

Time Slot Management in Attended Home Delivery

Niels Agatz, Ann Campbell, Moritz Fleischmann
and Martin Savelsbergh

ERIM REPORT SERIES <i>RESEARCH IN MANAGEMENT</i>	
ERIM Report Series reference number	ERS-2008-022-LIS
Publication	April 2008
Number of pages	32
Persistent paper URL	http://hdl.handle.net/1765/12245
Email address corresponding author	mfleischmann@rsm.nl
Address	Erasmus Research Institute of Management (ERIM) RSM Erasmus University / Erasmus School of Economics Erasmus Universiteit Rotterdam P.O.Box 1738 3000 DR Rotterdam, The Netherlands Phone: + 31 10 408 1182 Fax: + 31 10 408 9640 Email: info@erim.eur.nl Internet: www.erim.eur.nl

Bibliographic data and classifications of all the ERIM reports are also available on the ERIM website:
www.erim.eur.nl

REPORT SERIES
RESEARCH IN MANAGEMENT

ABSTRACT AND KEYWORDS	
Abstract	<p>Many e-tailers providing attended home delivery, especially e-grocers, offer narrow delivery time slots to ensure satisfactory customer service. The choice of delivery time slots has to balance marketing and operational considerations, which results in a complex planning problem. We study the problem of selecting the set of time slots to offer in each of the zip codes in a service region. The selection needs to facilitate cost-effective delivery routes, but also needs to ensure an acceptable level of service to the customer. We present two fully-automated approaches that are capable of producing high-quality delivery time slot offerings in a reasonable amount of time. Computational experiments reveal the value of these approaches and the impact of the environment on the underlying trade-offs.</p>
Free Keywords	<p>e-grocery, home delivery, time slots, vehicle routing, continuous approximation, integer programming</p>
Availability	<p>The ERIM Report Series is distributed through the following platforms:</p> <p>Academic Repository at Erasmus University (DEAR), DEAR ERIM Series Portal</p> <p>Social Science Research Network (SSRN), SSRN ERIM Series Webpage</p> <p>Research Papers in Economics (REPEC), REPEC ERIM Series Webpage</p>
Classifications	<p>The electronic versions of the papers in the ERIM report Series contain bibliographic metadata by the following classification systems:</p> <p>Library of Congress Classification, (LCC) LCC Webpage</p> <p>Journal of Economic Literature, (JEL), JEL Webpage</p> <p>ACM Computing Classification System CCS Webpage</p> <p>Inspec Classification scheme (ICS), ICS Webpage</p>

Time Slot Management in Attended Home Delivery

Niels Agatz <nagatz@rsm.nl>

Ann Campbell <ann-campbell@uiowa.edu>

Moritz Fleischmann <mfleischmann@rsm.nl>

Martin Savelsbergh <martin.savelsbergh@isye.gatech.edu>

April 15, 2008

Abstract

Many e-tailers providing attended home delivery, especially e-grocers, offer narrow delivery time slots to ensure satisfactory customer service. The choice of delivery time slots has to balance marketing and operational considerations, which results in a complex planning problem. We study the problem of selecting the set of time slots to offer in each of the zip codes in a service region. The selection needs to facilitate cost-effective delivery routes, but also needs to ensure an acceptable level of service to the customer. We present two fully-automated approaches that are capable of producing high-quality delivery time slot offerings in a reasonable amount of time. Computational experiments reveal the value of these approaches and the impact of the environment on the underlying trade-offs.

1 Introduction

The success of many e-commerce businesses hinges upon their ability to offer efficient and effective “last-mile” delivery. This task is especially challenging in attended home delivery, particularly for Internet grocers. The dramatic failures of e-grocery pioneers such as Streamline and Webvan underline the importance of a cost-effective delivery operation. Despite initial failures, the online grocery market has recovered and has grown spectacularly since 2003, showing an annual growth rate of 42% (Berning et al. [2005]).

To provide a high service level and to avoid delivery failures as much as possible, it is customary in attended home delivery services for the company to offer the customer a choice of narrow delivery time slots. The specific time slot offering does not only impact the perceived customer service, but also the expected delivery efficiency. In this paper, we consider the tactical problem of selecting the time slots to offer in each zip code of the delivery region. This selection needs to offer an acceptable level of service in each zip code

and needs to be able to yield cost-effective daily delivery routes. Once such a time slot schedule is put in place, there is the operational problem of managing the availability of the offered time slots. We will focus only on the tactical, not the operational problem, in this paper.

Our research effort is motivated by operations at Albert.nl, the Internet channel of Albert Heijn, the largest grocery supermarket chain in the Netherlands and a subsidiary of Royal Ahold. Albert.nl is currently the only full-scale Internet grocer in the Netherlands. Started in 2001, their delivery service is currently available to around 60% of the country's households. From dedicated warehouses, the company uses small vans to deliver a broad assortment of grocery products to each customer's home. The time slot offering at Albert.nl is typical for those in the e-grocery industry (see for example Peapod.com and Tesco.com). The customer can choose a 2-hour delivery slot from a menu of time slots available in his or her specific zip code. Each time slot has a corresponding delivery fee ranging from €4.95 to €8.95 depending on the day and time of delivery. On a day-to-day basis, when delivery capacity becomes tight, certain time slots in certain zip codes may be closed. Closing a time slot at Albert.nl means that it is no longer available for the customer to choose (the slot turns "grey" on the web site). Routes are planned for each morning and afternoon shift. After the order cut-off time for a shift has passed and thus all orders are known, the delivery routes for that shift are planned using a commercial vehicle routing package. The routing package properly takes delivery windows, dwell times, and vehicle capacities into account.

Albert.nl uses a differentiated time slot offering linked to the demand characteristics of each zip code. This means that a zip code with low demand potential receives only a few weekly time slots in an attempt to concentrate that demand, where high demand zip codes receive a larger number of slots. The slots offered, regardless of the total number, must also exhibit a balance over the week between morning and afternoon time slots to ensure an acceptable level of service and choice to the customers. Determining the specific time slots for each zip code involves a careful trade-off between marketing and operational considerations. After deciding the number of morning and afternoon slots that will satisfy marketing needs, referred to hereafter as service requirements, the specific time slots are selected for each zip code to satisfy operational needs such as a low-cost delivery schedule. Because delivery vehicles may visit several zip codes during a single time slot and a delivery route spans multiple time slots, routing considerations should play a vital role in the construction of a time slot schedule. Assigning specific time slots to a zip code should not be done in isolation, but should be considered jointly for neighboring slots. This results in a complex planning problem.

The time slot schedule employed at Albert.nl is created manually. Due to its complexity, this task takes several weeks to complete. Clearly, this is undesirable in an environment that continually changes as a result of a 30% annual growth rate. In particular, this manual process greatly limits their ability to assess and compare potential future scenarios. Therefore, Albert.nl is considering options for automating the time slot

schedule generation. In this paper, we present two approaches for making the time slot decision that apply not only to Albert.nl, but also for many other companies offering attended delivery services.

More specifically, we address the following time slot management problem (TSMP). Given service requirements and average weekly demands for each zip code in the delivery region, determine the set of time slots to offer in each zip code so as to minimize expected delivery costs while meeting the service requirements. We explore and compare two modeling and solution alternatives, including a continuous approximation and an integer programming approach for the TSMP. The continuous approximation focuses on a realistic evaluation of a given time slot schedule and then uses relatively simple optimization techniques to improve the schedule. The integer programming approach uses simpler routing cost approximations but more sophisticated optimization machinery. We study the effect of these different strategies.

Our main contributions are:

- The identification and explanation of a relevant planning concern in the e-grocery business that gives rise to challenging optimization problems.
- The development of two fully-automated approaches, one based on continuous approximation and one based on integer programming, capable of producing high-quality time slot schedules much faster than the current manual process.
- A computational study analyzing the viability and merits of the two approaches.
- A computational study investigating the impact of changes in the environment, such as increases in demand and demand variability, more relaxed or more stringent service requirements, and changes in the structure of the delivery time slot template.

The remainder of the paper is organized as follows. In Section 2, we summarize the relevant literature. In Section 3, we introduce notation and outline our modeling framework. In Sections 4 and 5, we present the continuous approximation and integer programming approaches, respectively. In Section 6, we describe the design of our computational experiments and present the results. Finally, in Section 7, we summarize our main insights and discuss directions for future research.

2 Literature

The choice of time slots is one of many critical decisions in the design of an attended home delivery system, as discussed in Agatz et al. [2007, 2008]. Other important issues include the choice between store-based and warehouse-based delivery, inventory allocation, pricing, and return handling. Home delivery and the design

and management of time slots has received little attention in the distribution and routing literature. Time slots are typically assumed to be exogenous information, e.g., specified by the customer or set by the sales and marketing department. The rise of e-commerce, which offers numerous opportunities for the use and active management of time slots, is changing this.

Much of the early research on “last-mile” delivery primarily examines the impact of different time slot lengths. For example, Punakivi and Saranen [2001] assess the impact of the time slot length on transportation costs for attended and unattended deliveries. Their results indicate substantial efficiency gains when time constraints are relaxed. Completely flexible, unattended delivery reduces costs by up to a third, relative to attended delivery with 2-hour time slots. Lin and Mahmassani [2002] summarize the delivery policies employed by many online grocers in the U.S. and use vehicle routing software to evaluate the performance of some of these policies on a few realistic instances of the problem. Both unattended and attended policies are considered along with different time slot lengths.

Another related stream of research focuses on the real-time management of time slots in attended home delivery. This work draws on revenue management concepts and applies them in a distribution setting. In this vein, both Bent and Hentenryck [2004] and Campbell and Savelsbergh [2005a] examine which deliveries to accept or reject. Their approaches exploit stochastic information about future requests to decide on requests under consideration. Bent and Hentenryck [2004] aim to maximize the number of accepted requests by controlling the time slots offered, but they do not consider rejecting an “expensive” delivery to preserve resources for more, future deliveries, in contrast with Campbell and Savelsbergh [2005a]. Campbell and Savelsbergh [2005b] study the use of incentive schemes to reduce delivery costs by encouraging the selection of time slots that result in low-cost delivery routes. In contrast to the real-time management of time slots, our TSMP is tactical in nature and allocates time slots geographically prior to actual order intake. The resulting time slot schedules serve as a starting point for subsequent real-time adjustments.

