

		Walkaway **is not** allowed					Walkaway **is** allowed				
B	PROBPAT	NOINC	CHPFLT	CHPLP	BSTLP	BST	NOINC	CHPFLT	CHPLP	BSTLP	BST
1	1	1212.43	1316.35	1311.81	1316.20	1427.93	1069.25	1217.64	1251.25	1267.22	1427.93
	2	1187.80	1265.56	1299.36	1259.94	1427.93	1094.57	1235.76	1264.47	1267.67	1427.93
	3	1177.15	1185.59	1200.45	1236.38	1427.93	1033.40	1267.62	1291.49	1294.49	1427.93
Average		1192.46	1255.83	1270.54	1270.84	1427.93	1065.74	1240.34	1269.07	1276.46	1427.93
% impr.		0.00	5.31	6.55	6.57	19.75	0.00	16.38	19.08	19.77	33.98
(C&S 2006)		(–)	(–)	(–)	(–)	(–)	(–)	(–)	(–)	(–)	(–)
2	1	1212.43	1357.85	1375.89	1287.59	1427.93	1069.25	1287.17	1315.46	1288.17	1427.93
	2	1187.80	1301.51	1327.32	1353.85	1427.93	1094.57	1280.86	1313.13	1316.21	1427.93
	3	1177.15	1232.67	1271.47	1279.06	1427.93	1033.40	1283.00	1325.06	1288.69	1427.93
Average		1192.46	1297.34	1318.89	1306.83	1427.93	1065.74	1283.68	1317.88	1297.69	1427.93
% impr.		0.00	8.80	10.60	9.59	19.75	0.00	20.45	23.66	21.76	33.98
(C&S 2006)		(0.00)	(8.57)	(10.82)	(12.27)	(9.72)	(–)	(–)	(–)	(–)	(–)
5	1	1212.43	1357.85	1360.75	1354.96	1427.93	1069.25	1287.17	1364.06	1345.29	1427.93
	2	1187.80	1301.51	1373.98	1376.53	1427.93	1094.57	1280.86	1320.42	1354.02	1427.93
	3	1177.15	1232.67	1292.47	1318.93	1427.93	1033.40	1283.00	1318.64	1327.46	1427.93
Average		1192.46	1297.34	1342.40	1350.14	1427.93	1065.74	1283.68	1334.37	1342.26	1427.93
% impr.		0.00	8.80	12.57	13.22	19.75	0.00	20.45	25.22	25.95	33.98
(C&S 2006)		(0.00)	(8.57)	(13.02)	(13.47)	(9.72)	(–)	(–)	(–)	(–)	(–)

The results are computed with $|U| = 2$. The improvements as in Campbell and Savelsbergh (2006) are displayed below our results. If no improvement is referred to, that experiment was not executed.

We look at the results where $B = 1$, $B = 2$ and $B = 5$ (the natural limit because $x = 0.2$)

such that an actual difference can be detected. For example, $B = 3.5$ gives more or less identical results to $B = 5$, such that we know that incentives above $B = 3.5$ either have no large impact or occur rarely. An interesting pattern occurs immediately, as we see that the CHPFLT gives smaller incentives, since for $B = 2$ and higher, the results are actually equal. However, limiting the incentives at 1 does result in a lower profit. This means that when giving flat incentives if the upper limit is 2 or higher, the upper bound following from the probabilities are more restricting than the actual upper limit. Therefore we can conclude that giving higher incentives result in a higher profit, also for the CHPFLT. This is the case for both allowing and not allowing walkaway.

For both LP-based methods, the profit actually increases if a higher incentive is possible. This is most likely due to the LP choosing the optimal size of the incentives. If higher incentives are possible, the profit can only improve compared to the case where lower incentives are allowed. This is present in the results in Campbell and Savelsbergh as well. Notable is that the improvements while restricting the upper limit still are very significant, especially in the case where walkaway is allowed, as those averages equal the averages where walkaway is not allowed. This points out the fact that offering small incentives already has a large impact on the profit by preventing customers from walking away. Because when the incentives have a maximum of €1, the maximum increase in the probability of choosing a time window is equal to x, 0.2 in this case. Thus it can also be concluded that even if a vendor is often not able to fully exclude the most costly time windows, offering incentives will still contribute to a higher profit.

