THE SECTORING OF A WASTE COLLECTION AREA

by

Philine Scheepers
27043275

Submitted in partial fulfillment of the requirements for the degree of

Bachelors of Industrial Engineering

in the

Faculty of Engineering, Built Environment and Information Technology

University of Pretoria

October 2010

Executive summary

Waste Management is an iterative process and optimization will thus lead to substantial cost savings. Although municipalities spend a significant amount of their budget on the waste collection and transportation process, it is not prioritized and ineffective strategies are followed when looking at the disposal of municipal solid waste. The objective of this paper is to improve the current waste collection strategy by partioning a waste collection service area into collection days and further into collection vehicles. The problem is identified as an Arc Routing Problem (ARP) and due to its complexity, heuristic procedures are incorporated in finding a solution. Three methods were evaluated and identified as suitable. The first method is called the Two-Phase Heuristic (TPH); in phase 1, sectors are built and in phase 2 the routing within each sector is established. The second method, called the Best Insertion Heuristic (BIH), computes the sectoring and routing simultaneously and sectors are built by adding tasks in a best insertion manner. With the third method, called the Exact Cost Heuristic (ECH), the sectoring and routing are also done simultaneously, but the sectors are built by adding a task and computing the exact cost of the sector by making use of a routing algorithm. These three methods are tested on benchmark problems and the results evaluated.

Acronyms

ARP	Arc Routing Problem
BIH	Best Insertion Heuristic
CARP	Capacitated Arc Routing Problem
CDA	Collection Days Algorithm
CTH	Circuit of Tasks Heuristic
CSIR	Council of Scientific and Industrial Research
ECH	Exact Cost Heuristic
IF	Intermediate Facility
MCARP	Mixed Capacitated Arc Routing Problem
RA	Routing Algorithm
SARP	Sectoring Arc Routing Problem
TPH	Two-Phase Heuristics

Contents

Executive summary ii

1 Introduction 1

1.1 Waste Definition 1

1.2 Current Residential Waste Collection Approach 2

1.2.1 Problems Faced by Municipalities 2

1.3 The Grant Project 3

1.4 Sectoring of a service area 4

1.5 Research Approach and Methodology 4

1.6 Document Structure 5

2 Literature review 6

2.1 Service Area Description 6

2.2 Arc Routing Problem 7

2.2.1 Heuristic Procedures 8

2.3 Sectoring a Service Area 9

2.3.1 Districting for salt spreading operations 9

2.3.2 Sector design for snow disposal operations 11

2.3.3 Sectoring the arc routing problem with heuristics 12

2.4 Conclusion 13

3 Models for the SARP 14

3.1 Problem Modelling and Notation 14

3.1.1 Network model 14

3.1.2 Modelling of sectors 14

3.2 The Two-phase heuristic 16

3.3 Best Insertion Heuristic 19

3.4 Exact Cost Heuristic 19

3.5 Intermediate Facilities . . . 19
3.6 Conclusion . . . 21

4 Computational Results 22
4.1 Evaluation criteria . . . 22
4.2 Instances and parameters . . . 23
4.3 Comparisons between the heuristics . . . 24
4.4 Fabricated Case Study . . . 28
4.4.1 CDA Evaluation . . . 29
4.5 Conclusion . . . 30

Appendix A 34

List of Figures

1.1 Operations Research Optimization Process 4

2.1 Arcs vs Edges 7
2.2 Area Presentation 8
2.3 Regions for Phases 1 and 2 10

List of Tables

4.1 Mathematical symbols 23
4.2 Benchmark features 24
4.3 Evaluation criteria results of lpr files 25
4.4 Evaluation criteria results of lpr-IF files 25
4.5 Results for lpr files 26
4.6 Results for lpr_IF files 27
4.7 CDA: Results 28
4.8 ECH-IF: Results 31

List of Algorithms

1 RouteCost 16
2 Single Task Heuristic 17
3 Two-phase heuristic 18
4 Exact Cost Heuristic 20
5 Collection Days Algorithm 29

Chapter 1

Introduction

1.1 Waste Definition

Waste is characterized by the Government Gazette, 24/08/1990, as an undesirable product, emission or a residue of any process or activity. Municipalities expend approximately 75% of their budget on the transportation and collection of solid waste [10]. Municipal solid waste is divided into three categories [11]: (1) Commercial Waste, (2) Roll-on-Roll-off Waste and (3) Residential Waste.