Anticipating the impact of the offered time slots on delivery routes is crucial when solving the TSMP. This links the TSMP to the vast body of literature on vehicle routing with time windows. See Braysy and Gendreau [2005a,b] for an extensive review. In particular, the TSMP shares some aspects with dynamic and stochastic vehicle routing. In dynamic vehicle routing, as discussed in Gendreau and Potvin [1998], not all inputs are known beforehand but become known concurrently with the determination of the solution. In stochastic routing, some components of the problem are stochastic, such as travel times (Kenyon and Morton [2003]), demand (Dror et al. [1989]), or customers (Hvattum et al. [2006]). Another problem with features similar to those of the TSMP is the period vehicle routing problem with service choice (PVRP-SC), where vehicle routes need to be constructed over a period of time and service frequency is a decision in the model (Francis et al. [2006], Francis and Smilowitz [2006]).

Given the computational complexity of the vehicle routing problem alone, it is unrealistic to incorporate a full routing model in the tactical TSMP. Instead, routing costs resulting from time slot selections have to be approximated. One way is to model routing costs using the continuous approximation method (Daganzo [1987a,b]). This approach represents demand by continuous functions and assesses system-wide costs by aggregating over “local” cost estimates. For an overview of continuous approximation models in logistics, see Daganzo [2005] and Langevin et al. [1996]. Another option is to explicitly model routing decisions but on a more aggregate level than a full VRP. We will explore both of these alternatives.

3 Assumptions and Notation

The objective of the TSMP is to minimize the expected delivery costs for the customer orders ensuing from the offered delivery time slots. This requires assumptions on the demand response to a certain time slot offering. Throughout our analysis we make the following two key assumptions, where demand is measured in number of customer orders:

1. The expected weekly demand for each zip code is known and independent of the set of offered time slots.
2. The expected weekly demand for each zip code is divided evenly over the set of offered time slots.

The first assumption supposes that customers exhibit a certain degree of flexibility, such that changing the time slots offered does not result in lost sales. Historical data from Albert.nl support the validity of this assumption for moderate changes in the time slot offering. The second assumption implies that all time slots are equally popular. Differentiated delivery fees are introduced specifically for this reason. They counterbalance differences in time-slot popularity and smooth demand. Historical data from Albert.nl show the effectiveness of differentiated delivery fees and indicate that the equal time slot popularity assumption is reasonable.

We further assume that all orders are the same size in terms of the number of totes. The size of an order impacts the number of orders that fit in a delivery vehicle. We conducted a few computational experiments in which we varied order sizes and found that it had little or no impact on the results.

The marketing-imposed service requirements limit the set of feasible time slot offerings. Our models treat these service requirements as exogenous. The service requirements employed at Albert.nl specify for each zip code the required number of available delivery slots in each shift, e.g. one on Monday morning and two on Tuesday afternoon. As a result of these specific service requirements and the above demand assumptions, the problem decomposes by shift. Thus, separate problems need to be solved for each shift.

Our objective function considers fixed costs as well as variable costs, incorporating both distance-related and time-related costs. Initially, we do not take into account delivery fees and solely focus on minimizing expected delivery costs. Later, in Section 6, we consider the delivery fee revenues as part of the objective function.

In Table 1, we introduce notation that is common to both of our modeling approaches. Note that e_i is the weekly demand for zip code i divided by the number of time slots that need to be offered in zip code i .

\mathcal{Z} :	set of zip codes (indexed by i)
\mathcal{T} :	set of time slots (indexed by s)
\mathcal{V} :	set of vehicles
e_i :	expected demand for zip code i for each time slot
Q :	vehicle capacity (in terms of the number of orders that can be accommodated)
f :	vehicle fixed cost
c^n :	cost per mile for vehicle and driver usage when driving within a zip code and within a time slot
c^z :	cost per mile for vehicle and driver usage when driving between different zip codes within a time slot
c^t :	cost per mile for vehicle and driver usage when driving between zip codes in consecutive time slots
c^0 :	cost per mile for vehicle and driver usage when driving to or from the depot
d_{ij} :	distance between centers of zip codes i and j
τ :	dwell time per order

Table 1: Notation

4 Continuous Approximation Approach

At the heart of the continuous approximation approach is a model for estimating the delivery cost of a *given* time slot schedule for a set of zip codes. Based on this delivery cost estimate, a local search is performed to iteratively improve the time slot schedule. The original continuous approximation approach proposed by Daganzo [1987a] does not consider time slots and distinguishes two components of a vehicle tour, namely the stem distance between the depot and the delivery area and the distance between consecutive stops within the delivery area. To account for time slots, we expand these ideas and distinguish four components of a vehicle tour (see Figure 1):

- d^0 : the stem distance to or from the depot to a stop in a time slot;
- d^n : the distance between stops within a zip code within a time slot;
- d^z : the distance between stops in different zip codes within a time slot; and
- d^t : the distance between stops in consecutive time slots.

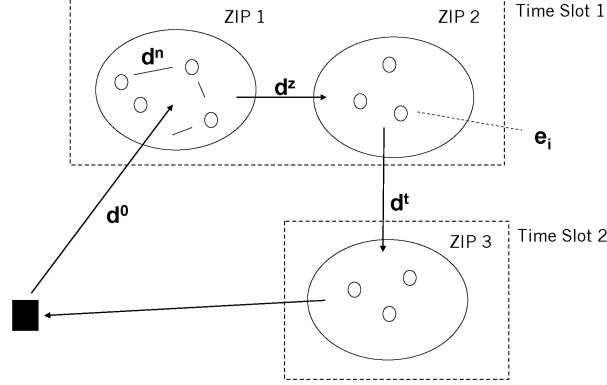


Figure 1: Routing components

The corresponding travel times are denoted by h^0 , h^n , h^z , and h^t .

In line with the continuous approximation methodology, we estimate the distance values for these components of a vehicle tour based on *local* data. We estimate the distance values from the perspective of each zip-code time-slot combination (i, s) in the time slot schedule. These local estimates are then used to compute a local estimate of the distance per order (excluding the stem distance to and from the depot) as follows:

$$DPO(i, s) = \frac{1}{n_{is}} \left[n_{is}^t \left(\frac{n_{is}^t - 1}{n_{is}^t} d_{is}^t + n_{is}^z \left(\frac{n_{is}^z - 1}{n_{is}^z} d_{is}^z + (n_{is}^n - 1) d_{is}^n \right) \right) \right],$$

where

- n_{is} denotes the estimated number of orders per route,
- n_{is}^t denotes the estimated number of time slots covered by a route,
- n_{is}^z denotes the estimated number of zip codes visited in time slot on a route, and
- n_{is}^n denotes the estimated number of orders delivered in a zip code in a time slot on a route.

These n -values are again estimated based on local data from the perspective of a given zip-code time-slot (i, s) .

By multiplying the distance per order with the cost per kilometer, we obtain a local estimate for the distance-related costs (excluding stem distances). This, in turn, yields a local estimate of the cost per order (excluding stem costs and fixed vehicle costs):

$$CPO(i, s) = \frac{1}{n_{is}} \left[n_{is}^t \left(\frac{n_{is}^t - 1}{n_{is}^t} d_{is}^t c^t + n_{is}^z \left(\frac{n_{is}^z - 1}{n_{is}^z} d_{is}^z c^z + (n_{is}^n - 1) d_{is}^n c^n \right) \right) \right].$$

We obtain an estimate of the expected delivery cost associated with a given time slot schedule by multiplying the expected number of orders e_i for a zip-code time-slot combination (i, s) with the cost per order $CPO(i, s)$, aggregating over all zip codes and time slots, and adding stem costs and fixed vehicle costs:

$$\text{expected delivery cost} \approx \sum_{i \in \mathcal{Z}} \sum_{s \in \mathcal{T}} e_i CPO(i, s) + d^0 c^0 + fv,$$

where d^0 is the estimated total stem distance and v is the estimated number of vehicles required. Note that we sum over all zip codes and all *offered* time slots for that zip code. Also note that we omit labor costs for the dwell times since these costs are constant and independent of the offered schedule.

4.1 Estimating Vehicle Tour Components – Part I

In this subsection, we discuss how the vehicle tour components are estimated from the perspective of a given zip-code time-slot combination (i, s) . We start with the vehicle tour components used in the calculation of the cost per order.

4.1.1 Distance between stops within a zip code within a time slot (d_{is}^n)

Statistical analysis of recent routing data at Albert.nl shows that the driving distance between two customers within the same zip code is similar across zip codes; we use d^n to represent this distance. The same is true for the travel time between two customers within the same zip code, so we use h^n to represent this travel time.

4.1.2 Distance between stops in different zip codes within a time slot (d_{is}^z)

Daganzo [2005] approximates the distance between two consecutive stops of a route through a region with a slowly varying demand density δ by $k/\sqrt{\delta}$, where k is a dimensionless constant that is independent of the region shape. We apply this approach on a zip code level and consider the density of *open* zip codes in time slot s in the neighborhood of i . Let \mathcal{Z}_i be a collection of zip codes in the neighborhood of zip code i (including zip code i itself). In our calculations, we take \mathcal{Z}_i to be the set of zip codes within a given maximum distance from the center of zip code i . Let δ_{is} be the density of open zip codes in \mathcal{Z}_i during time slot s . If a_j denotes the surface area of zip code j and \mathcal{I}_{js} denotes whether zip code j is open during time slot s ($\mathcal{I}_{js} = 1$) or whether it is closed during time slot s ($\mathcal{I}_{js} = 0$), then

$$\delta_{is} = \frac{\sum_{j \in \mathcal{Z}_i} \mathcal{I}_{js}}{\sum_{j \in \mathcal{Z}_i} a_j}.$$

The estimate of d_{is}^z is then $k/\sqrt{\delta_{is}}$. We impose an upper bound \bar{d}^z on this estimate to handle very low densities δ_{is} . In our calculations, we set \bar{d}^z equal to twice the average diameter of a zip code. Thus,

$$d_{is}^z = \min(k/\sqrt{\delta_{is}}, \bar{d}^z).$$

The above calculation of δ_{is} does not take into account that there may be open zip codes with an expected demand per slot e_i smaller than 1. To account for this situation, we modify δ_{is} to be a weighted density:

$$\delta_{is} = \frac{\sum_{j \in \mathcal{Z}_i} \min(1, e_j) \mathcal{I}_{js}}{\sum_{j \in \mathcal{Z}_i} a_j}.$$

This can be interpreted as treating values of e_i smaller than 1 as probabilities of a demand occurrence.