Table 4: Base results for varying incentive impact levels

		Walkaway **is not** allowed					Walkaway **is** allowed				
x	PROBPAT	NOINC	CHPFLT	CHPLP	BSTLP	BST	NOINC	CHPFLT	CHPLP	BSTLP	BST
0.1	1	1212.43	1341.33	1361.29	1300.95	1427.93	1069.25	1260.17	1313.60	1325.28	1427.93
	2	1187.80	1287.51	1348.97	1394.75	1427.93	1094.57	1258.66	1298.97	1277.17	1427.93
	3	1177.15	1221.03	1267.92	1303.82	1427.93	1033.40	1263.26	1268.13	1243.45	1427.93
Average		1192.46	1283.29	1326.06	1333.17	1427.93	1065.74	1260.70	1293.57	1281.97	1427.93
% impr.		0.00	7.62	11.20	11.80	19.75	0.00	18.29	21.38	20.29	33.98
(C&S 2006)		(0.00)	(7.40)	(7.71)	(7.94)	(9.72)	(–)	(–)	(–)	(–)	(–)
0.2	1	1212.43	1357.85	1360.75	1354.96	1427.93	1069.25	1287.17	1364.06	1345.29	1427.93
	2	1187.80	1301.51	1373.98	1376.53	1427.93	1094.57	1280.86	1320.42	1354.02	1427.93
	3	1177.15	1232.67	1292.47	1318.93	1427.93	1033.40	1283.00	1318.64	1327.46	1427.93
Average		1192.46	1297.34	1342.40	1350.14	1427.93	1065.74	1283.68	1334.37	1342.26	1427.93
% impr.		0.00	8.80	12.57	13.22	19.75	0.00	20.45	25.22	25.95	33.98
(C&S 2006)		(0.00)	(8.57)	(13.02)	(13.47)	(9.72)	(–)	(–)	(–)	(–)	(–)
0.3	1	1212.43	1363.36	1351.81	1381.89	1427.93	1069.25	1296.17	1373.01	1391.85	1427.93
	2	1187.80	1306.18	1377.59	1383.12	1427.93	1094.57	1288.26	1320.70	1362.70	1427.93
	3	1177.15	1236.55	1297.54	1324.18	1427.93	1033.40	1289.58	1296.41	1324.02	1427.93
Average		1192.46	1302.03	1342.31	1363.06	1427.93	1065.74	1291.34	1330.04	1359.56	1427.93
% impr.		0.00	9.19	12.57	14.31	19.75	0.00	21.17	24.80	27.57	33.98
(C&S 2006)		(–)	(–)	(–)	(–)	(–)	(–)	(–)	(–)	(–)	(–)

The results are computed with $|U| = 2$. The improvements as in Campbell and Savelsbergh (2006) are displayed below our results. If no improvement is referred to, that experiment was not executed.

Varying the impact of the incentives also leads to a clear pattern. Even though the LP-based methods still do better than the CHPFLT, a small decline in the profits is present when incentives are given a lower impact. This is most likely due to not being able to fully exclude options for time windows, as the incentive upper limit comes into play. On the other hand, if we increase x, the profit is only slightly higher. Probably, the scheduling can not be made more optimal easily, as time windows in this format can already be excluded if $x = 0.2$ without needing to cross the upper limit B of 5. Therefore, it is difficult to form a conclusion, as the way the incentive impact is set up, linearly, create this result where incentives will definitively have an impact. The still very significant improvement when $x = 0.1$ shows us, just as when the upper limit on incentives is small, that also if a vendor is not able to fully exclude time windows, incentives, small or large, still have a large impact on the profit.

In the results so far, customers are not very restrictive about their delivery windows, as customers could accept one out of eight consecutive delivery time windows. It might be interesting to also consider shorter patterns. Therefore, in Table 5, results are computed with four acceptable delivery windows.

Table 5: Base results with four acceptable delivery windows

		Walkaway **is not** allowed					Walkaway **is** allowed				
PROBPAT	|U|	NOINC	CHPFLT	CHPLP	BSTLP	BST	NOINC	CHPFLT	CHPLP	BSTLP	BST
1	1	1152.34	1208.02	1214.90	1227.58	1238.35	1078.43	1134.08	1147.33	1135.57	1238.35
	2	1152.34	1200.16	1206.62	1238.81	1238.35	1078.43	1134.56	1156.21	1177.98	1238.35
	3	1152.34	1166.09	1204.62	1236.58	1238.35	1078.43	1117.94	1154.14	1154.44	1238.35
	4	1152.34	1152.34	1202.97	1236.58	1238.35	1078.43	1078.43	1141.18	1154.44	1238.35
Average		1152.34	1181.65	1207.28	1234.89	1238.35	1078.43	1116.26	1149.71	1155.61	1238.35
% impr.		0.00	2.54	4.77	7.16	7.46	0.00	3.51	6.61	7.16	14.83
(C&S 2006)		(0.00)	(−0.44)	(6.67)	(4.19)	(7.70)	(–)	(–)	(–)	(–)	(–)
2	1	1113.78	1201.44	1243.02	1240.71	1238.35	1073.21	1180.51	1179.40	1191.34	1238.35
	2	1113.78	1159.65	1230.08	1235.14	1238.35	1073.21	1106.70	1180.85	1183.14	1238.35
	3	1113.78	1120.56	1221.67	1215.82	1238.35	1073.21	1084.02	1204.40	1200.24	1238.35
	4	1113.78	1113.78	1220.42	1215.82	1238.35	1073.21	1073.21	1190.67	1200.24	1238.35
Average		1113.78	1148.86	1228.80	1226.87	1238.35	1073.21	1111.11	1188.83	1193.74	1238.35
% impr.		0.00	3.15	10.33	10.15	11.18	0.00	3.53	10.77	11.23	15.39
(C&S 2006)		(0.00)	(−1.21)	(8.17)	(6.50)	(7.71)	(–)	(–)	(–)	(–)	(–)
3	1	1098.85	1164.21	1213.16	1218.79	1238.35	1054.74	1161.86	1180.90	1140.15	1238.35
	2	1098.85	1147.69	1198.22	1214.27	1238.35	1054.74	1112.53	1192.64	1199.27	1238.35
	3	1098.85	1093.98	1193.77	1203.02	1238.35	1054.74	1078.96	1190.86	1213.35	1238.35
	4	1098.85	1098.85	1203.31	1203.02	1238.35	1054.74	1054.74	1191.30	1213.35	1238.35
Average		1098.85	1126.18	1202.12	1209.77	1238.35	1054.74	1102.02	1188.92	1191.53	1238.35
% impr.		0.00	2.49	9.40	10.09	12.70	0.00	4.48	12.72	12.97	17.41
(C&S 2006)		(0.00)	(−0.25)	(5.96)	(5.92)	(7.14)	(–)	(–)	(–)	(–)	(–)

