Coversheet

This is the accepted manuscript (post-print version) of the article.
Contentwise, the post-print version is identical to the final published version, but there may be differences in typography and layout.

How to cite this publication
Please cite the final published version:

Publication metadata

A Variable Neighborhood Search for the Multi-Period Collection of Recyclable Materials

Maria Elbek[a], Sanne Wøhlk[a,*]

[a]*CORAL - Cluster for OR and Logistics, Department of Economics and Business, School of Business and Social Sciences, Aarhus University, Fuglesangs Allé 4, 8210 Aarhus V, Denmark.*

Abstract

We consider an approach for scheduling the multi-period collection of recyclable materials. Citizens can deposit glass and paper for recycling in small cubes located at several collection points. The cubes are emptied by a vehicle that carries two containers and the material is transported to two treatment facilities. We investigate how the scheduling of emptying and transportation should be done in order to minimize the operation cost, while providing a high service level and ensuring that capacity constraints are not violated. We develop a heuristic solution method for solving the daily planning problem with uncertain accretion rate for materials by considering a rolling time horizon of a few days. We apply a construction heuristic in the first period and re-optimize the solution every subsequent period with a variable neighborhood search. Computational experiments are conducted on real life data.

Keywords: Inventory routing problem, multi-period routing, multi-compartment vehicle, rolling time horizon, waste management

1. Introduction

Modern society generates ever-increasing amounts of waste and for environmental and sustainability reasons the public authorities naturally focus on increasing the amount of recycling. Recycling and recycling costs have thus become issues of vital importance for both society and environment.

Research in collection and handling of waste and reusable material is therefore a matter of large practical relevance. Beullens et al. (2010) give an overview of transportation of waste and reusable material from an Operations Research point of view. They argue that research in reverse logistics systems is very inadequate and sparse compared to that of (normal) forward logistics, and that more research needs to be conducted with the goal of making reverse logistics systems efficient.

Our project focuses on the transportation related to collection of two recyclable materials generated by households. The aim of this project is to find improved scheduling procedures for collection of materials for recycling and the related transportation to the treatment facilities with the goal of minimizing cost while ensuring efficient flow and high service levels.

From a theoretical point of view, the problem under consideration is a multi-period inventory routing problem with two commodities and the addition of transportation to treatment facilities.

To the best of our knowledge, the special capacity constraints related to the problem have not been discussed in existing literature. Bogh et al. (2014) do, however, study another aspect of the same problem.

In the following, we shortly describe the real life case used for our study of the problem. An illustration of the case setup is provided in Figure 1 and a more thorough description of the details is given in Section 3.
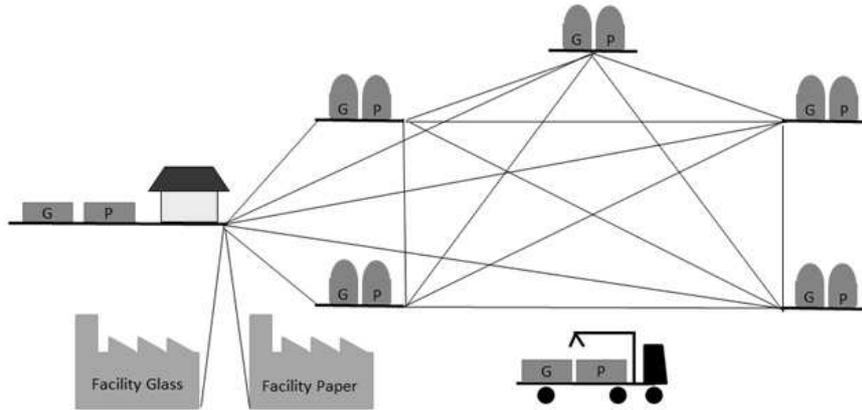

Figure 1: Setup of the case. Illustration of principle with only five locations, each with two cubes.

So-called *cubes* which can hold approximately $1.6\,\text{m}^3$ of material, are scattered around at locations in Denmark and are used for collection of paper and glass for recycling. This paper focuses on the case in Djursland where 211 locations are available, holding a total of 240 cubes for glass and 198 cubes for paper. The majority of the locations have one cube for each type of material, but some of them have up to six cubes located together. The quantity of material deposited by the citizens in the cubes on a given day is unknown. Therefore, the fill level of a cube is not known until the location is visited. The need for service varies between every second week and every second month depending on location.

The collection is performed with a vehicle carrying two $36\,\text{m}^3$ containers, one for glass and one for paper. Hence glass and paper are collected from the cubes simultaneously and the vehicle can be viewed as a multi-compartment vehicle. The vehicle is equipped with a crane that makes it possible to lift up the cube and remove the bottom of the cube. In this way the material in the cube can be emptied into the proper container before the cube is restored to its place. It takes 3 minutes to empty the first cube, 2 minutes for the first cube of the other type, and 1 minute for any additional cube.

The vehicle starts and ends each collection route at a depot, where an extra container for each type of material is located. When the vehicle returns to the depot, the crane unloads the paper from the vehicle to the extra paper container, while, due to shattering, this is not possible for glass. Instead, when the glass container of the vehicle is sufficiently full, it is swapped with the extra glass container.

Once two containers of the same type are filled, they are transported to the proper treatment facility, where the materials will be processed for recycling. The time for offloading at the treatment facility is 15 minutes for two containers. Due to the limited amount of material, such trips to the treatment facilities are not needed on a daily basis. Therefore, the available capacity in the glass container of the vehicle each day depends heavily on the work carried out during the previous days.

2

Likewise, once the extra paper container is full, the capacity available in the paper container of the vehicle also depends on the work completed during the previous days.

The collection of waste and recyclable material in Djursland is handled by a semi public company, Reno Djurs. Their responsibility is to provide a high service level for the citizens, which is equivalent to ensuring that the cubes are emptied before being overfilled. In practice, the work is outsourced to a third party logistics provider, but in this paper we consider the problem as if they provide the service themselves. They seek to minimize their cost, which is composed of cost of driving on the routes and to the treatment facilities and a cost of service for the cubes and at the treatment facilities.

In this paper, we define the problem based on the Reno Djurs case. The aim is to create a multi-period schedule for combined collection of paper and glass that minimizes the total cost, while ensuring that the cubes are not overfilled and that capacity constraints of the containers are not violated. We treat the problem as an inventory routing problem where two commodities must be collected jointly but handled separately and transported to separate treatment facilities. The daily filling at each location is stochastic and will be assumed to follow a normal distribution. The service is carried out by a single multi-compartment vehicle, where the available container capacity can vary from day to day.

The remainder of the paper is organized as follows. In Section 2, we present the related literature and in Section 3, we provide a description of the problem and the related notation. We give a detailed description of our solution approach in Section 4 and provide the results of our computational experiments in Section 5. Conclusions and directions for further research are given in Section 6. Finally in the appendix, we offer a mathematical model of a deterministic version of the problem, i.e. a model based on the assumption that the daily input is known for all cubes.

2. Related Literature

A number of studies on waste management are available in the literature. According to Golden et al. (2002) the waste collection business can be divided into three major areas: commercial, residential, and roll-on-roll-off. The collection of residential waste is often dealt with as an arc routing problem because of the large number of households that typically have to be visited and the small distances between them. Collection of recyclable materials from cubes is a special case of residential waste management since the waste is generated by households. However, the materials are not collected directly from the households, but from separate locations and it is most naturally modeled as a node routing problem. In the following we present a brief overview of node routing problems with characteristics that are similar to our problem.

Within the field of waste management, Angelelli and Speranza (2002) consider two case studies. They suggest a deterministic model for estimating the cost of three waste collection systems and present a solution method based on the Periodic Vehicle Routing Problem (PVRP) with intermediate facilities. Since they did not have access to data concerning distance between every given collection point, they group the locations that are geographically close to each other, hold the same type of waste, have the same daily accretion rate of waste, and share the same visiting schedules. They implement a Tabu Search heuristic for solving the problem.

Kim et al. (2006) address a real life waste collection vehicle routing problem with driver's rest periods, multiple final destinations for each type of waste, and time windows associated with commercial customers, final destinations, and the depot. Since the commercial customers do not change the frequency of service very often, the weekly service schedule is fairly static and

assumed to be predetermined. Each vehicle can, and typically does, make multiple trips to the final destinations per day. They develop an algorithm for a capacitated clustering-based waste collection vehicle routing problem with time windows.

Real world constraints such as drivers' rest periods, multiple facilities, and time windows are also studied by Benjamin and Beasley (2010). However, they only consider the problem for a single period. They solve the problem with three metaheuristics: Tabu Search, Variable Neighborhood Search, and Variable Neighborhood Tabu Search. In the latter heuristic, the variable neighborhood is searched via Tabu Search. The VNS provides the best results.