Commercial waste is found at small businesses, restaurants and apartments. It is required of vehicles to visit the same customer more than once a week. Vehicles have to unload at a landfill site two to three times daily due to the big demand in these areas. There is on average a daily activity change of 20% with commercial routes and the workload per street is inconsistent [11].

Roll-on-Roll-off waste (Industrial Waste) is found at industrial areas, construction sites and areas that generate high volumes of waste. The difference between industrial routes and commercial routes is the size of the container. The industrial container is up to four times bigger than that of the commercial container. As a result, a vehicle can only handle one industrial container at a time. Vehicles collect the containers, unload it at landfill sites and return them to the customer.

Residential Waste is found at households. Household waste needs to be collected at least once a week. The weekly demand for service in these areas are mostly consistent. A fixed improvement strategy can thus be implemented, as it doesn't have to be adjusted on a regular basis. In South Africa residential waste is either stored in bags, bins or in containers on the curb side in front of properties where service vehicles collect them. The optimisation of residential waste collection will be the focus of this project as (1) it forms the biggest percentage of waste collected by municipalities and (2) the collection process is repetitive, thus a small improvement can make a significant difference.

1.2 Current Residential Waste Collection Approach

Residential waste in South Africa is mostly collected on a weekly basis. Households place generated waste next to the street in front of their properties, where vehicles collect the waste on specified collection days. Each truck is assigned to a specific area that it has to service for a given day. All vehicles are kept at a depot and start their servicing at a certain street segment. Once a vehicle has reached its full capacity, it travels to the nearest intermediate facility, unloads the collected waste and returns to its original route to continue with waste collection.

Depots are locations where waste is kept permanently to be recycled and reworked whereas intermediate facilities (IF) are temporary waste storage locations distributed throughout the service areas. The IFs minimizes the travelling distances of the vehicles as they don't have to drive all the way to the depots to unload their collected waste. Another set of vehicles are assigned to transport the waste stored at the IF's to the nearest depot.

The assignment of vehicles to service areas is done by dividing a map into the amount of service days after which these areas are further divided into sections that are equal to a single vehicle's capacity. There is no fixed method currently used for sectoring the areas as it is done manually [7]. Two main problems with the current division of the service area are [5]: (1) the demand in each sector will vary and this will lead to some vehicles experiencing idle time and others will be overworked; (2) all areas included in the service areas will not necessarily be residents paying for waste services because of their existence being unknown.

1.2.1 Problems Faced by Municipalities

Studies show [4] that with the waste management strategy followed in South Africa, municipalities are faced with service delivery backlogs and struggle to sustain existing services. The following are general challenges faced by local municipalities in terms of solid waste management [4]: (1) volume increase in domestic waste;

(2) cost increases for waste disposal services; (3) deficient systems and integrated planning employed by municipalities in waste management; (4) insufficient funding for waste programs; (5) the improvement of waste management is not a priority to municipalities or the government.

Gauteng is the smallest province in South Africa with a population of over 9 million (2005) residents and is also economically the fastest growing province in South Africa [4]. The population is growing faster than the growth in basic services such as housing, infrastructure, etc. The demand has increased for waste management services in terms of transportation, collection, storage facilities and the treatment of disposed waste. Waste expenditure is getting out of proportion and a fixed strategy for municipal solid waste collection needs to be developed.

1.3 The Grant Project

The student is part of a project initiated by the CSIR. The aim of this project is to address municipal waste collection transportation by evaluating four critical components namely: (1) the relevent number and type of collection vehicles; (2) breaking the problem up into more manageable steps by districting the collection area; (3) finding the optimal number and location of landfill sites; (4) and determine balanced routes for collection vehicles. The question that the Grant Project will answer is:

> *How should the four optimization models be integrated to provide complete and holistic decision support to municipalities?*

The first three components were handed to three students at The University of Pretoria. These components need to be investigated and algorithms are to be formulated. The focus of this paper will be on the districting of collection areas and will answer the following research question:

> *How will the sectoring of a service area into collection days, and further sectoring of collection days into collection vehicle areas influence the efficiency of solid waste management?*

1.4 Sectoring of a service area

The sectoring problem is defined by Mourao et al. [5] as the partitioning of a large service area into smaller districts to ease the management of certain activities. A sector is often constructed by assigning it to a certain resource or facility location. Sectoring approaches have been developed by Mourao et al. [5], Perrier et al. [8] and Muyldermans et al. [6] where a sector's size is typically constrained by the workload capacity of a single resource i.e. a resource starts and ends at a depot location and goes on to service the next sector. The waste collection problem is complicated due to the many factors to incorporate, such as capacity, demand, time, etc. It can be defined as NP-hard (nondeterministic class problem) as it can only be solved with polynomial algorithms in small instances and it will require an excessive amount of computer time to arrive at an optimal solution. The complexity of the problem can be reduced by breaking the problem up into smaller, more manageable problems. The sectoring of a service area will make the municipal solid waste collection procedure easier to manage.