The improvements as in Campbell and Savelsbergh (2006) are displayed below our results. If no improvement is referred to, that experiment was not executed.

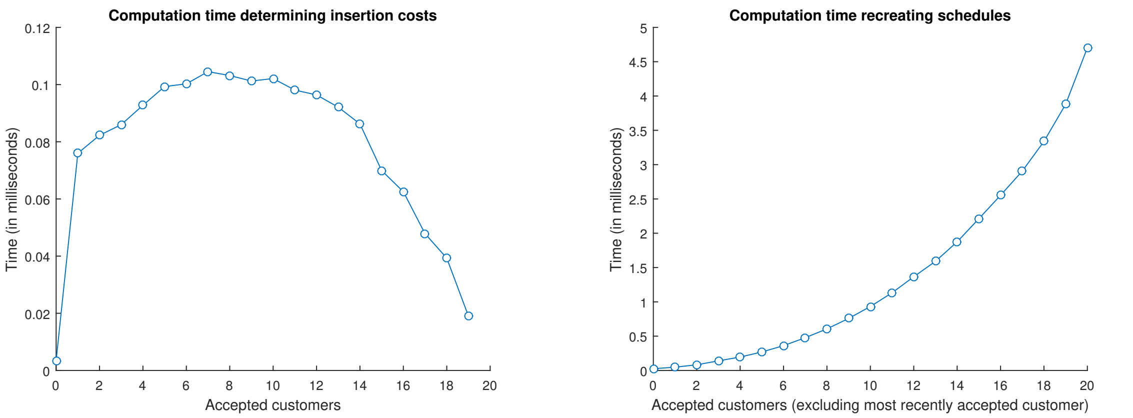
Notice the immediate drop in the profit for all methods for both not allowing and allowing walkaway. This makes sense, as there is less of a choice in time windows, such that the probability to choose the optimal time window (considering all twelve time windows) decreases and an optimal schedule is out of reach. The results of the CHPFLT stand out as well. Not only is offering incentives to less time windows beneficial, the case where $|U| = 4$ shows the exact same results as the NOINC. This points out the fact that incentives will not be offered if there are as many options for time windows as time windows receiving an incentive. Our results regarding this case are not in line with those in Campbell and Savelsbergh (2006), because there a decrease in profit is noticeable. A cause could be the actual offering of incentives if the insertion costs are unequal, as it is not stated in Campbell and Savelsbergh what happens if the insertion costs are unequal, but incentives would be offered to all time windows which might be chosen.

When we consider the CHPLP and BSTLP, it is very interesting to see that when walkaway is allowed, the improvements are only slightly higher than when walkaway is not allowed. Furthermore, $|U|$ does not seem to be of influence, as in most cases the results are quite similar, except for the BSTLP with $|U| = 1$ in PROBPAT 3, where walkaway is allowed. Also, the BSTLP method in the case where walkaway is not allowed almost equals the BST. This points out the major improvement offering incentives will make to the profit. Even if customers have a smaller selection window, offering incentives could increase the profit. However, the improvement is most present for the CHPLP and BSTLP.

5.2 Computation times

It is interesting to know how long the methods will take, as slow calculations might lead to less satisfied customers. Plus, customers could arrive shortly after each other. It should be prevented that customers will be put at the same place in the schedule, while there is no time for both of them, or that incentives are calculated based on a schedule excluding a recently placed order. As more customers lead to more possible insertion places in a schedule, the running time is not equal at all times. This causes us to present the computation times as a function of the amount of so far accepted customers. All computation times are averaged over the three probability patterns. Also, before measuring the actual computation times, all instances are run over the first three methods and all four sizes of U once, to activate the turbo boost.

First, we report the computation times of calculating the insertion costs and the computation time of recreating all schedules, as those do not differ per method. The latter also includes the updating of the earliest and latest delivery times in the new least cost schedule. The results are displayed in Figure 2. Both procedures have a clear pattern. The computation time of determining the insertion costs is a parabola, as the computation time gets larger when more customers are accepted, until the schedule is approximately half-full (around 8 customers). Afterwards, the running time decreases significantly. This pattern arises because the number of possible new insertions grow in the beginning due to rising available insertion places, while after a while the schedule becomes full and no insertions can be made in some places in the schedule. The computation time is in this case not yet a tenth of a millisecond, too small to notice unless we would have a lot of schedules.