4.1.3 Distance between stops in consecutive time slots (d_{is}^t)

We take a similar approach when computing d_{is}^t , but using zip code densities in preceding and succeeding time slots. We set $d^{pre} = d^n$ if time slot $s-1$ is open for zip code i , i.e., for consecutive open time slots we simply use the average distance between stops within a zip code. Otherwise, we set $d^{pre} = \min\{\bar{d}^z, k/\sqrt{\delta_{i,s-1}}\}$, where $\delta_{i,s-1}$ is the density of open zip codes in \mathcal{Z}_i in time slot $s-1$. Analogously, $d^{suc} = d^n$ if time slot $s+1$ is open for zip code i and $d^{suc} = \min\{\bar{d}^z, k/\sqrt{\delta_{i,s+1}}\}$ otherwise, where $\delta_{i,s+1}$ is the density of open zip codes in \mathcal{Z}_i in time slot $s+1$. Finally, we set $d_{is}^t = d^{pre}$ (d^{suc}) if s is the first (last) time slot of the shift and $d_{is}^t = (d^{pre} + d^{suc})/2$ otherwise.

4.1.4 Number of stops in a zip code in a time slot on a route (n_{is}^n)

The number of stops in a zip code in a time slot is limited by the demand e_i and by the length of the time slot l . Let ν^n be the maximum number of stops that can be made during a time slot of length l and recall that τ denotes the dwell time. Then $l - h_{is}^t = \nu^n \tau + (\nu^n - 1)h_{is}^n$ and thus $\nu^n = \frac{l - h_{is}^t + h_{is}^n}{\tau + h_{is}^n}$, where we subtract h_{is}^t from l to account for travel time between consecutive time slots. The number of stops in a zip code during a time slot n_{is}^n is therefore $\min\{e_i, \nu^n\}$.

4.1.5 Number of zip codes visited in a time slot on a route (n_{is}^z)

Analogously, the number of zip codes visited in a time slot is limited by the number of neighboring open zip codes n^{open} and by the length of the time slot l . Let ν^z denote the maximum number of zip codes that can be visited during a time slot of length l . Then $l - h_{is}^t = \nu^z [n_{is}^n \tau + (n_{is}^n - 1)h_{is}^n] + (\nu^z - 1)h_{is}^z$, and thus $\nu^z = \frac{l - h_{is}^t + h_{is}^z}{n_{is}^n \tau + (n_{is}^n - 1)h_{is}^n + h_{is}^z}$. We then set $n_{is}^z = \min\{n^{open}, \nu^z\}$.

4.1.6 Number of time slots covered on a route (n_{is}^t)

The number of time slots covered on a delivery tour is limited by the number of time slots in the shift $|\mathcal{T}|$ and by the number of orders that can be accommodated by the delivery vehicle Q . Therefore, we set

$$n_{is}^t = \min\{|\mathcal{T}|, \frac{Q}{n_{is}^n n_{is}^z}\}.$$

4.1.7 Number of stops on a route (n_{is})

Using the previously estimated parameters, the number of stops on a delivery tour is $n_{is}^t n_{is}^z n_{is}^n$.

4.2 Estimating Vehicle Tour Components – Part II

In the previous subsection, we have shown how we compute local estimates for the vehicle tour components used in the computation of the local estimate of the cost per order. What remains to be shown is how to estimate the number of vehicles required (v) and the stem distance (d^0).

4.2.1 Number of vehicles (v)

We start with the estimate of the number of vehicles required. The number of vehicles required for serving the total demand of a shift depends on the vehicle capacity and on the number of orders that can be delivered in a time slot, i.e. on physical capacity and on available time. We address each of these constraints separately.

We have local estimates for the number of orders per route (n_{is}). We take a demand-weighted average over these local estimates to obtain a global estimate of the number of orders per route. Dividing the expected demand for a shift by this global estimate of the number of order per route gives an estimate of the number of routes r . More precisely, let \mathcal{Z}^s denote the set of open zip codes in time slot s . Then we have

$$r = \frac{\sum_{i \in \mathcal{Z}^s, s \in \mathcal{T}} e_i}{\sum_{i \in \mathcal{Z}^s, s \in \mathcal{T}} \left(\frac{e_i}{\sum_{i \in \mathcal{Z}^s, s \in \mathcal{T}} e_i} \right) n_{is}} = \frac{(\sum_{i \in \mathcal{Z}^s, s \in \mathcal{T}} e_i)^2}{\sum_{i \in \mathcal{Z}^s, s \in \mathcal{T}} e_i n_{is}}.$$

Similarly, we estimate the number of vehicles required to serve all orders in a given time slot v^s , based on the expected demand per time slot and the estimated number of stops per time slot, with

$$v_s = \frac{(\sum_{i \in \mathcal{Z}^s} e_i)^2}{\sum_{i \in \mathcal{Z}^s} e_i n_{is}^z n_{is}^n}.$$

The number of vehicles required is the maximum of the number of routes r and the number of vehicles required to serve all orders in a time slot v^s . This is represented by $v = \max\{r, \max_{s \in \mathcal{T}} v_s\}$.

4.2.2 Stem distance (d^0)

Next, we consider the stem distance d^0 . We estimate d_s^0 , the stem distance traveled by a vehicle making its first delivery in time slot s , by taking the average distance between the depot and the open zip codes in time slots s as follows:

$$d_s^0 = \frac{\sum_{j \in \mathcal{Z}_i} \mathcal{I}_{js} \times d_{0j}}{\sum_{j \in \mathcal{Z}_i} \mathcal{I}_{js}}.$$

The same estimate applies to the distance traveled by a vehicle returning to the depot after its last delivery in time slot s . We use the estimates of the number of vehicles required in each time slot (v_s) for estimating

the number of vehicles that start their trip in time slot s and for estimating the number of vehicles that end their trip in time slot s . Let $v_0 = 0$ and $v_{|\mathcal{T}|+1} = 0$, then we estimate the stem distance as

$$d^0 = \sum_{s=0}^{|\mathcal{T}|} (\max\{0, v_{s+1} - v_s\} d_{s+1}^0 + \max\{0, v_s - v_{s+1}\} d_s^0).$$

4.2.3 Handling Overlapping Time Slots

A further complication arises if time slots are overlapping, as is the case at Albert.nl. In our calculations, we handle this situation by creating adjusted non-overlapping slots. To this end, we apportion the length of the overlapping part of two windows to both of the individual windows, based on their expected demand volumes. For example, consider two two-hour windows that overlap for one hour. Further, assume that the total expected demand from all open zip codes in the first time slot is twice as large as the total expected demand in the second time slot. Then, we create a first time slot of size 1.66 hours and a second slot of size 1.33 hours.

4.3 Solution Method

The discussion above shows how to estimate the expected costs of a given time slot allocation. The next step is to optimize the time slot allocation based on this evaluation. Note that this optimization problem is non-linear and non-convex. Therefore we use a simple greedy iterative improvement heuristic. We can start from any feasible time slot schedule. In our experiments, we begin with the schedule currently in place at Albert.nl. Next, we determine for each zip code the time slot allocation for that zip code that results in the minimum expected delivery cost (keeping the time slot allocation for all other zip codes fixed) by complete enumeration. We then adjust the current time slot schedule by implementing the time slot allocation for the zip code that achieves the minimum expected delivery cost. We repeat this process as long as there is a reduction in expected delivery costs greater than some threshold or until a maximum number of iterations is reached.

5 Integer Programming Approach

The continuous approximation model presented in the previous section does not model the operational-level routing decisions explicitly. Instead, it aims to reflect them implicitly in the employed cost approximations. In this section, we complement this approach with a model that deals with the embedded routing component of the TSMP in a more explicit way. We formulate TSMP as an integer program. As with the continuous

approximation model, the objective is to construct a time slot schedule that satisfies the service requirements and minimizes delivery costs.

Intuitively, one could model the routing costs by introducing variables that represent, for each vehicle, the sequence of stops at various zip codes within each time slot. However, since vehicles visit multiple time slots and may visit multiple zip codes within a time slot, the embedded routing component makes the integer program very hard to solve. Even finding feasible solutions in a reasonable amount of time is quite difficult with modern, sophisticated, and powerful integer programming solvers.

Therefore, we approximate the routing costs by using seeds. Even though the number of stops a vehicle makes during a shift may be large, the number of time slots in a shift is small; no more than four in the case of Albert.nl. Solving routing problems with at most four “stops” is much easier. This cost approximation is expressed graphically in Figure 2. The dashed circles surround the customers served in a single time slot with the center dot representing the seed. With the use of seeds, we can approximate the routing costs for a vehicle by the sum of two costs: (1) the cost of a route through the seeds associated with each time slot plus (2) a simple estimate of the costs incurred when visiting the customers within each single time slot based on the distance to the seed.

Our integer program has the following variables to represent the different decisions and associated costs:

- y_i^{vs} is 1 if zip code i is the seed for vehicle v for time slot s and 0 otherwise;
- x_i^{vs} is 1 if vehicle v visits zip code i in time slot s and makes a delivery and 0 otherwise;
- X_{ij}^{vs} is the distance from zip code i to zip code j if vehicle v serves customer i in time slot s and zip code j is the seed for time slot s and 0 otherwise;
- Y_{ij}^{vs} is the distance between the seed of time slot $s - 1$ and the seed of time slot s for vehicle v if zip code i is the seed of time slot $s - 1$ and zip code j is the seed of time slot s and 0 otherwise.