1.5 Research Approach and Methodology

The project will incorporate tools and techniques from the discipline of Operations Research and thus according to Randin and Maynard [9], the process is divided into four phases. These four phases are presented in Figure 1.1.

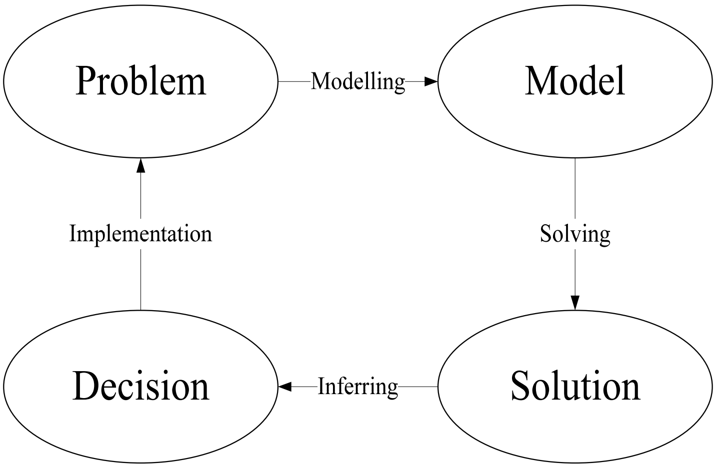

Figure 1.1: Operations Research Optimization Process

This approach can be applied to solve the problem as follows: In the *first phase* a model of the sectoring problem is constructed, consisting of a formal mathematical formulation. The *second phase* entails the solving of the developed model (constructed in phase one) by an optimization algorithm. This algorithm will be tested on benchmark problems and will be revised and improved continuously until the solution is adequate to the addressed problem. In the *third phase*, a decision will be made according to the applicability of the algorithm to the sectoring problem. In the *final phase*, the algorithm will be implemented and used to generate solutions for the sectoring problem and this will aid the municipalities in decision-making.

1.6 Document Structure

The sectoring problem and its application in literature are discussed in Chapter 2. These applications are investigated and an applicable solution strategy is identified. Chapter 3 presents an in-depth discussion of the strategy identified and the algorithm developed. In Chapter 4 the computational results are presented, with comparisons between two methods.

Chapter 2

Literature review

Most of the approaches used in literature for the optimization of solid waste collection focus only on the optimization of vehicle routes and schedules. There are, however, a few cases where the sectoring of a service area is proposed. These cases will be described and investigated in this paper.

2.1 Service Area Description

Residential areas consist of streets and street intersections (nodes). Each street contains a certain amount of waste that needs to be collected on a specific collection day; this can be referred to as the demand of a street. All streets have specific *service times* and *deadheading times*. Service time is defined as the time it takes a vehicle to service a specific street and deadheading time is the time it takes for a vehicle to traverse a street without servicing it. Streets can be classified as either arcs or edges, see Figure 2.1. Edges are two-way streets that can be serviced in any direction and both sides are serviced simultaneously whereas arcs represent one-way streets or large two-way streets where the two sides cannot be serviced simultaneously. These are represented as two arcs. Arcs can be classified as required or non-required. A required arc has a demand and it is obligatory to service it, where a non-required arc doesn't have a demand and can be traversed as necessary. An example of a residential area, including the arc characteristics is presented in Figure 2.2. The depot node and intermediate facilities, as mentioned in Section 1.2, are also shown in this figure.