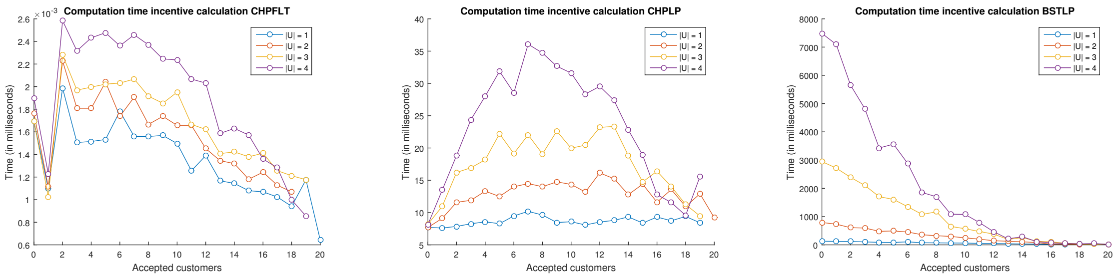
(a) Computation times of the insertion costs.

(b) Computation times of recreating all schedules.

Figure 2: Computation times procedures for incentive calculation.

The computation time of recreating the schedules is exponential. This is also as expected, as every extra to be included customer leads to an extra customer of whom the insertion costs in the new schedule needs to be checked. In this case, the recreation time is still in milliseconds and is not very significant. However, as the computation time is exponential, this could quickly rise above a second. Therefore, would more customers be present who can arrive shortly after each other, another method of schedule recreation could be considered.

In Figure 3, the computation times of the incentives can be found. While the CHPFLT only takes approximately 2 microseconds for all $|U|$, this is not the case for the LP-based methods. Regarding the CHPLP, the running time also is parabolic for $|U| = 4$, where it takes the most time halfway filling the schedule. The other sizes remain approximately constant, where a smaller size results in a faster running time, where $|U| = 1$ is approximately 3 to 5 times faster than $|U| = 4$ with 8 milliseconds. The BSTLP however, takes much longer. Already for these small instances, only $|U| = 1$ is acceptable, as $|U| = 2$ already takes almost a second for the first customers. Because the results of the CHPLP and BSTLP are quite similar, it might not be worth it to have such a long computation time, especially if only one time window can be given an incentive. A solution might be to already make a selection of the to be considered combinations before starting to calculate the expected profit of all of them.



(a) CHFLT (b) CHPLP (c) BSTLP

Figure 3: Computation times of incentive calculation, split up per method and $|U|$.

5.3 Incorrect customer behaviour estimates

In the previous subsection, the major impact incentives can have was already displayed. However, we do still not know the consequences of incorrect customer behaviour estimates. For now, we are only interested in the influence of the degree of incorrectness on the profit, such that we evaluate the base case as in Section 5.1, only now with incorrect customer behaviour. We want to be able to include the option of a completely wrong estimate (for example, the vendor argues customer i chooses from windows one to six, while i actually chooses from windows seven to twelve), such that we decrease the length of the probability patterns to six. Lastly, since the pattern of the averages found in previous results were quite alike, we use PROBPAT 2, such that every customer at least has a preference.

For the first experiment, we will only deviate the correctness of the estimate of the behaviour of a customer (COREST), such that the development of the revenue can be captured gradually. The amount of time windows which is estimated correctly per customer will be varied from zero to six. The COREST parameter will therefore be given in percentages. Thus, a correct estimate of five out of the six time windows means a COREST percentage of 83.3%. Also, for computational ease, we will assume that the windows which are estimated correctly are all in consecutive hours. This is justified by the fact that the majority of people mostly are available on only half of the day, instead of a few hours randomly divided over a day. Furthermore, this is in line with the earlier assumptions about the probability patterns, which is also why the possibility of wrapping around the beginning and end of the day remains available. The incorrect windows can be both at the beginning or at the end of the series of time windows with an equal probability, such that randomness is kept present. In Table 6, the results of a base case are displayed. For each case, all customers are estimated correctly with the given percentage.

All results where COREST = 100% have the same characteristics as the results in Section 5.1, since the estimate consists only of correct windows. Also, note that most results are not lower than the NOINC. Furthermore, when COREST = 0%, all incentive methods have an equal result. This is because all offered incentives will have no influence. Because the results are actually the averages of the methods (and are not filled in manually), this shows us the correct work of our algorithm choosing windows, such that different percentages can be compared with each other.