One zip code will serve as the seed (or representative) for the customers served in a particular time slot/vehicle combination, and this decision is represented by the y variable. The set of zip codes served in a particular time slot/vehicle combination is captured by the x variables. The estimate of the cost to serve a set of zip codes in a time slot is based on the sum of the distances between the zip codes and the seed, which are represented by the X variables. The cost of visiting the seeds associated with a vehicle, i.e., the route through the seeds, is captured by the Y variables. Note that the desired time slot schedule is implied by the x variables. Slot s is offered in zip code i if and only if $x_i^{vs} = 1$ for some v .

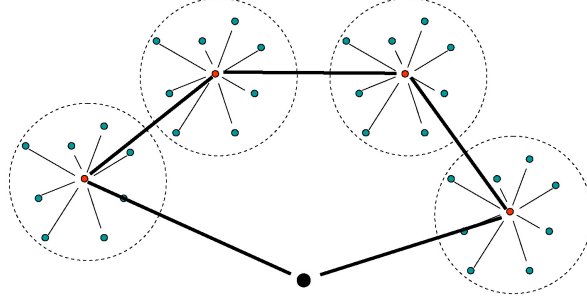


Figure 2: Routing with Seeds

Given the parameters defined in Table 1 and these variables, the objective function can be expressed as:

$$\begin{aligned} \min \sum_{v \in V} f + \sum_{i \in Z, j \in Z, v \in V, s \in S: s > 1} c^t Y_{ij}^{vs} + \\ \sum_{i \in Z, j \in Z, v \in V, s \in S} c^n X_{ij}^{vs} + \sum_{j \in Z, v \in V, s \in S} c^0 X_{0j}^{vs}. \end{aligned} \quad (1)$$

The first term represents the fixed cost for using vehicles/drivers. The second term represents the cost to travel between seeds. The last two sums represent the cost to serve zip codes from seeds and the depot from seeds, respectively. Since the first sum represents a fixed cost, we can eliminate it during the optimization. Also note that, as in the continuous approximation model, we omit labor costs for the dwell times since they are constant.

The continuous approximation model presented in Section 4 evaluates the cost of a *given* time slot schedule. Here, though, the model needs not only to capture the costs, but also needs to define what makes a schedule feasible. To do so, we must include some new parameters:

- r_i : the number of slots that must be open for customer i in a shift, i.e., the service requirement for i ,
- t^n : the travel time between customers in the same zip code,
- t^t : a multiplier to convert the distance between zip codes into a travel time between zip codes.

Next, we describe the constraints that define feasible solutions. First, we examine the linkage between the X and Y variables and the x and y variables. For the X variables, the linkage is defined as follows:

$$X_{ij}^{vs} = d_{ij} x_i^{vs} y_j^{vs} \quad i \in Z, j \in Z, v \in V, s \in S. \quad (2)$$

Equation 2 sets X_{ij}^{vs} equal to the distance from zip code i to zip code j if zip code i is served in time slot s by vehicle v and zip code j is the associated seed. This quadratic expression can be linearized as:

$$X_{ij}^{vs} \geq d_{ij} (x_i^{vs} + y_j^{vs} - 1) \quad i \in Z, j \in Z, v \in V, s \in S. \quad (3)$$

Equation 3 is valid because it forces X_{ij}^{vs} to have nonzero value d_{ij} only if both x_i^{vs} and y_j^{vs} equal 1; otherwise, the right hand side has a value less than or equal to zero. In the same way, we can define the linkage for the Y variables with:

$$Y_{ij}^{vs} \geq d_{ij}(y_i^{vs-1} + y_j^{vs} - 1) \quad i \in Z, j \in Z, v \in V, s \in S : s > 1 \quad (4)$$

Note that Equation 4 is only used for $s > 1$ since the cost to travel to the first seed from the depot will be handled by the X variables.

The next set of constraints enforces that every time slot has a seed:

$$\sum_{i \in Z} y_i^{vs} = 1 \quad v \in V, s \in S. \quad (5)$$

Note that enforcing that every time slot has a seed does not create unnecessary travel costs in case a vehicle serves no customers in a time slot, because the seed can be chosen to be the same as the seed of the previous (or subsequent) time slot.

The next set of constraints enforces that the demand served by a single vehicle does not exceed the vehicle capacity Q :

$$Q \geq \sum_{i \in Z, s \in S} e_i x_i^{vs} \quad v \in V. \quad (6)$$

Recall that we have assumed that the demand in a zip code is the same in every time slot offered.

Next, we ensure that the number of times a zip code is visited satisfies the service requirements:

$$\sum_{v \in V, s \in S} x_i^{vs} = r_i \quad i \in Z. \quad (7)$$

For simplicity we restrict each zip code to be visited by at most one vehicle during a particular time slot in our basic model:

$$\sum_{v \in V} x_i^{vs} \leq 1 \quad i \in Z, s \in S. \quad (8)$$

When we discuss extensions to the basic model, we will show how to relax this constraint to accommodate situations in which the demand in zip code is such that a visit by a single vehicle during a particular time slot is insufficient. Where possible, though, this more restrictive model is used in our computational experiments. These constraints restrict the search space, which allows the solver to find better solutions in a shorter amount of time.

To properly estimate the routing costs, especially the travel time to and from the depot, we force the depot to be served in the first and last time slot, i.e., time slot 1 and time slot \bar{s} :

$$x_0^{v1} = 1 \quad v \in V, \quad (9)$$

$$x_0^{v\bar{s}} = 1 \quad v \in V. \quad (10)$$

Recall from the continuous approximation model that the maximum number of deliveries that a vehicle can make is constrained not only by the vehicle capacity but also by the length of each time slot. Enforcing this time constraint is one of the most complex parts of the integer programming model. The situation is further complicated by the fact that the number and the length of the time slots can vary by day and by shift. Finally, there is a need to distinguish the first time slot, “in between” time slots, and the final time slot.

Let l_1 represent the width of the first time slot, let l_2 represent the width of an “in between” time slot, and let l_3 represent the width of the final slot. Recall that travel from the depot and back to the depot can occur outside of the time windows, so it does not need to be incorporated within these time window length constraints. Then the constraints to limit the time spent on deliveries during a time slot are:

$$\sum_{i \in Z, j \in Z} t^t X_{ij}^{v1} + \sum_{i \in Z} (\tau e_i + t^n (e_i - 1)) x_i^{v1} + 0.5 \sum_{i \in Z, j \in Z} t^t Y_{ij}^{v2} \leq l_1 \quad v \in V \quad (11)$$

$$0.5 \sum_{i \in Z, j \in Z} t^t Y_{ij}^{vs} + \sum_{i \in Z, j \in Z} t^t X_{ij}^{vs} + \sum_{i \in Z} (\tau e_i + t^n (e_i - 1)) x_i^{vs} + 0.5 \sum_{i \in Z, j \in Z} t^t Y_{ij}^{vs+1} \leq l_2 \quad v \in V \quad (12)$$

$$0.5 \sum_{i \in Z, j \in Z} t^t Y_{ij}^{v\hat{s}} + \sum_{i \in Z, j \in Z} t^t X_{ij}^{v\hat{s}} + \sum_{i \in Z} (\tau e_i + t^n (e_i - 1)) x_i^{v\hat{s}} \leq l_3 \quad v \in V \quad (13)$$

Constraint (11) concerns the first time slot. The first term captures the travel time to the seed and the travel time between the zip codes visited during the time slot. The second term in constraint (11) captures the stop time at a customer served in a zip code as well as the travel time between customers served in a zip code. The final term in constraint (11) captures the travel time from the seed of this time slot to the seed of the next time slot. The 0.5 in the final term reflects the fact that half of this travel time is allocated to this time slot and the other half is allocated to the next time slot. Constraints (12) and (13) concern the subsequent time slots and are structured similarly.

The situation becomes even more complicated when we consider overlapping time windows. Overlapping windows is a means to provide additional flexibility to both the customer and the firm (see Section 6.2.4). To handle overlapping time windows, we add, in addition to the constraints for each individual time slot, another constraint that assures that the total length of the combined slots is not exceeded. As an example, consider the time slots 8:00-10:00 and 9:00-11:00 (referred to below by 1 and 2, respectively) and assume that the 8:00-10:00 time slot is the first time slot of the shift and that the 9:00-11:00 time slot is not the last time slot of the shift. Let the total time period covered by the time slots be denoted by l , which is 3 hours in this example. We add the following constraint:

$$\sum_{i \in Z, j \in Z} t^t (X_{ij}^{v1} + X_{ij}^{v2}) + \sum_{i \in Z, j \in Z} t^t (Y_{ij}^{v1} + 0.5 Y_{ij}^{v2}) +$$

$$\sum_{i \in Z} (\tau e_i + t^t(e_i - 1))x_i^{v1} + \sum_{i \in Z} (\tau e_i + t^t(e_i - 1))x_i^{v2} \leq l.$$

Lastly, we add symmetry breaking constraints. These constraints are not necessary for the correctness of the model, but help the integer programming solver find better solutions faster. The symmetry breaking constraints force the lower indexed vehicles to be the ones with the largest number of stops:

$$\sum_{i \in Z, s \in S} x_i^{vs} \leq \sum_{i \in Z, s \in S} x_i^{v-1s} \quad v \in V : v > 1. \quad (14)$$

5.1 Extensions

5.1.1 Choosing an optimal number of vehicles.

Now that we have presented the basic model, we show how it can be extended to allow the model to choose to use only a subset of vehicles. This allows for solving the model when the exact number of vehicles required is not known. We introduce the variable z^v equal to 1 if vehicle v is used and 0 otherwise. The objective function is modified by replacing the term $\sum_{v \in V} f$ by $\sum_{v \in V} f \cdot z^v$. Relevant constraints are also modified. Constraints (5) become

$$\sum_{i \in Z} y_i^{vs} = z^v \quad v \in V, s \in S,$$

constraints (6) become

$$Qz_v \geq \sum_{i \in Z, s \in S} e_i x_i^{vs} \quad v \in V, \quad (15)$$

and constraints (9) and (10) become

$$x_0^{v1} = z^v \quad v \in V,$$

$$x_0^{v\hat{s}} = z^v \quad v \in V.$$

Finally, we again add constraints to reduce the symmetry in the model, which should help reduce solution times. When only a subset of the vehicles is used, there are many equivalent solutions. By imposing

$$z^v \leq z^{v-1} \quad v \in V : v > 1,$$

the number of equivalent solutions is significantly reduced.