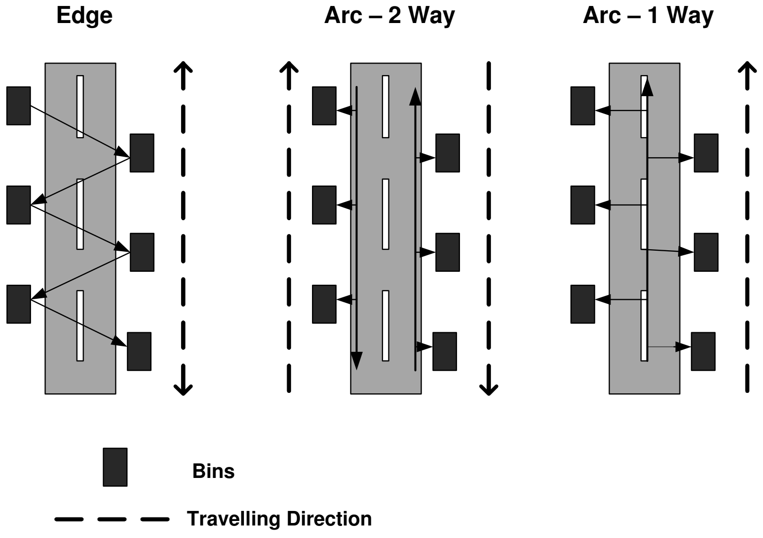

Figure 2.1: Arcs vs Edges

2.2 Arc Routing Problem

Routing problems can either be classified as a Node Routing Problem (points or nodes) or an Arc Routing Problem (arcs or edges). Residential waste collection is defined in literature as an Arc Routing Problem (ARP), as all streets (arcs and edges) have to be serviced [1]. For municipal waste collection, sectoring is combined with the ARP, therefore it is classified as a Sectoring Arc Routing Problem (SARP). The routing within each sector can be defined as a Capacitated Arc Routing Problem (CARP), as each arc consist of a specific demand (capacity).

A limited fleet of identical vehicles is assigned to a certain depot. A vehicle is responsible for the servicing of a specific set of arcs with each arc's capacity known. Each trip starts and ends at a depot. The demand of a trip must not exceed vehicle capacity and an arc is only serviced once.

Due to the complexity of real street networks, the CARP on its own will not be able to model the problem realistically [2]. This can be attributed to the fact that the CARP can only model two-way streets where its sides can be collected in any direction. In real networks, two-way streets have independent sides and one-way streets can only be serviced in one direction. Due to this mixed graph network, the problem can be defined as a Mixed Capacitated Arc Routing Problem (MCARP). The MCARP will be modelled as follows [2]: Let $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E} \cup \boldsymbol{A})$ be the mixed

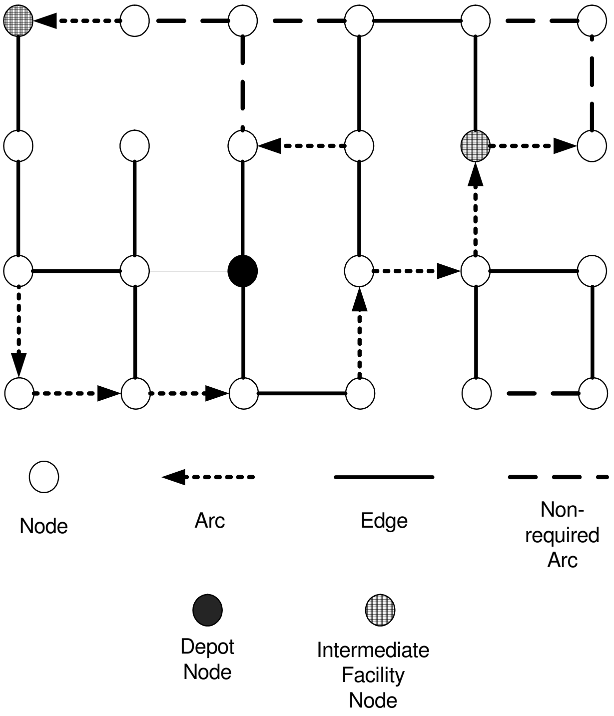

Figure 2.2: Area Presentation

network with a set $\boldsymbol{V}$ of n nodes, an arc-set $\boldsymbol{A}$ and an edge-set $\boldsymbol{E}$. There are K identical vehicles with maximum capacity Q and each vehicle is allocated to a depot node s. The deadheading cost incurred for every time a link u is traversed without being serviced can be defined by $\boldsymbol{C}(u)$. A subset of τ required links has to be serviced, this includes a subset of ϵ required edges, $\boldsymbol{E}_\epsilon \in \boldsymbol{E}$, and a subset of α required arcs, $\boldsymbol{A}_\alpha \in \boldsymbol{A}$. Thus, $\tau = \epsilon + \alpha$. Each link u has a service cost *w(u)* and a demand *r(u)*, with the service cost and demand being non-negative integers.

2.2.1 Heuristic Procedures

The waste collection problem is exceptionally complex as it models the routing of big areas with complex networks and incorporates a vast amount of factors. It also requires an excessive amount of computer time to solve. This problem can thus be defined as being NP-hard (nondeterministic class problem). Due to these facts heuristic procedures have to be applied to get a reasonable solution for the problem. Heuristic procedures are defined as being iterative: values of decision variables are exchanged with various feasible variables in an orderly fashion to reach an acceptable solution [13].