The most important conclusion to be drawn from these results, is that offering incentives

Table 6: Base results COREST

		Walkaway **is not** allowed					Walkaway **is** allowed				
COREST %	|U|	NOINC	CHPFLT	CHPLP	BSTLP	BST	NOINC	CHPFLT	CHPLP	BSTLP	BST
100%	1	1167.30	1230.42	1294.68	1248.06	1317.09	1029.07	1253.54	1260.38	1247.71	1317.09
	2	1167.30	1282.82	1290.73	1285.94	1317.09	1029.07	1196.12	1275.13	1272.57	1317.09
	3	1167.30	1222.60	1257.19	1272.07	1317.09	1029.07	1157.09	1271.74	1297.37	1317.09
	4	1167.30	1177.56	1268.99	1283.95	1317.09	1029.07	1095.30	1287.24	1281.63	1317.09
Average		1167.30	1228.35	1277.90	1272.50	1317.09	1029.07	1175.51	1273.62	1274.82	1317.09
% impr.		0.00	5.23	9.47	9.01	12.83	0.00	14.23	23.76	23.88	27.99
83.3%	1	1167.30	1259.74	1245.66	1235.81	1317.09	1029.07	1271.80	1254.90	1215.47	1317.09
	2	1167.30	1232.13	1256.49	1262.73	1317.09	1029.07	1190.04	1214.52	1245.15	1317.09
	3	1167.30	1254.52	1260.17	1277.66	1317.09	1029.07	1109.37	1192.36	1201.20	1317.09
	4	1167.30	1221.41	1264.07	1269.03	1317.09	1029.07	1079.57	1223.51	1225.36	1317.09
Average		1167.30	1241.95	1256.60	1261.31	1317.09	1029.07	1162.70	1221.32	1221.79	1317.09
% impr.		0.00	6.40	7.65	8.05	12.83	0.00	12.99	18.68	18.73	27.99
66.7%	1	1167.30	1230.71	1250.77	1247.95	1317.09	1029.07	1187.39	1177.68	1222.35	1317.09
	2	1167.30	1191.41	1219.78	1235.22	1317.09	1029.07	1148.39	1217.16	1220.09	1317.09
	3	1167.30	1189.71	1253.94	1241.38	1317.09	1029.07	1084.41	1215.18	1223.67	1317.09
	4	1167.30	1188.14	1239.88	1237.00	1317.09	1029.07	1067.13	1212.43	1212.32	1317.09
Average		1167.30	1199.99	1241.09	1240.39	1317.09	1029.07	1121.83	1205.61	1219.61	1317.09
% impr.		0.00	2.80	6.32	6.26	12.83	0.00	9.01	17.16	18.52	27.99
50%	1	1167.30	1232.64	1234.29	1225.28	1317.09	1029.07	1157.06	1154.37	1169.67	1317.09
	2	1167.30	1231.42	1250.53	1252.08	1317.09	1029.07	1170.06	1209.37	1235.60	1317.09
	3	1167.30	1169.36	1233.62	1228.20	1317.09	1029.07	1078.64	1214.85	1222.83	1317.09
	4	1167.30	1166.32	1227.46	1223.52	1317.09	1029.07	1060.15	1218.90	1214.93	1317.09
Average		1167.30	1199.94	1236.48	1232.27	1317.09	1029.07	1116.48	1199.38	1210.76	1317.09
% impr.		0.00	2.80	5.93	5.57	12.83	0.00	8.49	16.55	17.66	27.99
33.3%	1	1167.30	1182.95	1228.17	1209.34	1317.09	1029.07	1146.67	1151.95	1127.67	1317.09
	2	1167.30	1182.41	1226.84	1232.61	1317.09	1029.07	1138.26	1156.27	1158.71	1317.09
	3	1167.30	1154.79	1223.43	1229.99	1317.09	1029.07	1075.77	1153.75	1145.90	1317.09
	4	1167.30	1197.53	1230.29	1220.83	1317.09	1029.07	1052.45	1148.71	1146.74	1317.09
Average		1167.30	1179.42	1227.18	1223.19	1317.09	1029.07	1103.29	1152.67	1144.75	1317.09
% impr.		0.00	1.04	5.13	4.79	12.83	0.00	7.21	12.01	11.24	27.99
16.7%	1	1167.30	1174.33	1184.17	1195.23	1317.09	1029.07	1120.20	1124.93	1133.93	1317.09
	2	1167.30	1174.29	1183.50	1200.11	1317.09	1029.07	1085.08	1121.03	1135.77	1317.09
	3	1167.30	1174.37	1192.99	1200.72	1317.09	1029.07	1045.90	1119.11	1122.88	1317.09
	4	1167.30	1169.94	1197.35	1200.72	1317.09	1029.07	1051.58	1126.07	1126.17	1317.09
Average		1167.30	1173.24	1189.50	1199.20	1317.09	1029.07	1075.69	1122.79	1129.69	1317.09
% impr.		0.00	0.51	1.90	2.73	12.83	0.00	4.53	9.11	9.78	27.99
0%	1	1167.30	1167.30	1167.30	1167.30	1317.09	1029.07	1029.07	1029.07	1029.07	1317.09
	2	1167.30	1167.30	1167.30	1167.30	1317.09	1029.07	1029.07	1029.07	1029.07	1317.09
	3	1167.30	1167.30	1167.30	1167.30	1317.09	1029.07	1029.07	1029.07	1029.07	1317.09
	4	1167.30	1167.30	1167.30	1167.30	1317.09	1029.07	1029.07	1029.07	1029.07	1317.09
Average		1167.30	1167.30	1167.30	1167.30	1317.09	1029.07	1029.07	1029.07	1029.07	1317.09
% impr.		0.00	0.00	0.00	0.00	12.83	0.00	0.00	0.00	0.00	27.99