Teixeira et al. (2004) study vehicle route planning for a separate collection of three types of materials. Glass, paper and plastic/metal must be collected separately from locations scattered around the country by several vehicles. The vehicle capacity is fixed, but differs according to material type. They create static routes for every day of the month. These routes are repeated every month in a PVRP framework. The quantity of materials deposited daily in each cube is assumed to be constant. Part of the construction heuristic developed is inspired by Golden et al. (1984). That is, for each day in the planning period, a set of candidate locations for collection is defined based on the urgency of collection and the nearest neighbor. A route for each material type, work shift, and geographical zone is defined for the set of locations.

Bogh et al. (2014) study the same real world collection problem as presented in this paper, but they focus on the financial sustainability of the system. They discuss how outsourcing the planning and transportation of recyclable materials may result in conflicts of interest for the parties involved and how this leads to sub-optimal solutions from an overall sustainability perspective. Their analysis is based on simulations where the problem is solved with a number of different objective functions.

Paper, glass, and plastic/metal are not the only materials that are collected for reuse. In a recent paper, Aksen et al. (2014) study the collection of waste vegetable oil to be used in a production process for biodiesel. They model this problem as a periodic inventory routing problem where the service of each location is optional and where the requirement for material is flexible because the production process can be supplemented by virgin oil. The cost is made up of routing cost, inventory cost, and purchasing cost. They present a large neighborhood search algorithm for solving the problem and use a large number of moves to select the locations to service and modify the collection plan. Computational experiments are performed on real life data with up to 100 locations and a weekly schedule.

The literature also provides studies where multi-compartment vehicles are used for delivery or collection. Recently, Coelho and Laporte (2015) have provided classification and models for several general multi-compartment delivery problems. They provide an in-depth discussion and modeling of the decision regrading the allocation of products to vehicle compartments. Finally, they show extensive computational experiments based on branch-and-cut for both single and multi period cases of the problem. Muyldermans and Pang (2010) also study a multi-compartment vehicle routing problem, and focus on waste collection from several customers in a single period. They investigate if a company can benefit from co-collection compared to separate collection of different types of waste from the customers. In many cases, co-collection is found to be better than separate collection.

From a theoretical point of view, the problem studied in this paper is partly similar to the Inventory Routing Problems (IRP). In classic IRP, the supplier can control the inventory of a set of retailers. The supplier decides when to visit a retailer, how many products to deliver to

the retailer, and in which order the retailers are visited on a given day within the time horizon. The aim is to select a distribution policy that minimizes the total cost, often composed of routing cost, inventory cost, and stock-out cost, and to ensure that capacity constraints are not violated. Andersson et al. (2010) and Coelho et al. (2014b) provide detailed literature surveys and classifications for IRP.

Recently, Bertazzi et al. (2013) and Coelho et al. (2014a) have solved a variant of the IRP in a rolling horizon framework. The demands of the retailers are generated using a normal distribution and the retailers are controlled by an order-up-to level policy. Bertazzi et al. (2013) use the mean of the probabilistic information as a forecast for future demands, while Coelho et al. (2014a) take historical data into account for the demand forecast. Coelho et al. (2014a) compare two strategies: A reactive policy where the deliveries are scheduled one day ahead and a proactive policy that takes greater advantage of demand forecasts and schedules deliveries several days ahead. The proactive policy outperforms the reactive policy with respect to routing costs because vehicles can avoid consecutive and costly visits to the same geographical area when forecasts are used.

A general property of waste collection problems is that the schedule is periodic or static and that multiple trips to the facilities may be necessary each day. In the available studies, the different types of waste are typically collected separately or there is only one type of waste. An exception is Muyldermans and Pang (2010). The quantity of waste deposited daily is often assumed to be constant or completely known. However, for our problem, the above-mentioned approaches for waste collection cannot be adopted since we collect different types of materials simultaneously over a non-periodic time horizon. Furthermore, the quantity of material deposited in the cubes each day is unknown and fluctuates. Compared to the IRP, our problem is complicated by non-static capacity available in containers and by the transportation to facilities, hence the solution methods cannot be directly applied to our problem. Recently, Beullens et al. (2010) have pointed out that there is a lack of solution models for multi-period routing problems with combined collection in multi-compartment vehicles. In this paper we will therefore contribute to the existing literature by presenting an algorithm for solving the multi-period collection problem with special capacity constraints in a multi-compartment vehicle. The aim is to minimize the total cost while ensuring that capacity constraints in cubes and containers are not violated.

3. Problem Description

In this section, we describe the problem under consideration in detail and introduce the notation. For convenience, Table 1 gives an overview of the notation used in the model.

Let $\mathcal{G}$ and $\mathcal{P}$ be the set of glass cubes and paper cubes, respectively, and let $\mathcal{M} = \mathcal{G} \cup \mathcal{P}$. Let $\mathcal{L}$ be the set of locations. We use the term "family of cubes", denoted by $\mathcal{G}(l)$ and $\mathcal{P}(l)$ to refer to the cluster of cubes for glass and paper, respectively, in location $l \in \mathcal{L}$. The collection is performed over a finite time horizon H. Let $t \in \mathcal{T} = \{1, \ldots, H\}$ be the discrete time instant.

In Figure 2, we illustrate the sequence of the events that occur at time $t \in \mathcal{T}$. The amount of material at each location is first estimated based on the information available about the daily filling (A). Next, any planned transportation to the treatment facilities is performed (B) and then any planned collection route is executed (C). Finally, in the last part of the time instant, materials are deposited in the cubes (D).

The quantity of material deposited in cube i at time t is given by r_{it} and is uncertain. The quantity of material collected from cube i at time t is given by x_{it}. F_{it} represents the quantity of

Table 1: Overview of Notation

Index Sets and Indices	
$\mathcal{L}$	The set of all locations
l	A location in $\mathcal{L}$
$\mathcal{M}$	The set of all cubes
$\mathcal{G}, \mathcal{P}$	The set of glass (paper) cubes
$\mathcal{G}(l), \mathcal{P}(l)$	The family of glass (paper) cubes at location $l \in \mathcal{L}$
$\mathcal{T}, \mathcal{T}'$	The set of the discrete time instants excluding (including) time zero
t	A discrete time instant in $\mathcal{T}$ or $\mathcal{T}'$

Scalars and Parameters	
H	The finite time horizon
C	The capacity of an empty container
U_i	The capacity of cube $i \in M$
r_{it}	The quantity of material deposited in cube $i \in M$ at time $t \in T'$
c_{ij}	The transportation cost between $i \in M$ and $j \in \mathcal{M} \cup \{0\}$
$k(\mathrm{g}), k(\mathrm{p})$	The transportation cost of a round trip to the treatment facility for glass (paper)
w_{ij}	The service cost for emptying cube $i \in M$ after cube $j \in M$
$w(\mathrm{g}), w(\mathrm{p})$	The service cost at the treatment facility for glass (paper)

Variables	
$\eta_t(\mathrm{g})$	Binary. Swap of glass containers at time $t \in T$
$\eta_t(\mathrm{p})$	Binary. Visit to treatment facility for paper at time $t \in T$
y_{ij}^t	Route traversal $\forall i \in \mathcal{M}, j \in \mathcal{M} \cup \{0\}, j < i, t \in \mathcal{T}$
x_{it}	The quantity of material collected from cube $i \in M$ at time $t \in T'$
F_{it}	The quantity of material in cube $i \in M$ at time $t \in T'$
$C_t(\mathrm{g}), C_t(\mathrm{p})$	The capacity available in the container for glass (paper) at time $t \in T'$

material present in cube i at time t (at point (A) in Figure 2) and is given by

$$F_{it} = F_{i(t-1)} - x_{i(t-1)} + r_{i(t-1)} \quad \forall i \in \mathcal{M}, t \in \mathcal{T}, \tag{1}$$

where the starting quantity of material in cube i, F_{i0} is given for the first time instant, and $x_{i0} = r_{i0} = 0$. Cubes cannot be partially emptied, therefore

$$x_{it} = \begin{cases} F_{it} & \text{if cube } i \in \mathcal{M} \text{ is emptied at } t \in \mathcal{T}, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

The fill level, F_{it} in cube i may not exceed the capacity of the cube. Hence

$$0 \le F_{it} \le U_i \quad \forall i \in \mathcal{M}, t \in \mathcal{T}', \tag{3}$$

where U_i represents the capacity of cube $i \in \mathcal{M}$ and $\mathcal{T}' = \{0\} \cup \mathcal{T} \cup \{H+1\}$.