5.1.2 Allowing multiple vehicles to visit the same zip code in the same time slot.

When we allow multiple vehicles to visit the same zip code in the same time slot, several changes have to be made. In the basic model, constraints (7) ensure that the service requirements are met by counting the

number of visits to a zip code (by any vehicle). This is correct since at most one vehicle is allowed to visit a zip code during a particular time slot (constraints (8)). If we simply relax (or remove) constraints (8), then a zip code can be visited by multiple vehicles during a single time slot, but each visit would be counted towards the zip code's service requirements, which results in over-counting. To circumvent this from happening, we introduce binary variable u_i^s representing whether or not a particular time slot s is offered for a particular zip code i and replace constraints (7) with the following sets of constraints

$$\sum_{s \in S} u_i^s = r_i \quad i \in Z$$

and

$$\sum_{v \in V} x_i^{vs} \geq u_i^s \quad i \in Z, s \in S.$$

If multiple vehicles serve a zip code during a particular time slot, then the zip code's demand e_i must be distributed over the different vehicles. Thus, we also introduce variables q_i^{vs} that represents the portion of the demand of zip code i in time slot s that is served by vehicle v . These variables are defined by the sets of constraints

$$\sum_{v \in V} q_i^{vs} \geq e_i u_i^s \quad i \in Z, s \in S$$

and

$$q_i^{vs} \leq \min(e_i, Q) x_i^{vs} \quad v \in V, i \in Z, s \in S.$$

To complete the modification, the value e_i in constraints (11), (12), and (13) needs to be replaced by q_i^{vs} so as not to overestimate the delivery time and it needs to be replaced in constraints (15) to properly capture vehicle capacity.

6 Computational Experiments

In this section, we report on the results of a computational study to evaluate the performance of the proposed models and to analyze the impact of various characteristics of the environment on their performance. Recall that the TSMP is a tactical planning problem with an embedded routing problem on the operational level. We evaluate alternative time slot schedules through simulation. To this end, we generate multiple demand realizations for each schedule and determine the corresponding detailed routing costs using a commercial routing package. To evaluate the impact of characteristics of the environment on the time slot schedules, we vary individual characteristics and analyze the results.

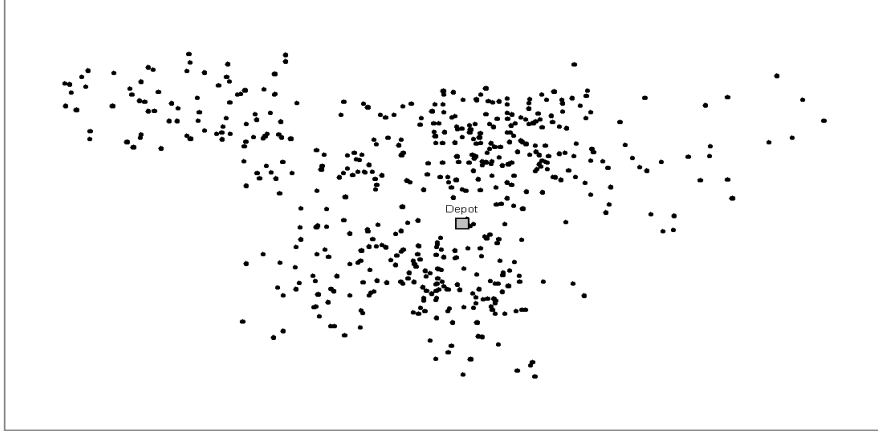


Figure 3: Geographical Distribution in Nijmegen Region

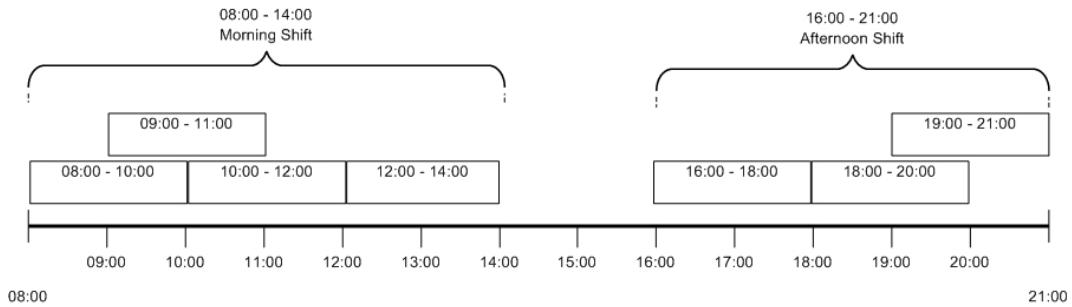


Figure 4: Albert.nl Time Slots

For our study, we use real-life data from Albert.nl. Specifically, we focus on the Nijmegen region, a subset of the service area of Albert.nl consisting of 30 3-digit zip codes with varying demand densities and covering a total area of approximately 1000 km². Figure 3 shows the geographic distribution of demand in the region. The area is served through a delivery hub in the city of Nijmegen. We consider a typical morning shift and a typical afternoon shift, because they differ in terms of total length, number of time slots, and time slot overlap (see Figure 4).

Furthermore, we use the current service requirements at Albert.nl for the Monday morning and the Tuesday afternoon. (We also performed simulations for other weekdays and obtained similar results.) The employed service requirements imply that in the morning schedule 24 zip codes have to receive 1 time slot, 3 zip codes have to receive 3 time slots, and 3 zip codes do not receive any time slot. In the afternoon schedule, 25 zip codes have to receive 1 time slot and 5 zip codes have to receive 2 time slots.

In our experiments we evaluate the following time slot schedules:

- CA: The schedule produced by the continuous approximation approach;

- IP: The schedule produced by the integer programming approach;
- ALBERT: The schedule that is currently in place at Albert.nl;
- ALL: The schedule in which all time slots are made available in all zip-codes where Albert.nl currently offers at least one time slot;
- NO-SLOT: The schedule in which a single time slot spanning the entire shift is made available in all zip-codes where Albert.nl currently offers at least one time slot.

The schedules ALL and NO-SLOT are included to provide bounds on the performance and to provide insights into the cost-service trade-off. Note that these two schedules do not use the service requirements of Albert.nl. Instead, the schedule ALL offers the highest level of customer service, and the schedule NO-SLOT offers the lowest level of customer service. The service levels will be reflected in the delivery costs, so the NO-SLOT schedule serves as a lower bound, albeit a weak lower bound, and the ALL schedule serves as an upper bound.

We supplement the comparison outlined above with values derived from the simulation results of a few randomly generated feasible time slot schedules. More precisely, we randomly generate 5 feasible time slot schedules, perform a simulation for each of them, and average the statistics obtained from the simulation runs. In our results tables, the resulting values will be labeled with RAND-ASGN.

For each problem instance, we generate 20 random demand instances for our simulations as follows. We start with a list of 448 customer addresses in the Nijmegen area. For each demand instance, we randomly select a number of addresses from this list for each zip code, based on the expected number of orders per time slot in that zip code. Specifically, we decide randomly and independently for each address whether or not to select it. We set the probability of selection of an address equal to the ratio between the expected number of orders in the zip code to which that address belongs and the total number of addresses in that zip code. Therefore, the total number of orders varies across demand realizations, with an average equal to the number of orders that was used as input to the optimization models.

For 20 different streams of random numbers, this results in 20 different random sets of addresses. We use the same sets for each schedule, in order to reduce variability across schedules to obtain more comparable results. Using this approach, demand realizations for different time slot schedules differ only in terms of the time slots associated with each order. For example, if time slot schedule CA has slots 2 and 4 open and time slot schedule IP has slots 3 and 4 open, then the same customer addresses have to be visited in slots 2 (CA) and 3 (IP) and in slots 4 (CA) and 4 (IP). For the ALL schedules, we determine the time slot associated with an order by means of a uniform random draw. For the NO-SLOT schedule, all orders are associated

with a single time slot spanning the entire shift.

The continuous approximation model is implemented in Excel with the heuristic implemented in VBA. The time slot schedules were produced on an Intel Pentium M 1.6GHz machine. The heuristic takes about 2 minutes to converge. The integer programming model is implemented in AMPL with CPLEX 9.0 as the solver. The time slot schedules were produced on an Intel Pentium D CPU 3.20GHz (x2) machine. A limit of 8 hours of CPU time was imposed. The limit was always reached without proving optimality. We used Shortec version 7.3.2.1, the routing tool from ORTEC (www.ortec.com), to construct the delivery routes and determine associated costs for each demand realization. Simulation experiments were run on an Intel Xeon 3.0 GHz machine and take approximately 25 minutes of computing time for 20 demand realizations of a time slot schedule.

6.1 Model Comparison

The first set of experiments is aimed at assessing differences in performance, if any, of the different time slot schedules. We compute the following statistics to compare the different time slots schedules: the average cost, the average number of vehicles used, the average distance, and the average total time (where the averages are taken over the 20 demand realizations). For convenience, we also display the percentage cost savings relative to ALL. For confidentiality, the cost figures are normalized by setting the cost for ALL for the morning shift to 100.