Results computed with probability pattern 2.

will have a positive influence, even if the behaviour of all customers is estimated very poorly. Especially in the case where walkaway is allowed, would every customer be estimated such that only one time window is correct, an improvement of the average profit of 9 to 10% can still be made using the CHPLP or BSTLP. Also in the case of not allowing walkaways, an improvement is present, even though it is small. While the CHPFLT has smaller improvements, we again see the good performances of the CHPFLT when $|U|$ is 1 or 2. This also holds for when walkaway is allowed, such that the average of the CHPFLT might give a distorted picture, because small sizes of U perform well, would the estimate not be almost completely incorrect. This same pattern is not present for the LP-based methods, as it depends on the percentage which $|U|$ performs best, such that it will most likely depend on the instance. However, for all $|U|$, the profit is higher than the NOINC, such that we will on average improve our profit.

Now we add the other direction of estimating customers incorrectly, NUMCOR. Because in Table 6, the improvements were larger in the case where walkaway is allowed, we only discuss the results where walkaway is allowed to make analysis easier and be able to draw better conclusions regarding the differences in profit. Also, we will evaluate the results with NUMCOR as a percentage in steps of ten percent. For example, three out of ten customers are estimated correctly, while the other seven customers are estimated falsely with the given COREST percentage. Thus, we add the possibility of customers being estimated correctly. Next to this, we exclude the BSTLP because of its large computation time and almost equal performance to the CHPLP.

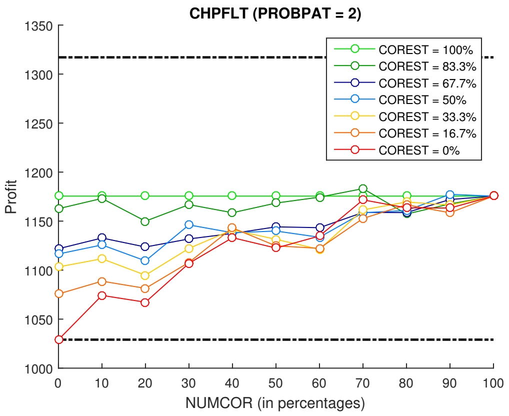
The decision on which customer is estimated correctly is tricky. Firstly, we do not want all the customers grouped at the beginning or at the end, because in most cases, only the first twenty customers are actually considered because of a full schedule. Furthermore, if customers are estimated incorrectly shortly after each other, which could be the case, our schedule might be ruined completely due to a couple of unlucky choices. Lastly, we want to be able to compare the different results regarding the NUMCOR and draw conclusions from it, such that we need to know the buildup. Therefore we choose to spread out those customers estimated correctly, divided in groups of ten (the first ten, the eleventh to twentieth, and so forth). We spread the correct customers centered from the middle, such that all NUMCOR percentages have an equal buildup, which can be found in Table 7. This particular buildup of spreading out those customers is chosen such that the possibility of extreme outliers in the form of extremely unlucky divisions is kept to a minimum, for a good first analysis.

Table 7: Correctly estimated customers grouped by NUMCOR percentage

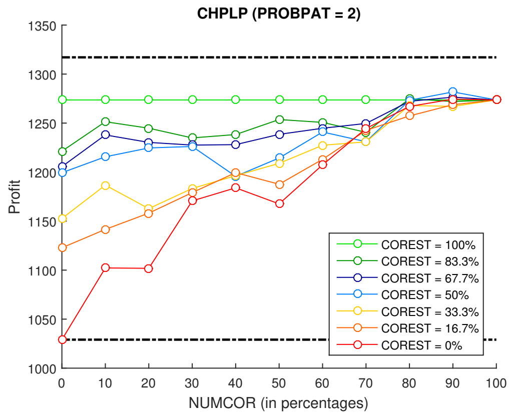
NUMCOR	Correct customers in each group of ten
0%	-
10%	{5}
20%	{3, 8}
30%	{3, 6, 10}
40%	{2, 4, 7, 9}
50%	{2, 4, 6, 8, 10}
60%	{1, 3, 5, 6, 8, 10}
70%	{1, 2, 4, 5, 7, 8, 9}
80%	{1, 2, 4, 5, 6, 7, 9, 10}
90%	{1, 2, 3, 4, 6, 7, 8, 9, 10}
100%	{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

In Figure 4, the results are averaged over all $|U|$, but the COREST percentages can be differentiated by the colours. The lower black, striped bar represent the lower bound of the NOINC method, while the higher black, striped bar represent the results of the best case scenario, such that the performance of the methods can be evaluated relatively.