The quantities of glass and paper collected at time t may not exceed the capacity available in the container for glass $C_t(\mathrm{g})$ and paper $C_t(\mathrm{p})$, respectively, that is

$$\sum_{i \in \mathcal{G}} x_{it} \le C_t(\mathrm{g}) \text{ and } \sum_{i \in \mathcal{P}} x_{it} \le C_t(\mathrm{p}) \quad \forall t \in \mathcal{T}. \tag{4}$$
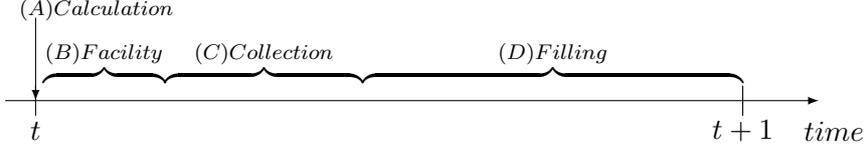
Figure 2: Timeline

The capacities $C_t(\mathrm{g})$ and $C_t(\mathrm{p})$ available in the containers on the vehicle vary over the time horizon and are highly dependent on the time when the materials are transported to the treatment facilities. If we transport the materials to the treatment facilities at time t, then the capacity available in the container will be equal to the capacity of an empty container C. For glass, an empty container can also be obtained by swapping to an empty container at the depot. But due to the possibility of simply offloading the paper, the capacity $C_t(\mathrm{p})$ available in the paper container will be equal to an empty container C as long as the extra paper container at the depot is not full. At other times, the capacities $C_t(\mathrm{g})$ and $C_t(\mathrm{p})$ available in the containers depend on the work completed during the previous days. This means that the capacity available at time t is given by the capacity available at time $t-1$ minus the total quantity of collected material at time $t-1$. For glass, the capacity $C_t(\mathrm{g})$ can be described as

$$
C_t(\mathrm{g}) = \begin{cases} C & \text{if we swap to an empty container at time } t, \\ C_{(t-1)}(\mathrm{g}) - \sum_{i \in \mathcal{G}} x_{i(t-1)} & t \in \mathcal{T} \quad \text{otherwise.} \end{cases} \tag{5}
$$

When the glass containers have been swapped twice, a trip to the treatment facility is required, which imposes a travel cost $k(\mathrm{g})$ and a service cost $w(\mathrm{g})$. We set the variable $\eta_t(\mathrm{g})$ to 1 if the glass containers are swapped at time $t \in \mathcal{T}$, and zero otherwise. Notice that we can swap to an empty container if there is one available at the depot or if we have visited the treatment facility at time t. But due to the possibility of offloading paper, the capacity $C_t(\mathrm{p})$ available in the container for collecting paper at time t can be described as

$$
C_t(\mathrm{p}) = \begin{cases} C & \text{if the container at the depot is not full at time } t, \\ C_{(t-1)}(\mathrm{p}) - \sum_{i \in \mathcal{P}} x_{i(t-1)} & t \in \mathcal{T} \quad \text{otherwise.} \end{cases} \tag{6}
$$

The variable for paper is slightly different than the one used for glass, as it controls the trips to the treatment facility rather than the swaps. We let $\eta_t(\mathrm{p})$ be 1 if the two paper containers are emptied at the treatment facility at time $t \in \mathcal{T}$, and zero otherwise. The travel and service costs of the trip are given by $k(\mathrm{p})$ and $w(\mathrm{p})$, respectively.

Every time we empty cube i after cube j, a service cost w_{ij} is applied, which is defined as

$$
w_{ij} = \begin{cases} 1 & \text{if cube } i \text{ and cube } j \text{ belong to the same location and contain the same type of material,} \\ 2 & \text{if cube } i \text{ and cube } j \text{ belong to the same location and contain different types of material,} \\ 3 & \text{otherwise.} \end{cases}
$$

Travel costs between cube i and j (i and the depot, if $j = 0$) are given by c_{ij} and the variable y_{ij}^t is used to represent the traveling at time $t \in \mathcal{T}$. We aim to minimize the total cost, which

7

Table 2: Overview of Additional Notation

$\mathcal{PP}$	The set of time instants in the short planning period, $\mathcal{PP} \subset \mathcal{T}$
$\mathcal{PP}^*$	$\mathcal{PP}$ plus an additional artificial day
$\mathcal{R}_t$	The route at time $t \in \mathcal{T}$
α, α_j	Thresholds for including a family of cubes in the planning, where j represents the classification of the family of cubes
β	A threshold determining which cubes to empty at a visit to a location
λ	A threshold for how full a family of cubes must at least be before it is considered for emptying
$\mu_l(\mathrm{g})$, $\mu_l(\mathrm{p})$	The mean of the daily accretion rate for the family of glass (paper) cubes at location $l \in \mathcal{L}$
$\sigma_l(\mathrm{g})$, $\sigma_l(\mathrm{p})$	The standard deviation of the daily accretion rate for the family of glass (paper) cubes at location $l \in \mathcal{L}$
$Z(\mathrm{g})$, $Z(\mathrm{p})$	The quantity of glass (paper) left over at location $l \in \mathcal{L}$ at the last visit
$Z_i(\mathrm{g})$, $Z_i(\mathrm{p})$	The quantity of glass (paper) left over in cube $i \in \mathcal{M}$ at the last visit
$Q(\mathrm{g})$, $Q(\mathrm{p})$	Future cost of visiting the treatment facility for glass (paper) as planned in the planning period
Q_{lt}	Future cost of visiting location $l \in \mathcal{L}$ at time $t \in \mathcal{T}$

consists of both cost for routes and cost of trips to the treatment facilities. We also include a service cost for the locations on the routes and at the treatment facilities. This service cost is the cost of emptying cubes on the locations and emptying containers at the treatment facilities, respectively. Formally, we seek to minimize

$$\sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{M} \cup \{0\}, j < i} \sum_{t \in \mathcal{T}} (c_{ij} + w_{ij}) y_{ij}^t + \tfrac{1}{2}(k(\mathrm{g}) + w(\mathrm{g})) \sum_{t \in \mathcal{T}} \eta_t(\mathrm{g}) + (k(\mathrm{p}) + w(\mathrm{p})) \sum_{t \in \mathcal{T}} \eta_t(\mathrm{p}) \quad (7)$$

We refer the reader to the appendix for a full mathematical model of the problem.

4. Solution Method

We present an algorithm for planning the collection of two recyclable materials using a vehicle with varying capacity and transporting them to their treatment facilities. The collection problem is NP-hard, since it includes the traveling salesman problem. Therefore – and because of the size of the problem, a heuristic approach is a natural choice. For convenience, Table 2 gives an overview of the additional main notation used in this section.

It is not reasonable to cluster the locations based on geography and similar daily accretion rates as we see massive fluctuations in the accretion rate of materials. But a classification of the families of cubes is made based on the accretion rate in order to distinguish between cubes that get filled fast and cubes that get filled slowly. The non-static accretion rate of materials is also a reason why it is not convenient to model the problem as a PVRP. We therefore develop an algorithm for solving the non-periodic collection problem. The solution approach is embedded into a rolling horizon framework, where we apply a construction heuristic once and reoptimize the solution in every subsequent period. Thereby we eliminate the need for daily handling of all locations.

As an attempt to steer the algorithm towards better solutions we introduce three thresholds. As we need only empty the semi-full or full cubes, we introduce β as a threshold for determining which cubes to empty. If a cube is less than β percent full, it will not be emptied. The values of the α-thresholds determine when a family of cubes is included in the planning, and the values of these thresholds are determined by the accretion rate classification. Finally, a family of cubes is only planned to be emptied if the estimated fill level is above the threshold λ, where $1 \geq \lambda \geq \beta \geq 0$.

4.1. Overview of the Algorithm

This section gives an overview of the algorithm used for solving the collection problem. A look-ahead approach is used to avoid costly and repeated visits to nearby locations as it is also done in Coelho et al. (2014a). Thus, a short planning period (PP) is defined in a rolling horizon framework. The length of PP is selected to 11 days, as according to historical data, the cube with the highest accretion rate must be emptied within 11 days to avoid overfilling. Computational experiments show no benefit of using a shorter PP. Furthermore, PP* is defined as PP plus an additional artificial day with special requirements. We use $\mathcal{PP}$ ($\mathcal{PP}^*$) to denote the set of days in PP (PP*). In a construction heuristic, families of cubes are classified based on their accretion rates and all locations are inserted into the time horizon. The solution is reoptimized every subsequent day by improving the collection in PP. The solution is improved by reorganizing locations between days in PP* with the purpose of minimizing the overall total cost. The transportation to the treatment facilities is scheduled simultaneously with planning the collection, and consequently adjustments of the available capacities in the containers are done every day. The structure of the algorithm is shown below. The steps in the algorithm will be described in greater details in the following sections.