The results of the simulation experiments can be found in Tables 2 and 3. We observe that the gap

Table 2: Schedule Comparison - Morning

	# veh.	time	distance	cost	% savings
ALL	5.3	1399	443	100	-
RAND-ASG	4.8	1348	408	94	-5.6%
ALBERT	4.5	1361	384	92	-7.5%
CA	4.3	1328	377	90	-9.9%
IP	4.6	1310	393	91	-8.9%
NO-SLOTS	3.9	1210	329	82	-18.4%

between the upper bound (ALL) and the lower bound (NO-SLOTS) in terms of average cost is substantial. The absence of time slots in NO-SLOTS allows for more cost-effective delivery routes, but of course is inconvenient for the customers as they have to be home during the entire morning or afternoon. This illustrates the core trade-off between service and delivery cost, and is in line with the results of Punakivi and Saranen [2001] and Lin and Mahmassani [2002]. Time slot schedules provide a means for managing this trade-off by concentrating demand while still providing acceptable service levels.

Table 3: Schedule Comparison - Afternoon

	# veh.	time	distance	cost	% savings
ALL	5.4	1352	444	98	-
RAND-ASG	5.1	1310	417	94	-4.3%
ALBERT	4.6	1266	400	89	-8.9%
CA	4.5	1277	385	89	-9.5%
IP	4.6	1260	386	89	-9.6%
NO-SLOTS	3.9	1171	329	80	-19.0%

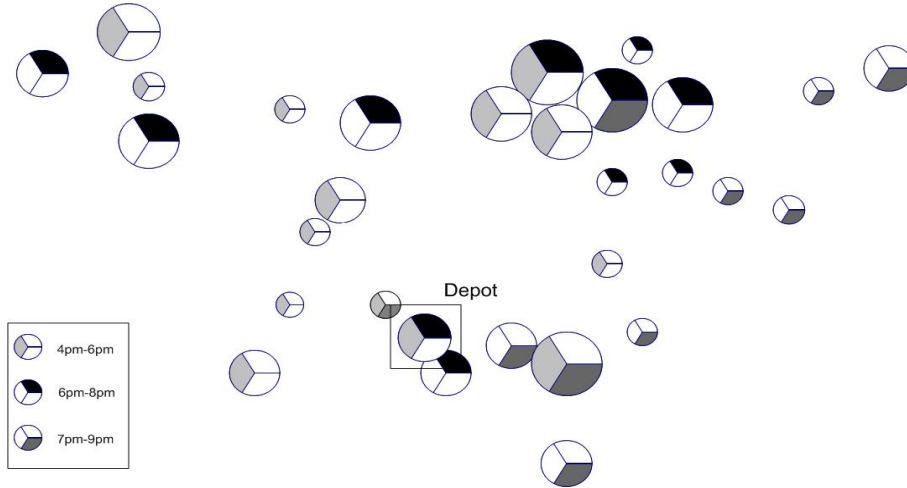


Figure 5: Afternoon Solution CA

Next, we observe that the morning and afternoon schedules currently in place at Albert.nl (ALBERT) are improved by both CA and IP, but that the savings are moderate. Albert’s current schedule seems to perform quite well. We also see that randomly constructed time slot schedules (RAND-ASGN) already yield a noticeable improvement over simply offering all time slots (ALL), but do not achieve the quality of Albert.nl’s current time slot schedule and that of the time slot schedules produced by our optimization models. We also observe, not surprisingly, that the average number of vehicles required in NO-SLOT is less than the average number of vehicles required by the optimized schedules, which in turn is less than the average number of vehicles required by ALL (and RAND-ASGN). This again reflects the trade-off between efficiency and service.

In addition to comparing performance differences, it is interesting to compare the actual time slot schedules. Figures 5 and 6 show the afternoon time slot schedules produced by the continuous approximation model and the integer programming model, respectively. For each zip code, the figure shows the assigned time slot(s). We use the size of the circle to reflect the number of expected orders for the zip code.

We observe that the distinct modeling paradigms of the CA and the IP approach are reflected in different

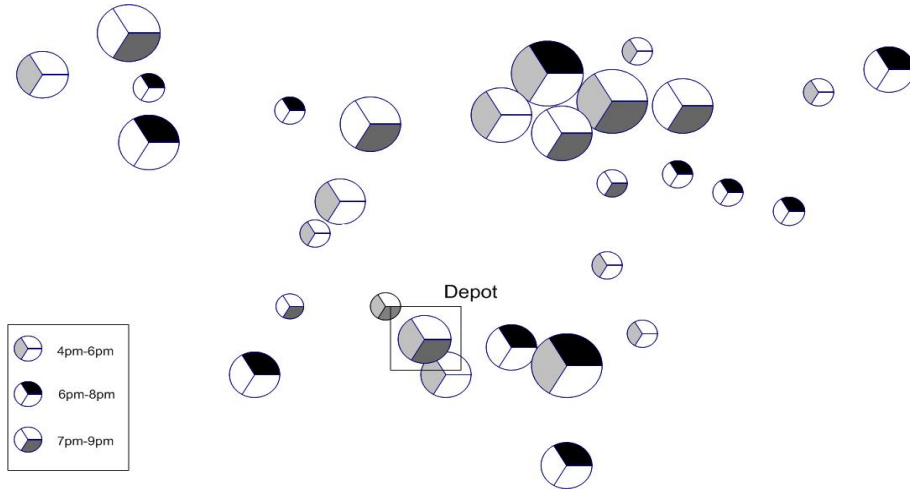


Figure 6: Afternoon Solution IP

characteristics of the resulting schedules. In Figure 5, we see that the continuous approximation model assigns the same time slot to zip codes that are geographically clustered together. This is especially the case for the zip codes that receive the late afternoon 19.00-21.00 time slot. In contrast, we see in Figure 6 that the integer programming model does not necessarily clusters geographically close zip codes, but forms “vehicle routes.” A possible explanation for the observed behavior is the fact that the continuous approximation approach aims to increase the demand density within a time slot for each zip code. Therefore, it tends to construct large zip code clusters for each time slot. Because the integer programming model explicitly considers the individual vehicle routes, it is more concerned with the zip code groups that can be visited by individual vehicles over the entire time period. From an application perspective, there does not appear to be a reason for either of these schedules to be preferable.

We conclude from the above experiments that our models achieve our main objective, namely supporting the design of effective time slot schedules in real-life scenarios. By automizing the generation of time slot schedules the models significantly reduce the planning effort compared to the current practice of manual planning. Thereby, they allow what-if analyses of different scenarios. In what follows, we address some of these scenarios and discuss the insights that they yield.

6.2 Changes in the Environment

As the e-grocery channel is still a relatively new distribution channel, understanding the impact of characteristics of the environment on its performance is of enormous value to companies experimenting with it. Therefore, the second part of our computational study focuses on these issues. All of the percentage differences reported are relative to the costs found in Tables 2 and 3 for the same schedule type.

6.2.1 Demand

The e-grocery market is experiencing huge annual growth rates. Therefore, it is interesting to study the impact of an increase in demand on the resulting time slot schedules and their performance. We consider a scenario in which the service requirements remain unchanged, but every zip code is experiencing a 30% increase in demand. The results can be found in Tables 4 and 5. (We omit time slot schedule ALBERT from these and further experiments as it was not designed for these environments.)

Table 4: Demand Increase - Morning

	# veh.	time	distance	cost	Δ cost
ALL	6.6	1734	522	123	23.00 %
RAND-ASG	5.9	1677	485	116	23.40 %
CA	5.5	1618	451	110	22.22 %
IP	5.8	1664	465	114	25.27 %
NO-SLOTS	5.0	1511	386	101	23.17 %

Table 5: Demand Increase - Afternoon

	# veh.	time	distance	cost	Δ cost
ALL	6.7	1665	519	120	22.45%
RAND-ASG	6.2	1607	485	114	21.28%
CA	5.9	1570	467	110	23.60%
IP	6.0	1574	462	111	24.72%
NO-SLOTS	4.8	1459	386	98	22.50%

We observe a cost increase of a little over 20% for a demand increase of 30%, which demonstrates, as expected, that there are economies of scale. This supports the importance of growth in order for a company to become more profitable.

6.2.2 Vehicle Capacity

An important consideration when setting up any delivery operation is the vehicle fleet and mix. As Albert.nl wants to operate a homogeneous fleet of vehicles, we focus on the impact of vehicle capacity. We consider two scenarios. One in which larger vehicles are used, i.e., the vehicle capacity is increased by 25%, and one in which smaller vehicles are used, i.e., the vehicle capacity is decreased by 25%. The results can be found in Tables 6 and 7.

Not surprisingly, a capacity reduction results in a cost increase whereas a capacity expansion results in a cost decrease. What is interesting, however, is that the cost penalty of a capacity reduction is much stronger than the cost benefit of additional capacity. This suggests a decreasing marginal value of capacity.

Table 6: Vehicle Capacity Change - Morning

		# veh.	time	distance	cost	Δ cost
-25%	ALL	5.7	1433.4	456	103	3.00%
	RAND-ASG	5.4	1402.1	433	100	6.38%
	CA	5.3	1414.8	414	99	10.00%
	IP	5.3	1435.3	428	101	10.99%
	NO-SLOTS	5.2	1282.4	370	91	10.98%
+25%	ALL	5.3	1394.9	440	100	0.00%
	RAND-ASG	4.7	1342.1	407	94	0.00%
	CA	4.1	1287.6	371	87	-3.33%
	IP	4.5	1328.5	394	91	0.00%
	NO-SLOTS	3.5	1183.7	311	78	-4.88%

Table 7: Vehicle Capacity Change - Afternoon

		# veh.	time	distance	cost	Δ cost
-25%	ALL	5.5	1355	444	99	1.02%
	RAND-ASG	5.3	1329	428	96	2.13%
	CA	4.9	1319	418	94	5.62%
	IP	5.1	1324	416	95	6.74%
	NO-SLOTS	4.7	1220	361	86	7.50%
+25%	ALL	5.4	1344	437	97	-1.02%
	RAND-ASG	5.0	1303	412	93	-1.06%
	CA	4.5	1271	401	89	0.00%
	IP	4.7	1256	383	89	0.00%
	NO-SLOTS	3.8	1165	327	79	-1.25%

To understand this, note that the physical vehicle capacity is only one of the factors that limits the number of orders that a vehicle can deliver. In particular, recall from our models that time constraints, based on the length of a time slot and of a shift, form another constraining factor. The above results suggest that these time constraints become more relevant with increasing vehicle capacity.