Immediately, one can see the excellent performance of the CHPLP compared with the CHPFLT, as it outperforms the CHPFLT for all NUMCOR percentages. When we look at the different NUMCOR percentages, we see the COREST percentages stacked in accordance with their percentages. NUMCOR = 0% represents the performance as given in Table 6, as all customers are assumed to be estimated falsely. As NUMCOR increases, so does the profit. The build up of differences between the COREST percentages is present until half of the customers are estimated correctly for the CHPFLT, while this pattern persists another 10% extra for the CHPLP. Afterwards, all COREST percentages perform almost equally, even though a higher COREST percentage is still advantageous. This points out that when approximately 70 to 80% of all customers are estimated correctly, the accuracy of the estimates of the customers' behaviour is not very relevant, since the profit can only be improved marginally.



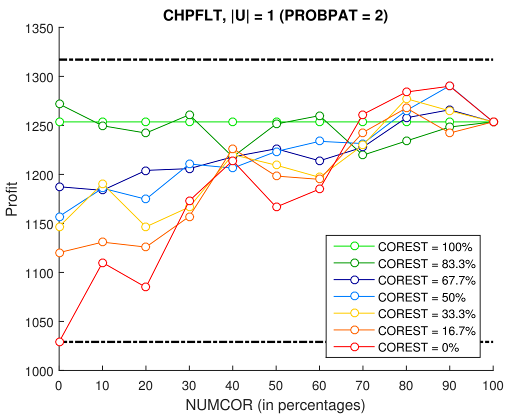
(a) CHPFLT



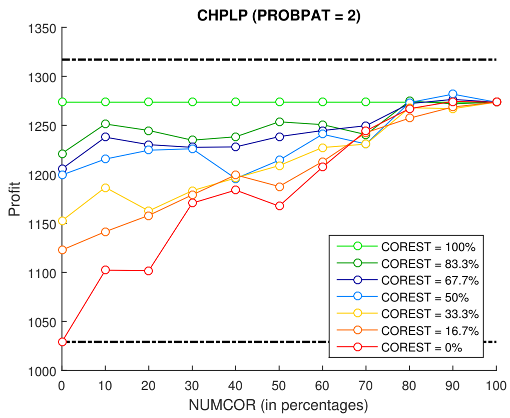
(b) CHPLP

Figure 4: Results from varying NUMCOR and COREST where walkaway is allowed.

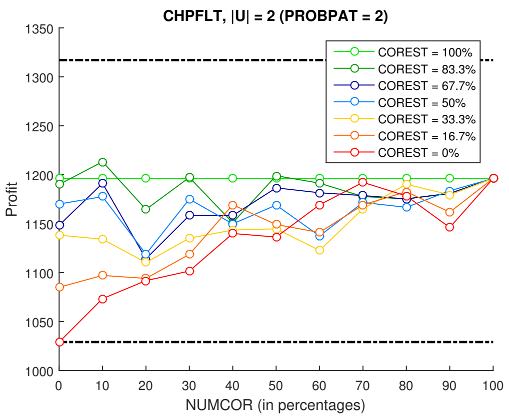
Another important observation is that when the more than one fifth of the customers is estimated correctly, the improvement in profit is already getting very significant. For both the CHPFLT and the CHPLP, the improvement when NUMCOR = 30% for COREST = 0% is more than half of the improvement of COREST = 100%. This shows us that offering incentives can already be very useful even if many customers are estimated incorrectly, regardless of the accuracy of the estimates.



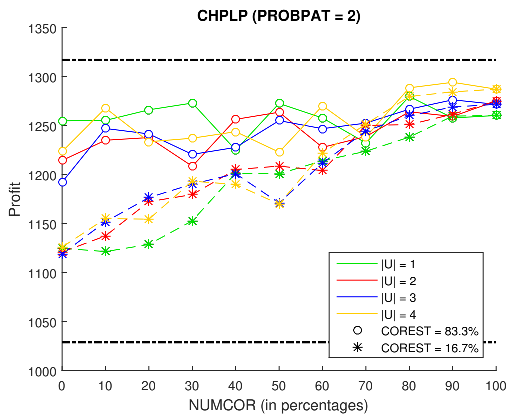
(a) CHPFLT, with results of $|U| = 1$



(b) CHPLP, with average results



(c) CHPFLT, with results of $|U| = 2$



(d) CHPLP, with results of all $|U|$

Figure 5: Comparison of results when the profit is split up in accordance with $|U|$.

When we take a closer look at the different $|U|$, we notice major differences. First of all, in Figure 5d we see an example of the behaviour of the results of the different sizes when considering two COREST percentages. One might immediately see the different behaviour of the $|U|$ for both percentages, even though this is just a highlight of two COREST percentages. When we consider all COREST percentages, no pattern can be found to draw a conclusion on which $|U|$ has better results. We also concluded this from Table 6, when NUMCOR was not present yet.