Algorithm 1: Overview of the algorithm

Initialization;
Classification of families of cubes;
Insertion of all locations in routes in $\mathcal{T}$;
$t = 1$;
while $t \leq H$ **do**
    Improve the collection plan in $\mathcal{PP}$;
    Transport materials to the treatment facilities, if necessary;
    Execute the route at day t;
    Update capacities;
    Insert each visited location on a route at a day $\hat{t} > t$, $\hat{t} \in \mathcal{T}$;
    $t = t + 1$;
end
Check for overfilled cubes at the end of the time horizon;

4.2. Handling Unknown Information

The daily accretion rate of materials r_{it} in a cube is unknown for all $i \in \mathcal{M}$ and $t \in \mathcal{T}$. Like Bertazzi et al. (2013), Coelho et al. (2014a), and Bogh et al. (2014), we handled this unknown information by generating a forecast for the daily accretion rate as a normal distribution. The mean is estimated based on historical data from Reno Djurs. Reno Djurs further estimates the

daily accretion rate at a location to usually vary within $\pm 10\%$ of the mean of the location. It is therefore assumed that for each location, the daily accretion rate is normally distributed with the estimated mean and a standard deviation of 5% of the mean. For a location l we define $\mu_l(\mathrm{g})$ and $\mu_l(\mathrm{p})$ as the means and $\sigma_l(\mathrm{g})$ and $\sigma_l(\mathrm{p})$ as the standard deviations for the family of glass cubes and paper cubes, respectively. These distributions are used when planning visits to locations.

Below we explain how to find the day t^*, when the family of glass cubes at a location is expected to be α percent full. Let the day $\tilde{t} \in \mathcal{T}'$ represent the day of the last visit to location $l \in \mathcal{L}$. The day t^* is established by finding the largest integer t^*, that satisfies the following:

$$\mu_l(\mathrm{g}) \cdot \left(t^* - \tilde{t}\right) + 1.6449\sqrt{\sigma_l(\mathrm{g})^2 \cdot \left(t^* - \tilde{t}\right)} + Z(\mathrm{g}) \leq \sum_{i \in \mathcal{G}(l)} U_i \cdot \alpha, \tag{8}$$

where the number 1.6449 corresponds to the 95%–quantile of the normal distribution and $Z(\mathrm{g})$ is a known parameter, which indicates the quantity of glass left over at location l at day $\tilde{t}$. Often $Z(\mathrm{g})$ is zero, due to the fact the all glass cubes are emptied. However, $Z(\mathrm{g})$ is strictly positive if some glass cubes were less than β percent full at day $\tilde{t}$. If the threshold β is zero, all cubes are emptied at a visited location and thereby $Z(\mathrm{g})$ is zero. t^* is found in (8) by solving a quadratic equation. The resulting day t^* is the day, where the family of glass cubes $\mathcal{G}(l)$ at location l is considered approximately α percent full. A similar day t^* is identified for the family of paper cubes $\mathcal{P}(l)$. Notice that if α is equal to 100 percent, then day t^* is defined as the last possible day $t^\dagger \in \mathcal{T}$ for emptying the glass cubes at the location to avoid overfilling with a probability of 95%. In the planning process, day $t^\dagger$ is used as an upper bound indicating the latest possible time of emptying the family of cubes.

4.3. Classification of the Families of Cubes

All families of cubes are divided into three sets: red, yellow, and green, since they get filled with very different rates. Historical data shows that some families of cubes need emptying almost weekly while others only need emptying every second month. The red set contains the critical families of cubes that need to be emptied often, typically cubes in city areas. The yellow set contains the semi-critical families of cubes that do not need to be emptied as often as those in the red set. The green set contains the non-critical families of cubes that only need to be emptied a few times during the time horizon, typically cubes in rural areas. Thresholds α_{red}, α_{yellow}, and α_{green} are defined for each set and are used for the classification and insertion of the families of cubes.

This is how the classification is made for a family of glass cubes at a location: First, day $t^\dagger$ is found as the unknown t^* by solving (8) with α equal to one and $Z(\mathrm{g})$ equal to zero. In the same way, days t' and t'' are identified as the unknown t^* by solving (8) with α equal to α_{red} and α_{yellow}, respectively. If $t^\dagger - t' \leq |\mathcal{PP}|$, the family of glass cubes belongs to the red set. If $t^\dagger - t'' \leq |\mathcal{PP}|$, the family of glass cubes belongs to the yellow set. Otherwise, the family of glass cubes belongs to the green set. The procedure is repeated for the family of paper cubes at the same location. Notice that a location can have different set colors for the different families of cubes. The values of α_{green}, α_{yellow}, and α_{red}, with $1 \geq \alpha_{\mathrm{green}} \geq \alpha_{\mathrm{yellow}} \geq \alpha_{\mathrm{red}} \geq 0$, are determined from computational experience.

4.4. Insertion of Locations

When a location is inserted into $\mathcal{T}$, it is done in the following way. The day t' where the family of the glass cubes at location l is considered α_j percent full, is found by solving (8) where

j represents the classification of the family. Similarly, day t'' is identified for the family of paper cubes at location l. Thereby location l is inserted into the route on day $\hat{t} = \min\{t', t''\}$.

Once $\hat{t}$ has been selected, the levels of materials in the family of glass cubes and paper cubes are estimated for this day. If the estimated amount is at least $\beta \cdot \sum_{i \in \mathcal{G}(l)} U_i$, then we plan to empty the family of glass cubes. The same is valid for paper. Thus, we can plan to collect either one type of material or both at day $\hat{t}$.

To initialize, all locations are inserted into $\mathcal{T}$ by solving (8), where day $\tilde{t}$ is zero and $Z(\mathrm{g})$ and $Z(\mathrm{p})$ are the given start quantities of glass and paper at the location. Thereby all locations are in $\mathcal{T}$, but not necessarily in $\mathcal{PP}$ after the initialization. Hence, in the daily planning, only a subset of locations are considered.

4.5. Construction of Objective Function including Future Cost

To capture the long term effect on the overall total cost of the modifications made in the improvement, we use a refashioned objective function for evaluation of the collection plan in PP containing two terms: The direct cost and a future cost. The direct cost consists of routing cost, service costs for the locations and treatment facilities, and the transportation cost to the treatment facilities in PP*. The future cost consists of an estimated cost Q_{lt} of visiting a location $l \in \mathcal{L}$ at a given time $t \in \mathcal{T}$, and an estimated cost $Q(\mathrm{g})$ and $Q(\mathrm{p})$ of visiting the treatment facilities for glass and paper as planned. The intuition behind the development of a future cost is that visiting a location earlier than necessary can create the need for an additional visit of this location at the end of the time horizon. The same applies for the treatment facilities. Furthermore we know that if a location is placed far away from others, it can be difficult to include the location in a route without adding large routing cost. As we use a rolling horizon for the planning, this effect is not captured by routing, service, and the treatment facilities costs alone. Therefore Q_{lt}, $Q(\mathrm{g})$, and $Q(\mathrm{p})$ are used to model the future cost for locations and treatment facilities, respectively. The future cost for location $l \in \mathcal{L}$ on day $t \in \mathcal{T}$, $t \leq t^\dagger$, is computed as

$$Q_{lt} = \frac{t^\dagger - t}{t^\dagger - \tilde{t}} \cdot (\gamma_l + \phi_l), \tag{9}$$

where γ_l is the average routing cost from location l to its ten nearest neighbors multiplied by two to represent a tour. Hence, the future cost Q_{lt} will increase if location l is located far away from the other locations. The parameter ϕ_l represents the service cost for location l where all cubes are emptied. Intuitively, the first term in (9) is the fraction of extra visits needed to location l and the second term is an estimate for the cost of an additional visit.

To give an incentive to move locations that do not necessarily have to be visited in PP to the artificial day, we divide the future cost for the artificial day, calculated in (9), by two. In this way the locations are moved forward in the time horizon and can therefore be included in the next PP.

The future cost for the treatment facility for glass is computed as

$$Q(\mathrm{g}) = \frac{\zeta(\mathrm{g})}{C} \cdot \frac{k(\mathrm{g}) + w(\mathrm{g})}{2}, \tag{10}$$

where $\zeta(\mathrm{g})$ is the estimated quantity of glass, which has to be collected in PP* after the latest planned visit to the treatment facility. Intuitively, $\zeta(\mathrm{g})$ is the amount of glass which has to be transported to the treatment facility at a later point in the time horizon. Intuitively, the first term in (10) is the fraction of extra visits needed to the treatment facility and the second term is an

estimate for the cost of an additional visit. The future cost $Q(\mathrm{p})$ for the treatment facility for paper is computed in a similar way.