The extent to which vehicle capacity is binding depends on the shift and on the service level. The optimization-based time slot schedules and the NO-SLOT time slot schedule appear to be more sensitive to capacity changes because they can better exploit capacity, thus achieving a higher utilization. Moreover, the morning shift seems to be more sensitive to a reduction in vehicle capacity (10% versus 7%), most likely because it covers a period of 6 hours, as compared to 5 hours in the afternoon, whereby time constraints are less tight.

6.2.3 Service Level

A key challenge for e-grocers is to find a proper trade-off between service level and delivery costs. A higher service level typically comes at the expense of higher delivery costs. We clearly see this when comparing the

simulations results for the NO-SLOTS schedules with those for the schedules that use the service requirements of Albert.nl and with those of the ALL slots schedules. We now investigate this issue further by varying the imposed service requirements.

We consider two scenarios. In the first, we increase the required service level by offering more time slots, more specifically by offering one additional time slot in each zip code except in those zip codes where all time slots are already offered. In the second, we reduce the service level by offering fewer time slots, more specifically by offering one fewer time slot in each zip code except in those zip codes where only a single time slot is currently offered. Note that in the latter scenario, the changes are minor as most zip codes require only a single time slot to be offered in the base case.

Note also that these changes do not affect the ALL and NO-SLOT schedules since they do not use the service requirements. We only recall their performance as a point of reference. The results can be found in Tables 8 and 9.

Table 8: Service Level Adjustments - Morning

		# veh.	time	distance	cost	Δ cost
reduce service	ALL	5.3	1399	443	100	0.00%
	RAND-ASG	4.9	1376	412	96	2.13%
	CA	4.4	1325	387	91	1.11%
	IP	4.5	1341	402	92	1.10%
	NO-SLOTS	3.9	1210	329	82	0.00%
increase service	ALL	5.3	1399	443	100	0.00%
	RAND-ASG	5.0	1418	425	99	5.32%
	CA	5.1	1384	416	97	7.78%
	IP	4.9	1434	432	99	8.79%
	NO-SLOTS	3.9	1210	329	82	0.00%

Table 9: Service Level Adjustments - Afternoon

		# veh.	time	distance	cost	Δ cost
reduce service	ALL	5.4	1352	444	98	0.00%
	RAND-ASG	5.1	1318	412	94	0.00%
	CA	4.5	1265	397	89	0.00%
	IP	4.5	1274	387	89	0.00%
	NO-SLOTS	3.9	1171	329	80	0.00%
increase service	ALL	5.4	1352	444	98	0.00%
	RAND-ASG	5.1	1385	439	98	4.26%
	CA	5.0	1385	443	98	10.11%
	IP	5.1	1342	412	95	6.74%
	NO-SLOTS	3.9	1171	329	80	0.00%

As expected, the impact on cost of increasing the service level is much higher than the impact of reducing

the service level, as far fewer changes were made to the service level requirements. What may be more surprising is that for the morning shift a reduction of the service level actually leads to a cost increase! This seems counterintuitive, because one would expect to gain operational efficiency from a decrease in customer service. However, we have to realize that we only reduced the number of time slots offered in zip codes in which we used to offer more than one time slot. These zip codes tend to be urban areas with a relatively high demand density. A reduction in the number of time slots offered forces demand to be concentrated in fewer time slots, which results in a less balanced workload and thereby in a lower capacity utilization.

This example highlights two opposing effects. Capacity utilization benefits from a well-balanced workload and therefore from spreading demand over multiple time slots. On the other hand, however, the distance between customers decreases when clustering demand in fewer slots. The interplay between both of these effects results in complex trade-offs, which are very hard to make without any analytical support tools.

In both the morning and afternoon shifts, we do see increases in cost associated with improving the service level. This is not surprising, but shows a valuable aspect of developing such models. They reveal the cost associated with such changes, so companies can make better decisions with regard to deciding the level of service to offer to their customers.

6.2.4 Time Slot Template

The set of time slots that may be offered, known as the time slot template, has overlapping time slots in both shifts (see Figure 4). A time slot template with overlapping time slots may be attractive for both the delivery company and the customer. Overlapping slots provide more routing flexibility to the company than shorter non-overlapping slots. At the same time, they provide a higher service to the customer than a single overall slot. Note that from a modeling perspective the overlap is challenging.

We investigate the impact of overlapping time slots by analyzing alternative time slot templates. We consider two such time slot templates. The first uses time slots with smaller widths that do not overlap, i.e., 8:00-9:20, 9:20-10:40, 10:40-12:00, 12:00-14:00 for the morning and 16:00-18:00, 18:00-19:30, 19:30-21:00 for the afternoon. The second uses different overlapping 2-hour time slots, i.e. 8:00-10:00, 9:30-11:30, 11:00-13:00, 12:00-14:00 for the morning and 16:00-18:00, 17:30-19:30, 19:00-21:00 for the afternoon. The results for these different time slot templates can be found in Tables 10 and 11. Note that the different templates do not affect the NO-SLOTS scenario, which is just reported as a benchmark.

As expected, we see that the use of non-overlapping time slots with smaller widths translates in increased costs, especially in the morning shift. The results for the alternative overlap template are less conclusive. In general, the cost effects appear to be smaller than for removing the overlap. Note that the total amount of overlap, the number of hours which fall into more than one slot, is the same as in the original templates,

Table 10: Time Slot Template Changes - Morning

		# veh.	time	distance	cost	
no overlap	ALL	5.7	1469	480	106	6.00%
	RAND-ASG	5.2	1434	446	101	7.45%
	CA	4.5	1430	401	96	6.67%
	IP	4.6	1394	416	96	5.49%
	NO-SLOTS	4.0	1218	330	82	0.00%
alternative overlap	ALL	5.1	1391	435	98	-2.00 %
	RAND-ASG	4.7	1353	409	94	0.00%
	CA	4.4	1325	391	91	1.11%
	IP	4.7	1350	408	94	3.30%
	NO-SLOTS	4.0	1218	330	82	0.00%

Table 11: Time Slot Template Changes - Afternoon

		# veh.	time	distance	cost	
no overlap	ALL	6.0	1391	467	103	5.10%
	RAND-ASG	5.3	1339	434	97	3.19%
	CA	4.7	1302	406	92	3.37%
	IP	4.8	1291	403	92	3.37%
	NO-SLOTS	3.9	1171	329	80	0.00%
alternative overlap	ALL	5.4	1351	440	98	0.00%
	RAND-ASG	4.8	1293	407	92	-2.13%
	CA	4.8	1270	403	90	1.12%
	IP	4.6	1279	399	90	1.12%
	NO-SLOTS	3.9	1171	329	80	0.00%

however distributed differently.

The overlap was a difficult issue in terms of modeling for both CA and IP. The continuous approximation model heuristically adjusts the individual time slot lengths based on the order volumes and virtually removes the overlap. Conceptually, the IP model deals with the issue in a more accurate way by taking into account the exact timing of a zip code delivery. However, this comes at the cost of an increased problem size and it introduces symmetry in the model. But, we see that the overlap does translate to lower cost solutions, so it is important to consider it.

6.2.5 Profit maximization

All of the experiments so far have focused entirely on cost. However, Albert.nl charges for deliveries and the delivery fees vary for the different time slots. The delivery fees charged by Albert.nl for the time slots in the Monday morning shift are €7.95, €7.95, €6.95, and €5.95, and for the time slots in the Tuesday afternoon shift are €4.95, €6.95, and €6.95. Delivery fees are an important demand management mechanism for home delivery services. Differentiated delivery fees can counterbalance popularity differences between time slots

and smoothen demand over the time slots. This is vital to effectively utilize the delivery vehicles during a shift and the warehouse capacity during the week. At the same time, the delivery fees directly contribute to the company’s bottom line.

It is not difficult to incorporate delivery fees into our optimization models. The reason why we have not done so in our computational experiments up to now is that the available service constraints were not designed to account for differences in delivery fees. Incorporating delivery fees into the optimization models may lead to fewer zip codes receiving the lower priced time slots, which may be undesirable from a marketing perspective. This would call for defining different service requirements. While the interaction between pricing and slotting is certainly a relevant issue for future research, it surpasses the scope of this study.

In order to gain some initial insight into the potential of such a broader demand management approach, we simulated a profit maximization version of our model. To this end, we included the delivery fees plus the average product margin per order into the objective function.

The results can be found in Tables 12 and 13. We normalized the profit figures by setting the profit of ALL in the morning case to 100. For all schedules we report the relative profit increase compared to ALL. For the optimization based schedules we also report the relative profit increase compared to the cost-minimization version. As the routing software used in our simulation experiments does not allow for profit maximization, we do not present a NO-SLOTS lower bound here.