On the other hand, in Figure 5a we see the results of the CHPFLT using $|U| = 1$. In Table 6, its excellent performance compared to the other $|U|$ was already present. Now, when adding the NUMCOR percentages, this pattern perseveres, as all profits are significantly higher than the profits when $|U| = 2$, which can be seen in Figure 5c. $|U| = 3$ and $|U| = 4$ have even lower results, bringing down the averages of the profit to approximately the same as the results of $|U| = 2$. Furthermore, when we compare the CHPFLT where $|U| = 1$ with the CHPLP, showed in Figure 5b, we can see how well the CHPFLT actually performs. The results are sometimes even higher than the average of the CHPLP. Thus, we can conclude that we could also use the CHPFLT with $|U| = 1$ to get good results, would the computation time be important, even if the customer behaviour estimates are very inaccurate.

6 Conclusion and discussion

In this paper, we evaluated the influence of offering incentives to customers in order to increase the profit for home delivery services. We were able to create methods to decide on the distribution of incentives, such that the profit was actually increased. The use of the proposed incentive schemes make sure delivery schedules can be created more efficiently, such that delivery costs are reduced, more customers can be accepted and thus a higher profit can be achieved. Would a vendor not be able to fill its delivery schedule, incentives can still increase profit, especially if more enhanced incentive schemes are used. These schemes also showed us that incorporating intelligence in the incentive method increases the profit. However, if the computation times of the incentives would be very important, less sophisticated incentive schemes should be used.

The positive influence of incentives on the profit is not depending on the ability to fully exclude expensive time windows from a customers' choice range. We found the use of small incentives, which were not completely excluding time windows, to have a positive impact on the profit.

The amount of time windows receiving an incentive does also not need to be very large, as less or equal than three will be sufficient for an improvement. For simple incentive schemes, the largest improvement is made if only one time window is considered, while the more sophisticated incentive schemes are better off using approximately three time windows eligible for incentives.

We found that the estimates of customers' behaviour can be incorrect, while still increasing the profit by offering incentives. However, a trade-off occurs immediately, as the possible extra profit will depend on the method and the degree of incorrectness. Considering that degree, no evidence was found that the accuracy of the estimates and the total amount of customers estimated correctly are not equally important. Would we be able to estimate roughly three quarters of all customers correctly, the extra profit will only decrease marginally. This also is the case if we are slightly off with our estimates of the behaviour. However, for both criteria holds, would we be incorrect to a large extent, we lose at least half of our extra profit created by offering incentives. Most importantly though, the use of incentives will increase our profit, regardless of the accuracy of the joint customer behaviour.

All results are under influence of the customer behaviour model. Not only the current setup of the probability patterns can be discussed, but also the influence of the incentives. A proportional influence of these incentives might be more realistic, as it is difficult to completely exclude a time window for a customer. This also solves our problem with the simulation of probability changes in our model where customer behaviour is estimated incorrectly. More research on these behaviour models is essential, for both other probability patterns and other models concerning the influence of incentives, to draw better conclusions on the exact impact of incentives.

Furthermore, the data instance is quite small, such that only one vehicle is used. This influenced our results, as companies using many vehicles have a much larger choice regarding the division of customers to vehicles. Furthermore, we therefore had to assume that the capacity of the used vehicle was larger than the capacity needed. Campbell and Savelsbergh argue the constraints which follow from the large travel times and small delivery windows already restrict the number of orders which can be included in a delivery route, such that the capacity of a delivery truck is almost always enough. However, nowadays, this has changed. Many customers are located nearby such that a vehicle does not have long travel times and needs to handle many orders. Taking a restriction on the capacity into account will change the difficulty of this problem and follow-up research is desired.

Our results were only intended to investigate the possibilities of incentives. Especially for unknown customer behaviour, we only wanted to evaluate whether incentives are still useful. To fully understand the consequences of incorrect customer behaviour estimates, further research is needed, such that methods can be developed to handle this problem.

7 Bibliography

[1] Agatz, N., Campbell, A.M., Fleischmann, M. & Savelsbergh, M.W.P. (2011). Time Slot Management in Attended Home Delivery. *Transportation Science 45(3)*: 435-449.

[2] Bent, R.W. & Van Hentenryck, P. (2004). Scenario-Based Planning for Partially Dynamic Vehicle Routing with Stochastic Customers. *Operations Research 52(6)*: 977-987.

[3] Campbell, A.M. & Savelsbergh, M.W.P. (2004). Efficient Insertion Heuristics for Vehicle Routing and Scheduling Problems. *Transportation Science 38(3)*: 369-378.

[4] Campbell, A.M. & Savelsbergh, M.W.P. (2005). Decision Support for Consumer Direct Grocery Initiatives. *Transportation Science 39(3)*: 313-327.

[5] Campbell, A.M. & Savelsbergh, M.W.P. (2006). Incentive Schemes for Attended Home Delivery Services. *Transportation Science 40(3)*: 327-341.

[6] Feo, T.A. & M.G.C. Resende (1995). Greedy Randomized Adaptive Search Procedures. *Journal of Global Optimization 6*: 109134.

[7] Madsen, O.B.G., Tosti, K. & Vælds, J. (1995). A heuristic method for dispatching repair men. *Annals of Operations Research 61(1)*: 213-226.

[8] Nemhauser, G.L., & Wolsey, L.A. (1988). *Integer and combinatorial optimization.* New York: John Wiley & Sons, Inc.