The objective function used to evaluate the solution in the improvement of PP is a modification of the objective function stated in (7) and is given by

$$
\begin{aligned}
\min \sum_{i\in\mathcal{M}} \sum_{j\in\mathcal{M}\cup\{0\},j<i} \sum_{t\in\mathcal{PP}^*} (c_{ij}+w_{ij})y_{ij}^t &+ \sum_{t\in\mathcal{PP}^*} \sum_{l\in\mathcal{R}_t} Q_{lt} \\
+ \tfrac{1}{2}(k(\mathrm{g})+w(\mathrm{g})) \sum_{t\in\mathcal{PP}^*} \eta_t(\mathrm{g}) &+ Q(\mathrm{g}) \\
+ (k(\mathrm{p})+w(\mathrm{p})) \sum_{t\in\mathcal{PP}^*} \eta_t(\mathrm{p}) &+ Q(\mathrm{p})
\end{aligned}
\tag{11}
$$

where $\mathcal{R}_t$ represents the route on day t. The PP objective function (11) reflects the cost of the collection in PP* and is composed of the direct cost and the future cost. The future cost is included in (11) to give an incentive to delay the emptying of the cubes and containers as long as possible. In Section 5.1 we investigate if this inclusion leads to better solutions for the collection problem.

4.6. Improvement of the Collection Plan

To improve the collection in PP the locations are reorganized between the days in PP* by use of Variable Neighborhood Search (VNS). We reorganize the locations that belong to PP*, while remembering that we may not plan for cubes to be overfilled and capacity in the containers may not be exceeded. In the reorganization, the expected fill levels are used as a proxy for the unknown fill levels. The artificial day can contain locations, that do not have to be emptied in PP, but can be included if it leads to a better plan. A location that belongs to the artificial day can be moved to a day in $\mathcal{PP}$ and visa versa. The improvement is realized due to decrease in the total cost in the PP objective function (11).

Our VNS contains four neighborhoods (k_{max}) in the VNS part and three neighborhoods (h_{max}) in the VND part. The layout is shown in Algorithm 2. We use the current solution x for PP* as input to the VNS, and the improvement proceeds until a time limit (t_{max}) or a stop criterion (b_{max}) is reached. When searching for a better solution, shakes are performed on a solution x', which is a replicate of x.

All shakes consist of a block move where a block (also called a sequence of locations) is moved from a route on day t to a route on day t' in $\mathcal{PP}^*$. The best block of a given size in the route on day t is found and moved to the best place in the route on day t' in $\mathcal{PP}^*$, where day t and t' are randomly chosen in $\mathcal{PP}^*$, $t \neq t'$. The best block and best place are defined by the block of locations that has the largest gain between the removal cost from day t and the insertion cost on day t'. We have a total of four neighborhoods with block moves, where the first three consist of a block of two, three, or four locations, respectively. For the last neighborhood, the size of the block is chosen randomly between two and five. If a route consists of fewer locations than the block size, the block size is reduced to the number of locations on the route.

The VND part consists of three neighborhoods: Swap, Move, and 2–opt exchange. In the swap move, two locations are swapped between a route on day t and a route on day t' in $\mathcal{PP}^*$, $t \neq t'$. The two locations are inserted in the routes such that we get minimum insertion cost and may be in totally different positions in the routes. For all locations on day t the best swap move is made if an improvement can be obtained, $\forall t \in \mathcal{PP}^*$. In the second neighborhood, a single location is moved from a route on day t to a route on day t' in $\mathcal{PP}^*$, $t \neq t'$. The location is inserted in the

---
Algorithm 2: Improvement with VNS

Data: $x, k_{max}, h_{max}, t_{max}, b_{max}$

$time = 0, it = 0$;

while $time < t_{max}$ *or* $it < b_{max}$ **do**

    $k = 1$;

    while $k < k_{max}$ **do**

        $x' = x$;

        if $k = 1$ **then** $x' \leftarrow$ Block move size 2 **end**;

        if $k = 2$ **then** $x' \leftarrow$ Block move size 3 **end**;

        if $k = 3$ **then** $x' \leftarrow$ Block move size 4 **end**;

        if $k = 4$ **then** $x' \leftarrow$ Block move size Rand(2,5) **end**;

        $h = 1$;

        while $h < h_{max}$ **do**

            $x'' = x'$;

            if $h = 1$ **then** $x'' \leftarrow$ Swap two location on different routes, $\forall t \in \mathcal{PP}^*$ **end**;

            if $h = 2$ **then** $x'' \leftarrow$ Move a location to another route, $\forall t \in \mathcal{PP}^*$ **end**;

            if $h = 3$ **then** $x'' \leftarrow$ 2–opt exchange on a route, $\forall t \in \mathcal{PP}^*$ **end**;

            if $f(x'') < f(x')$ **then** $x' = x''$, $h = 1$ **else** $h = h + 1$ **end**;

        end

        if $f(x') < f(x)$ **then** $x = x'$, $k = 1$ **else** $k = k + 1$ **end**;

    end

    Update $time$ and it;

end
---

route at the selected day t' in the place with minimum insertion cost. For all locations on day t the best move is performed if an improvement can be obtained, $\forall t \in \mathcal{PP}^*$. In the last neighborhood, 2-opt exchange is executed on all routes in $\mathcal{PP}^*$ until no further improvement can be obtained.

Whenever a move is considered, a feasibility check with respect to capacity in cubes and containers is made. A move needs to be feasible for the affected locations regardless of whether a location is moved forward or backward in the time horizon. A location can be moved forward in PP* until it reaches the day $t^\dagger$ and it can be move backwards in PP* to any day where it has a family of cubes which is at least λ percent full. If a route on any given day is empty, no moves can be carried out and in this case the next day $t \in \mathcal{PP}^*$ will be considered. If a block move is feasible, it is executed on the solution x' without considering the impact on the cost. A feasible move in VND is performed with lowest total cost $f(x'')$ given by the PP objective function in (11). Locations are moved until no further decrease in the objective function is found. For the neighborhood with 2-opt exchanges, it is only necessary to calculate the cheapest interchange cost in the route, thus the other cost indicators will not be affected when a move is performed on a single day.

Every time a location is moved from one day to another, the estimate of the quantity of material planned to be collected is bound to change. This change should be recognized when the feasibility of a move is checked. If the plan is to collect both glass and paper from a location because the families of cubes for glass and paper are estimated to be at least β percent full, then a move of the location to a earlier day in PP* could change the estimated quantity of materials dramatically, resulting in a situation where only one type of material needs be collected, since the other family of

cubes is estimated to be less than β percent full. But if a location is moved to a later day in PP*, both materials may need to be collected instead of only one type of material and moreover, the quantity of materials planned for collection will increase.

The total estimated quantity of materials for collection on a given day needs to be updated every time a location is inserted or removed from the day. The container capacity C restricts the estimate from above. When executing the steps in Algorithm 2, the transportation to the treatment facilities is planned at the same time. The decision regarding the transportation to the treatment facilities is included in the PP objective function (11) and is depending on the planned collection in PP*. Whenever a move is considered and carried out, the cost and trip scheduling for the treatment facilities are recalculated.

4.7. Executing the Route at Day t and Transportation to the Treatment Facilities

After searching for improvements by the use of Algorithm 2, the route at day t is ready to be executed, possibly preceded by transportation to the treatment facilities. If the estimated quantity of glass to collect at day t exceeds $C_t(\mathrm{g})$, we swap container if possible or go to the treatment facility for glass before the route is executed. An identical procedure for paper is carried out.

During a visit to a location in the route at day t, a cube is emptied if it is at least β percent full, i.e. if

$$F_{it} \geq \beta U_i \quad i \in \mathcal{M}.$$

Otherwise the material is noted and remains in the cube. This information is used later to determine the next visit to the location. Computing the amount of material F_{it} present in the cube i at time t is a two-step procedure. We describe the procedure for glass. The procedure for paper is equivalent.