Table 12: Profit Maximization - Morning

	# veh.	time	distance	profit	Δ profit ALL	Δ profit cost min.
ALL	5.3	1399	443	100		
RAND-ASG	4.8	1348	408	115	15%	
ALBERT	4.5	1370	390	112	12%	
CA	4.3	1298	376	131	31%	5%
IP	4.6	1340	400	120	20%	6%

Table 13: Profit Maximization - Afternoon

	# veh.	time	distance	profit	Δ profit ALL	Δ profit cost min.
ALL	5.4	1352	444	55		
RAND-ASG	5.1	1310	417	64	17%	
ALBERT	4.6	1266	400	72	31%	
CA	4.5	1251	393	76	37%	4%
IP	4.7	1254	392	79	43%	6%

We observe that the relative profit differences between the solutions are larger than the cost differences

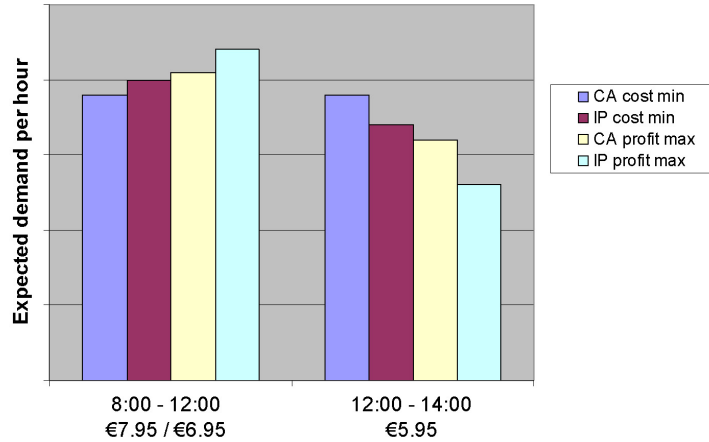


Figure 7: Demand Profile Morning

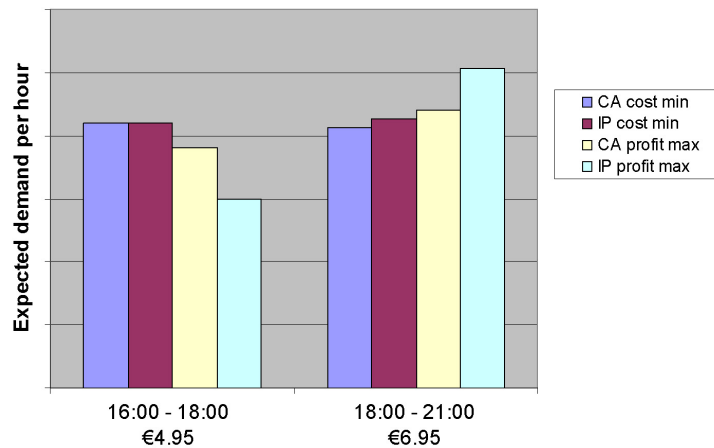


Figure 8: Demand Profile Afternoon

in the original scenarios. This is partly due to the fact that the absolute profit figures are smaller than the absolute cost figures. In addition, however, the relatively large differences of up to €2 between the delivery fees create additional room for optimization, which is exploited by the CA and IP methods.

When we compare profit maximization and cost minimization we observe profit increases of around 5%. In the corresponding solutions, we observe a shift of demand (i.e. open zip codes) to time slots with the higher delivery fees. Figures 7 and 8 illustrate this affect by displaying the expected demand per time during the shift for the different schedules. (Overlapping slots are grouped together.) The differentiated delivery fees introduce another factor in the trade-off between demand smoothing and clustering. Clustering demand now potentially yields the additional benefit of higher revenues from delivery fees. This comes at the expense of a lower capacity utilization and thus potentially more required vehicles. In the morning shift we observe

profit improvements without an increase in the number of required vehicles. This may relate to the longer shift length which make time less of a constraining factor. For the afternoon, we observe that the slightly increased number of vehicles in the IP solution is offset by the higher delivery fee revenues.

However, the above comments regarding the validity of our model for the profit maximizing case should be taken into account when interpreting these results. We therefore recommend a broader analysis of appropriate service requirements as an area for future research.

7 Conclusions

In this paper, we introduce the time slot management problem (TSMP), a novel tactical planning problem which is relevant to many businesses that offer attended home delivery, especially e-grocery. The problem entails the decision of which time slots to offer in each zip code so as to minimize expected delivery costs. The use of time slots gives rise to complex interactions between marketing and operational considerations and requires sophisticated decision support tools.

This paper presents two fully-automated approaches that are capable of producing high-quality solutions within a reasonable amount of time. The two approaches model the vehicle routing component of the TSMP in different ways. The IP approach explicitly models the vehicle routing decisions, while the continuous approximation approach estimates route costs per zip code based on "local" information. Our numerical results indicate that using service requirements together with our optimization methods yield substantial savings over simply offering all time slots.

Several complex interactions govern the time slot schedule design, such as the trade-off between demand clustering and demand smoothing. The advantage of demand clustering is that it minimizes the distances between successive stops, and thus travel costs. The disadvantage is the potential underutilization of the vehicle capacity, and a corresponding increase in vehicle costs. Our numerical experiments suggest that the time slot limits and the vehicle restrictions are critical components in this trade-off.

There remain several interesting directions for future research. We see the joint optimization of the service requirements and the time slot schedule as one of the most challenging. This will not only require a more complete understanding of customer behavior, but also more sophisticated solution approaches. Furthermore, we see great potential for additional research on the real-time management of demand and the use of incentives in attended home delivery.

References

- N. Agatz, A.M. Campbell, M. Fleischmann, and M. Savelsbergh. Challenges and opportunities in attended home delivery. In B. Golden, S. Raghavan, and E. Wasil, editors, *To Appear in The Vehicle Routing Problem: Latest Advances and New Challenges*. 2007.
- N.A.H Agatz, M. Fleischmann, and J.A.E.E. van Nunen. E-fulfillment and multi-channel distribution - a review. *European Journal of Operational Research*, 187(2), 2008.
- R. Bent and P. Van Hentenryck. Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, 52(6):977–987, 2004.
- C. Berning, S. Ernst, and N. H. Hooker. Are e-grocers serving the right markets? *Choices*, 20(1):77–80, 2005.
- I. Braysy and M. Gendreau. Vehicle routing problem with time windows, part i: Route construction and local search algorithms. *Transportation Science*, 39(1):104–118, 2005a.
- I. Braysy and M. Gendreau. Vehicle routing problem with time windows, part ii: Metaheuristics. *Transportation Science*, 39(1):119–139, 2005b.
- A. Campbell and M. Savelsbergh. Decision support for consumer direct grocery initiatives. *Transportation Science*, 39(3):313–327, 2005a.
- A. Campbell and M. Savelsbergh. Incentive schemes for attended home delivery services. *Transportation Science*, 40(3):327–341, 2005b.
- C.F. Daganzo. *Logistics Systems Analysis*. Springer, Berlin Heidelberg, fourth edition, 2005.
- C.F. Daganzo. Modelling distribution problems with time windows:part 1. *Transportacion Science*, 21(3):171–179, 1987a.
- C.F. Daganzo. Modelling distribution problems with time windows. part 2: Two customer types. *Transportacion Science*, 21(3):180–187, 1987b.
- M. Dror, G. Laporte, and P. Trudeau. Vehicle-routing with stochastic demands - properties and solution frameworks. *Transportation Science*, 23(3):166–176, 1989.
- P. Francis and K. Smilowitz. Modeling techniques for periodic vehicle routing problems. *Transportation Research Part B: Methodological*, 40(10):872–884, 2006.

- P. Francis, K. Smilowitz, and M. Tzur. The period vehicle routing problem with service choice. *Transportation Science*, 40(4):439–454, 2006.
- M. Gendreau and J. Potvin. Dynamic vehicle routing and dispatching. In T. Crainic and G. Laporte, editors, *Fleet Management and Logistics*, pages 115–126. Kluwer, 1998.
- L. M. Hvattum, A. Lokketangen, and G. Laporte. Solving a dynamic and stochastic vehicle routing problem with a sample scenario hedging heuristic. *Transportation Science*, 40(4):421–438, 2006.
- A. S. Kenyon and D. P. Morton. Stochastic vehicle routing with random travel times. *Transportation Science*, 37(1):69–82, 2003.
- A. Langevin, P. Mbaraga, and J. F. Campbell. Continuous approximation models in freight distribution: An overview. *Transportation Research Part B-Methodological*, 30(3):163–188, 1996.
- I.I. Lin and H.S. Mahmassani. Can online grocers deliver?: Some logistics considerations. *Transportation Research Record*, 1817:17–24, 2002.
- M. Punakivi and J. Saranen. Identifying the success factors in e-grocery home delivery. *International Journal of Retail & Distribution Management*, 29(4), 2001.

Publications in the Report Series Research * in Management

ERIM Research Program: "Business Processes, Logistics and Information Systems"

2008

An Evolutionary Framework for Determining Heterogeneous Strategies in Multi-Agent Marketplaces

Alexander Babanov, Wolfgang Ketter and Maria Gini

ERS-2008-002-LIS

<http://hdl.handle.net/1765/10972>

Choosing between Auctions and Negotiations in Online B2B Markets for IT Services: The Effect of Prior Relationships and Performance

Uladzimir Radkevitch, Eric van Heck and Otto Koppius

ERS-2008-004-LIS

<http://hdl.handle.net/1765/11288>

Key Issues in Expansion of End-User Mobile Communication in China

Sunanda Sangwan, Guan Chong and Louis-Francois Pau

ERS-2008-011-LIS

<http://hdl.handle.net/1765/11762>

Some Comments on the Question Whether Co-Occurrence Data Should Be Normalized

Ludo Waltman and Nees Jan van Eck

ERS-2008-014-LIS

<http://hdl.handle.net/1765/11810>

Bibliometric Mapping of the Computational Intelligence Field

Nees Jan van Eck and Ludo Waltman

ERS-2008-015-LIS

<http://hdl.handle.net/1765/11811>

Demand Management Opportunities in E-fulfillment: What Internet Retailers Can Learn from Revenue Management

Niels Agatz, Ann Campbell, Moritz Fleischmann, Jo van Nunen and Martin Savelsbergh

ERS-2008-021-LIS

<http://hdl.handle.net/1765/12244>

Time Slot Management in Attended Home Delivery

Niels Agatz, Ann Campbell, Moritz Fleischmann and Martin Savelsbergh

ERS-2008-022-LIS

<http://hdl.handle.net/1765/12245>

Mobile Service Affordability for the Needy, Addiction, and ICT Policy Implications

L-F Pau

ERS-2008-023-LIS

<http://hdl.handle.net/1765/12246>

* A complete overview of the ERIM Report Series Research in Management:

<https://ep.eur.nl/handle/1765/1>

ERIM Research Programs:

LIS Business Processes, Logistics and Information Systems

ORG Organizing for Performance

MKT Marketing

F&A Finance and Accounting

STR Strategy and Entrepreneurship