We first determine the quantity of material placed in the family of glass cubes at location l since the last visit at day $\tilde{t}$. For each day $h \in \{\tilde{t}, \ldots, t-1\}$, we therefore generate the accretion rate for the family of glass cubes. This is done using a normal distribution, where we use the estimated mean from Reno Djurs and a standard deviation of 5% of the mean. If a negative quantity is generated, it is substituted by zero. Hence, the total amount of glass placed in the family of glass cubes $\mathcal{G}(l)$ at location l since time $\tilde{t}$ is

$$D(\mathrm{g}) = \sum_{h=\tilde{t}}^{t-1} \sum_{i \in \mathcal{G}(l)} r_{ih}.$$

Next, we determine the fraction $D_i(\mathrm{g})$ of $D(\mathrm{g})$ to be placed in cube $i \in \mathcal{G}(l)$ where $\sum_{i \in \mathcal{G}(l)} D_i(\mathrm{g}) = D(\mathrm{g})$, and thereby the quantity of material F_{it} present in cube i at time t. At a location, some cubes are usually more convenient to access for the citizens than others. Moreover, the citizens will place material in a cube with available capacity rather than overfilling another cube in the family. We have developed an algorithm that takes these human factors into account when determining $D_i(\mathrm{g})$. Algorithm 3 shows a simplified version of our algorithm. In this simplified version, we only show the distribution of material when every cube in the family has sufficient spare capacity to be able to hold the quantity assigned to it. In the algorithm, $Z_i(\mathrm{g})$ represents the quantity of glass we left in cube i at day $\tilde{t}$, and thereby $\sum_{i \in \mathcal{G}(l)} Z_i(\mathrm{g}) = Z(\mathrm{g})$. It should be noted that using this algorithm, cube number $|\mathcal{G}(l)|$ will on average contain the most material at location l and it therefore represents the most easily accessible cube. We stress the fact that the distribution of material among cubes is unknown to our solution algorithm and F_{it} is thereby not revealed until the visit at day t.

14

Algorithm 3: Partition of the materials at time t

Initialization;

$D(g) = \sum_{h=\hat{t}}^{(t-1)} \sum_{i \in \mathcal{G}(l)} r_{ih}$;

$k = |\mathcal{G}(l)|$;

while $k > 0$ **do**

    Let i be the k'th cube in $\mathcal{G}(l)$;

    if $k = 1$ **then**

        $D_i(g) = D(g)$;

    else

        $D_i(g) = \text{Uniform}\left[\frac{1}{k}D(g), \frac{3}{2} \cdot \frac{1}{k}D(g)\right]$;

        if $D_i(g) > (U_i - Z_i(g))$ **then**

            $D_i(g) = (U_i - Z_i(g))$;

        end

    end

    $F_{it} = D_i(g) + Z_i(g)$;

    $D(g) = D(g) - D_i(g)$;

    $k = k - 1$;

end

After the execution of a route, paper from the container at the vehicle will be partly or fully offloaded to the extra container at the depot, if it is not full. Afterwards, the capacities available $C_{(t+1)}(g)$ and $C_{(t+1)}(p)$ for the next day are updated and overfilled cubes and containers at day t are reported. Finally, the visited locations are reinserted into routes in $\mathcal{T}$ based on (8) as described in Section 4.4. If the resulting day $\hat{t}$ is outside the time horizon, the insertion is not performed and therefore a final inspection for overfilled cubes is made at the very end of the time horizon.

5. Computational Results

In this section, we present our computational results. In Section 5.1, we investigate if an inclusion of the future cost as described in Section 4.5 leads to better solutions for the collection problem. This is followed by an analysis in Section 5.2 of the impact of threshold β on the solution quality. As the standard deviation of 5% of the mean of the daily accretion rates is based on an estimate provided by the case company, we analyze the effects of increasing the standard deviation in Section 5.3. Finally in Section 5.4, we examine how a possibility of offloading glass at the depot (similar to what is possible for paper) will affect the results.

The algorithm has been implemented in C++ using Microsoft Visual Studio 2010. All experiments were carried out on a Dell laptop with 4 GB RAM and a 2.5 GHz Intel®Core i5-3210M processor running a 64 bit version of Windows 7.

The analysis is based on real world data from Reno Djurs. All results are reported as mean of 100 simulations over a finite time horizon H of 365 days. In some situations, the minimum and maximum over the 100 simulations are also reported. All cost related results are reported as minutes. The average runtime per simulation is 45 minutes.

The best values for the thresholds λ and β have been determined by computational experiments to be 0.5 and 0.3, respectively. The best values of the thresholds α_{red}, α_{yellow}, and α_{green} are

determined to be 0.7, 0.8, and 0.9, respectively. In the VNS, the time limit t_{max} is set to 10 seconds and the stop criterion b_{max} becomes effective after 1000 iterations without any improvements.

5.1. Inclusion of the Future Cost

In this section, we present the results for the main case and investigate the benefit of including the future cost element described in Section 4.5. The results are presented in Tables 3–5, where the first result row represents the main case studied in this paper. Table 3 contains the cost related components. Table 4 gives an overview of the key performances related to cubes, which consist of the number of cubes emptied as well as the fill levels. Furthermore, the number of overfilled cubes and the associated amount of overfilled material of these cubes are reported. Table 5 contains the same key performances for the containers.

The main case has an average total cost of 37,346 for a collection plan for a year, corresponding to about 12 hours per week, with a small standard deviation over the simulations. Time spent on routing composes the largest components of the total time. Approximately one third of the total time elapsing on a route is spent on servicing the cubes. Transportation to the treatment facilities also accounts for a large share of the total time spent, which mainly reflects the relatively large distances between the depot and the treatment facilities, since the number of visit to the facilities is fairly small. On average, about 2,052 glass cubes and 1,764 paper cubes are emptied each year. When the cubes are emptied, the fill levels are about 70 percent for glass and 68 percent for paper, which is significantly more than the fill levels of 55 percent for glass and 40 percent for paper known from historical data. On average, there is 1 overfilled cube per year and the average quantity of overfilled material is $0.0045\,\mathrm{m}^3$, which corresponds to less than 0.3 percent of the capacity of a cube. The treatment facilities for glass and paper are visited about 35 and 29 times per year, respectively. On average, the containers are 93 percent full when emptied at the treatment facilities. The number of overfilled containers ranges from 0 to 5 with an average of 1.9 and the associated quantity of overfilled material is on average $0.2264\,\mathrm{m}^3$, which corresponds to less than 0.7 percent of the capacity of a container. Summarizing, the overall performance of our algorithm for solving this real world problem is satisfactory.

Table 3: Cost results for the algorithm with and without future cost.

	Total Cost				Routing	Service	Transport
	mean	SD	min	max			
With future cost	37346	318	36497	38116	16189	8707	12450
Without future cost	39600	359	38649	40446	17445	9105	13050

In the following, the main case with future cost included will be compared to a version of the algorithm without inclusion of future cost.

The algorithm including future cost outperforms the one without future cost on all cost indicators. The standard deviation of the total cost shows that the algorithm including future cost produces more stable results. The algorithm including future cost has an average of 1 overfilled cube compared to 0.7 overfilled cube for the algorithm without future cost. But the former empties fewer cubes per year, which results in lower service cost. And the fill levels for the cubes that are emptied are also higher. The performance measures related to containers, given in Table 5, show the same trend.

Table 4: Cubes results for the algorithm with and without future cost.

	No. emptied		Fill level		No. overfilled			m³ overfilled		
	glass	paper	glass	paper	mean	min	max	mean	min	max
With future cost	2052	1764	70.2	68.3	1.0	0	4	0.0045	0	0.0365
Without future cost	2133	1846	67.5	65.2	0.7	0	3	0.0029	0	0.0279

Table 5: Container results for the algorithm with and without future cost.

	No. trips		Fill level		No. overfilled			m³ overfilled		
	glass	paper	glass	paper	mean	min	max	mean	min	max
With future cost	35.0	29.2	92.8	92.6	1.9	0	5	0.2264	0	0.8256
Without future cost	36.6	31.0	88.8	86.8	0.6	0	3	0.1151	0	1.1530

The future cost is included in the algorithm as an incentive to delay the service of the cubes and containers as long as possible. We conclude that the desired effect has been achieved, as apparent from higher average fill levels for both cubes and containers. At the same time, lower costs without a significant increase in the number and quantities of overfilling are obtained by the use of future cost. We conclude that using the future cost element is indeed beneficial.

5.2. Different Settings of the Threshold β

In the following, we analyze the effects of adjusting the value of the threshold β. Selected results are presented in Table 6. It contains total cost, number of emptied cubes, and key performances for the containers, i.e. the number of overfilled containers and the associated amount of overfilled material. In the main case, shown in row four, β is 30 percent. The table shows that the main

Table 6: Results for different settings of the threshold β.

	Total Cost	No. emptied cubes		No. overfilled container			m³ overfilled container
		glass	paper	mean	min	max	
$\beta = 0.0$	37466	2115	1820	0.2	0	2	0.0050
$\beta = 0.1$	37471	2104	1802	0.4	0	2	0.0168
$\beta = 0.2$	37396	2091	1792	1.4	0	6	0.1081
$\beta = 0.3$	37346	2052	1764	1.9	0	5	0.2264
$\beta = 0.4$	37719	1993	1696	7.0	2	12	0.6399
$\beta = 0.5$	39112	1903	1605	13.9	8	20	1.0022

case provides the best result. However, the difference in the cost is not significant for β equal to 0–40 percent, whereas β equal to 50 percent gives a poorer result than the others. One explanation for this is that when β becomes close to λ, we sometimes visit a location where we expect a family of cubes to be more than λ percent full, but it turns out that the family is slightly less than β (and λ) percent full and consequently, the family of cubes is not emptied. In the experiment this happens 2.5 times on average during a year.

Note that β equal to 40 and 50 percent result in a substantially higher number of overfilled containers. This can be explained by the stochastic nature of the fill levels: If chance causes the estimate to be slightly off, it will impact the total quantity of material for collection more when β is equal to 40 or 50 percent than 0–30 percent, i.e. the amount of unexpected material to be collected is larger and will be less likely to fit into the container. Similarly, in the case of overfilling in a container, the amount of overfilled material is larger for high values of β.

As β increases, the number of emptied cubes decreases and consequently the service cost decreases. However, the total cost is relatively stable when β is in the range of 0–40 percent. The results for cubes are almost identical regardless of the setting of β. Even higher fill levels and lower service cost can be achieved by adjusting λ and β, but it results in higher total cost. For an efficient flow of the system without substantial overfillings, β in the range of 0–30 percent is recommended.

5.3. Increasing Uncertainty

In this section, we present and analyze results from an instance with higher standard deviation of the daily accretion rates. The instance has the same settings as the main case. For the distribution, the estimated mean from Reno Djurs is used, but the standard deviation is increased to 10 % of the mean for every location. Notice that the changed standard deviation will influence the classification of the families of cubes since the standard deviation is used in (8). Selected results are provided

Table 7: Results for different standard deviations.

	Total Cost		No. overfilled cubes			m^3 overfilled cubes			Container fill level	
	mean	SD	mean	min	max	mean	min	max	glass	paper
5 % of mean	37346	318	1.0	0	4	0.0045	0	0.0365	92.8	92.6
10 % of mean	38165	375	1.8	0	5	0.0189	0	1.2294	89.6	90.7

in Table 7, which contains the total cost and associated standard deviation as well as the number of overfilled cubes and the associated amount of overfilled material. Finally, the fill levels for containers are reported. As expected higher standard deviation leads to higher total cost and more overfilled cubes. In addition, the fill levels for containers are lower because a larger safety capacity is needed in the containers when uncertainty increases. However, the algorithm still provides acceptable results even though uncertainty is increased.

5.4. Offloading Glass

Due to shattering it is not currently possible for glass to be offloaded with the crane to the extra glass container at the depot. However, a new recycling procedure at the treatment facility for glass may remove the non-breaking restriction for glass. Consequently, glass and paper will be handled in the same way (but still separately) in the collection process. Consequences of a possibility of offloading glass are therefore examined. Table 8 provides the total cost and the transportation cost for the treatment facilities as well as the fill levels and quantity of overfilled material for containers. As expected, a lower total cost is obtained with a possibility of offloading glass. However, the gain is surprisingly small and no significant difference can be seen in the affected key performances for the containers. This can be explained by the fact that the container capacities are already utilized well. We conclude that no significant difference in the cost and performance is obtained if offloading glass at the depot becomes possibly.

Table 8: Results for a possibility of offloading glass.

	Total Cost	Transport Cost	Container fill level		m^3 overfilled container		
			glass	paper	mean	min	max
No offload	37346	12450	92.8	92.6	0.2264	0	0.8266
Offload	37269	12521	91.6	94.0	0.1558	0	0.7931

6. Concluding Remarks

In this paper, we have presented a solution approach for the collection of recyclable materials from so-called cubes. A VNS algorithm is used to solve the problem in a planning period of a few days. This is embedded into a rolling horizon framework which benefits from the inclusion of a future cost function that estimates long term effects of short term decisions. It is evaluated by a simulation study and shows promising and robust performance. Compared to the real world case, the algorithm shows more efficient use of capacities. Directions for future studies on this topic include an analysis of the possible benefits of installing fill level information systems in the cubes. Such systems can make it possible to partially remove the uncertainty in the problem and could thereby lead to better planning and minimization of the total cost.

References

Aksen, D., Kaya, O., Salman, F. S., Tüncel, Ö., 2014. An adaptive large neighborhood search algorithm for a selective and periodic inventory routing problem. European Journal of Operational Research 239 (2), 413–426.

Andersson, H., Hoff, A., Christiansen, M., Hasle, G., Løkketangen, A., 2010. Industrial aspects and literature survey: Combined inventory management and routing. Computers and Operations Research 37 (12), 1515–1536.

Angelelli, E., Speranza, M. G., 2002. The application of a vehicle routing model to a waste-collection problem: Two case studies. The Journal of the Operational Research Society 53 (9), 944–952.

Benjamin, A. M., Beasley, J. E., 2010. Metaheuristics for the waste collection vehicle routing problem with time windows, driver rest period and multiple disposal facilities. Computers and Operations Research 37 (12), 2270–2280.

Bertazzi, L., Bosco, A., Guerriero, F., Laganá, D., 2013. A stochastic inventory routing problem with stock-out. Transportation Research Part C: Emerging Technologies 27 (0), 89–107.

Beullens, P., Oudheuseden, D. V., Wassenhove, L. N., 2010. Collection and vehicle routing issuse in reverse logistics. In: Dekker, R., Fleischmann, M., Inderfurth, K., L.Wassenhove (Eds.), Reverse Logistics. Springer, Berlin, pp. 95–134.

Bogh, M. B., Mikkelsen, H., Wøhlk, S., 2014. Collection of recyclables from cubes - a case study. Socio-Economic Planning Sciences 48 (2), 127–134.

Coelho, L. C., Cordeau, J. F., Laporte, G., 2014a. Heuristics for dynamic and stochastic inventory-routing. Computers and Operations Research 52, Part A (0), 55–67.

Coelho, L. C., Cordeau, J. F., Laporte, G., 2014b. Thirty years of inventory routing. Transportation Science 48 (1), 1–19.

Coelho, L. C., Laporte, G., 2015. Classification, models and exact algorithms for multi-compartment delivery problems. European Journal of Operational Research 242 (3), 854–864.

Golden, B. L., Assad, A. A., Dahl, R., 1984. Analysis of a large scale vehicle routing problem with an inventory component. Large Scale Systems in Information and Decision Technologies 7 (2–3), 181–190.

Golden, B. L., Assad, A. A., Wasil, A. E., 2002. Routing vehicles in the real world: applications in the solid waste, beverage, food, dairy, and newspaper industries. In: Toth, P., Vigo, D. (Eds.), The vehicle routing problem. Society for Industrial and Applied Mathematics, USA, pp. 245–286.

Kim, B., Kim, S., Sahoo, S., 2006. Waste collection vehicle routing problem with time windows. Computers and Operations Research 33 (12), 3624–3642.

Muyldermans, L., Pang, G., 2010. On the benefits of co-collection: Experiments with a multi-compartment vehicle routing algorithm. European Journal of Operational Research 206 (1), 93–103.

Teixeira, J., Antunes, A. P., de Sousa, J. P., 2004. Recyclable waste collection planning – a case study. European Journal of Operational Research 158 (3), 543–554.

A. Mathematical model

A mathematical model for the deterministic version of the problem is presented here with detailed explanations. To the best of our knowledge, the constraints used to model the container capacities are not found elsewhere in the literature. In the model, we assumed that r_{it} are known for all $i \in \mathcal{M}$ and $t \in \mathcal{T}$. In addition to the variables introduced above, we need further variables for our mathematical model.

We define a binary variable θ_{it} to be 1 if cube $i \in \mathcal{M}$ is emptied at time $t \in \mathcal{T}$, and 0 otherwise. For $i = 0$, which represents the depot, $\theta_{it} = 1$ indicates that a collection route is performed at time t.

The variable $\eta_t(\mathrm{g})$ is introduced to control the swap between glass containers and is defined as

$$\eta_t(\mathrm{g}) = \begin{cases} 1 & \text{if the glass containers are swapped at time } t \in \mathcal{T}, \\ 0 & \text{otherwise.} \end{cases}$$

When the glass containers have been swapped twice, a trip to the treatment facility is required. The variable for paper is slightly different, as it controls the trips to the treatment facility rather than the swaps. We have

$$\eta_t(\mathrm{p}) = \begin{cases} 1 & \text{if the two paper containers are emptied at the treatment facility at time } t \in \mathcal{T}, \\ 0 & \text{otherwise.} \end{cases}$$

Instead of using $C_t(\mathrm{p})$ which represents the amount of capacity available for paper at the vehicle at time t, we use a variable $C_t(\dot{\mathrm{p}})$ that states the total amount of available capacity for paper at time t in the mathematical model.

To represent the routing of the vehicle, we introduce variables y_{ij}^t for each pair i and j with $i \in \mathcal{M}, j \in \mathcal{M} \cup \{0\}, j < i$, where $\{0\}$ represents the depot and each $t \in \mathcal{T}$. y_{ij}^t is defined as the number of direct travels by the vehicle between i and j at time t. Note that the direction of the travel is not given. When j represents the depot, y_{i0}^t can take the values 0, 1, or 2. For all other values of j, the variables are binary.

The model consists of three parts which are related to cubes, containers, and routing, respectively. We consider each part in turn.

Constraints (1) and (3) control the amount of material in the cubes, but (2), which defines the amount to be emptied, must be linearized. We have

$$x_{it} \geq U_i \theta_{it} - U_i + F_{it} \qquad \forall i \in \mathcal{M}, t \in \mathcal{T} \tag{A.1}$$

$$x_{it} \leq F_{it} \qquad \forall i \in \mathcal{M}, t \in \mathcal{T} \tag{A.2}$$

$$x_{it} \leq U_i \theta_{it} \qquad \forall i \in \mathcal{M}, t \in \mathcal{T} \tag{A.3}$$

$$x_{it} \geq 0, F_{it} \geq 0 \qquad \forall i \in \mathcal{M}, t \in \mathcal{T}' \tag{A.4}$$

Together, constraints (A.1)–(A.4) ensure that a cube is totally emptied if emptied at all. If $\theta_{it} = 1$, implying that cube i is visited, constraints (A.1) and (A.2) will force $x_{it} = F_{it}$. On the other hand,

if $\theta_{it} = 0$, representing that cube i is not visited at time t, constraints (A.3) and (A.4) will force $x_{it} = 0$. These constraints also ensure that cube i is emptied at time t if it would otherwise become overfilled at time $t+1$. Note that the upper bound on F_{it} forces x_{it} to take a strictly positive value in this case by constraints (1) and (3). The positive value of x_{it} forces $\theta_{it} = 1$ by constraints (3) and (A.1).

Next, we consider the amount of material in the glass container and the swap between containers. We have

$$\sum_{i\in\mathcal{G}} x_{it} \leq C_t(\mathrm{g}) \cdot \theta_{0t} \qquad \forall t \in \mathcal{T} \tag{A.5}$$

$$C_t(\mathrm{g}) = \Big(C_{(t-1)}(\mathrm{g}) - \sum_{i\in\mathcal{G}} x_{i(t-1)}\Big)(1 - \eta_t(\mathrm{g})) + C \cdot \eta_t(\mathrm{g}) \qquad \forall t \in \mathcal{T} \tag{A.6}$$

$$0 \leq C_t(\mathrm{g}) \leq C \qquad \forall t \in \mathcal{T}' \tag{A.7}$$

Constraints (A.7) bound the available capacity by the size of the container and constraints (A.5) are a modification of (4) where the addition of θ_{0t} ensures that collection can only be accomplished when a route is activated from the depot. Constraints (A.6) control the available capacity at time t. If $\eta_t(\mathrm{g}) = 1$, i.e. the containers are swapped at time t, the constraint reduces to $C_t(\mathrm{g}) = C$. If, on the other hand, $\eta_t(\mathrm{g}) = 0$, it reduces to $C_t(\mathrm{g}) = C_{(t-1)}(\mathrm{g}) - \sum_{i\in\mathcal{G}} x_{i(t-1)}$.

We consider the amount of material in the paper containers and the transportation to the treatment facility in a similar way. We have

$$\sum_{i\in\mathcal{P}} x_{it} \leq \min\{C, C_t(\dot{\mathrm{p}})\} \cdot \theta_{0t} \qquad \forall t \in \mathcal{T} \tag{A.8}$$

$$C_t(\dot{\mathrm{p}}) = \Big(C_{(t-1)}(\dot{\mathrm{p}}) - \sum_{i\in\mathcal{P}} x_{i(t-1)}\Big)(1 - \eta_t(\mathrm{p})) + 2C \cdot \eta_t(\mathrm{p}) \qquad \forall t \in \mathcal{T} \tag{A.9}$$

$$0 \leq C_t(\dot{\mathrm{p}}) \leq 2C \qquad \forall t \in \mathcal{T}' \tag{A.10}$$

Constraints (A.10) bound the total available capacity by the size of the two containers. Constraints (A.8) ensure that if a collection route is activated at time t, i.e. if $\theta_{0t} = 1$, then the amount of paper to be collected is bounded by the size of the container as well as by the current total available capacity. They also ensure that if no route is activated, then no material can be collected. Constraints (A.9) are similar to (A.6) and adjust the available capacity for paper.

Finally, we need a set of constraints to control the routing of the vehicle. We have

$$\sum_{j\in\mathcal{M}\cup\{0\},j<i} y_{ij}^t + \sum_{j\in\mathcal{M}\cup\{0\},j>i} y_{ji}^t = 2\theta_{it} \qquad \forall i \in \mathcal{M} \cup \{0\}, t \in \mathcal{T} \tag{A.11}$$

$$\sum_{i\in\mathcal{B}}\sum_{j\in\mathcal{B},j<i} y_{ij}^t \leq \sum_{i\in\mathcal{B}} \theta_{it} - \theta_{bt} \qquad \forall \mathcal{B} \subseteq \mathcal{M}, b \in \mathcal{B}, t \in \mathcal{T} \tag{A.12}$$

$$y_{ij}^t \in \{0,1\} \qquad \forall i \in \mathcal{M}, j \in \mathcal{M}, j < i, t \in \mathcal{T} \tag{A.13}$$

$$y_{i0}^t \in \{0,1,2\} \quad \forall i \in \mathcal{M}, t \in \mathcal{T} \tag{A.14}$$

Here, (A.11) ensure that if cube i is emptied at time t, i.e. $\theta_{it} = 1$, then two adjacent links are used for routing. Hence, the vehicle will enter and leave the cube. Constraints (A.12) are subtour elimination constraints which are activated for all cubes i that are visited at time t, i.e. those with $\theta_{it} = 1$. The final two constraints bound the variables.

To formulate the objective function, we define the constants $k(\mathrm{g})$ and $k(\mathrm{p})$ to be the cost of a round trip to the treatment facility for glass and paper, respectively, with two containers. Every time we empty containers at the treatment facility for glass and paper a service cost $w(\mathrm{g})$ and $w(\mathrm{p})$ is applied, respectively. Furthermore, we define c_{ij} as the cost of traveling between cube i and j (i and the depot, if $j = 0$). Every time we empty cube i after cube j a service cost w_{ij} is applied.

The complete model can now be formulated as follows, where (A.15a) restricts the cubes and (A.15b) and (A.15c) control the containers for glass and paper, respectively. (A.15d) are the routing constraints and the remaining constraints bound the variables.

$$\min \quad \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{M} \cup \{0\}, j < i} \sum_{t \in \mathcal{T}} (c_{ij} + w_{ij}) y_{ij}^t + \tfrac{1}{2}(k(\mathrm{g}) + w(\mathrm{g})) \sum_{t \in \mathcal{T}} \eta_t(\mathrm{g}) + (k(\mathrm{p}) + w(\mathrm{p})) \sum_{t \in \mathcal{T}} \eta_t(\mathrm{p})$$

$$\text{s.t.} \quad (1), (3), (A.1) - (A.4) \tag{A.15a}$$

$$(A.5) - (A.7) \tag{A.15b}$$

$$(A.8) - (A.10) \tag{A.15c}$$

$$(A.11) - (A.14) \tag{A.15d}$$

$$\eta_t(\mathrm{g}) \in \{0, 1\}, \quad \eta_t(\mathrm{p}) \in \{0, 1\} \qquad \forall t \in \mathcal{T} \tag{A.15e}$$

$$\theta_{it} \in \{0, 1\} \qquad \forall i \in \mathcal{M} \cup \{0\}, t \in \mathcal{T} \tag{A.15f}$$

This mixed integer model contains a total of $(\frac{|M|^2}{2} + 3.5|M| + 5)H + 4|M| + 4$ variables, of which $(2|M| + 2)H + 4|M| + 4$ are continuous, $|M|H$ are integral with domain $\{0, 1, 2\}$, and the remaining $(3 + \frac{|M|^2 + |M|}{2})H$ are binary. The model contains $7|M|H + 4|M| + 7H + 4$ constraints in addition to the domain constraints ((A.13), (A.14), (A.15e), (A.15f)), and an exponential number of subtour elimination constraints (A.12). As the purpose of this model is to present a precise description of the problem rather than to present a model to be directly solved, no attempt is made to linearize constraints (A.5), (A.6), (A.8), and (A.9). As it stands here, the model is therefore nonlinear and nonconvex