2.3 Sectoring a Service Area

The SARP has been addressed in literature and three approaches are identified to assist in solving the waste collection sectoring problem. The first approach to be investigated is used by Muyldermans et al. [6], where sectoring is applied to salt spreading operations.

2.3.1 Districting for salt spreading operations

Salt spreading on roads need to be improved due to the large amount of trucks needed for the operation and inefficient routing that lead to unnecessary fuel consumption. Salt is spread on roads to make it less slippery when ice, snow or frost occurs. The following steps are followed in order to sector the service area into smaller, more manageable areas: (1) locating depots, (2) partitioning by assigning arcs to depots and (3) routing. Each sector is built around a fixed depot. Four stages are used to build sectors and perform routing simultaneously:

Stage 1: Pre-processing

The road network is partitioned into elemental cycles, where each cycle contains two to three arcs. In this stage the cycle weights (capacity) and ratios (average distance to reach cycle j from depot i) are calculated. These computations indicate the proximity of cycles to the various depots as each sector has a capacity limit and needs to be compact.

Stage 2: Initial partial assignment

In this stage the district building process is initiated by first assigning cycles that are neighbouring to the depot and secondly cycles that are close to the depot (the ratio, as discussed in Stage 1, is smaller than 0.5). This region of assigned cycles is indicated in Figure 2.3. Only suitable cycles are assigned. A cycle is suitable when it has not yet been assigned to another depot and it contains at least one node that has already been assigned to that particular depot.

Stage 3: Two-phase iterative assignment In this phase all districts are built simultaneously by assigning the arcs that are still available to the sectors initiated in Stage 2. The regions where these two phases are applicable are shown in Figure 2.3.

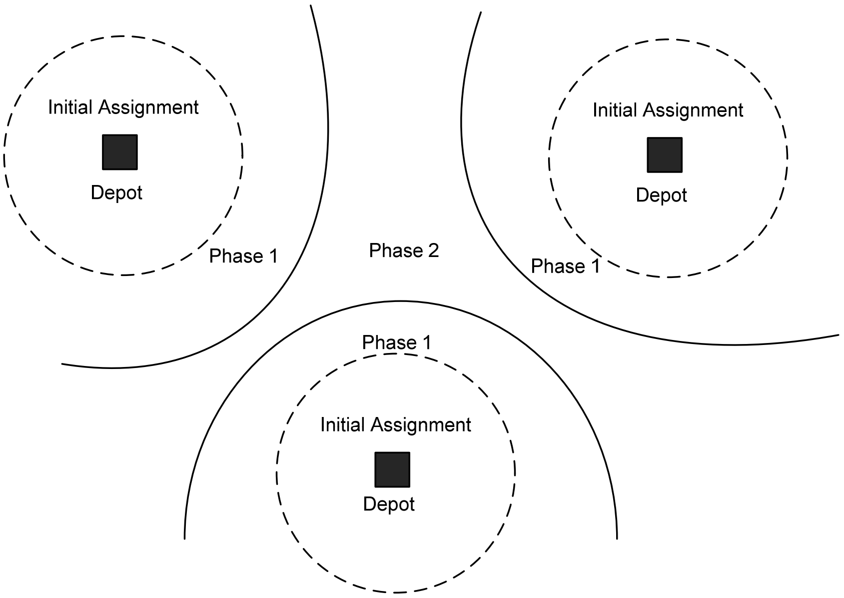

Figure 2.3: Regions for Phases 1 and 2

Stage 4: Improvement and other interaction

In this stage various shifts of cycles (heuristic procedures) are performed to get to an acceptable solution.

As seen in Figure 2.3, the phase one cycles are close to the depot and simple to assign. Up to this point the cycle with the smallest workload has been the next assigned cycle. Phase two starts when there are no more cycles in phase 1 to assign. For phase two, the largest weight cycle with the smallest ratio is the next to be assigned.

In the approaches applied by [6] heuristic procedures are used. The sectoring of salt spreading areas is closely related to waste collection, although in the case of salt spreading, each sector contains a fixed depot. It however shows that sectoring can be applied to a process as big as salt spreading and waste management. With waste collection there are multiple, fixed depots and it is not distributed in such a way that each sector can contain one. Thus the methods of [8] for the sectoring of snow disposal areas will be examined in the next section as the depot locations of snow disposal have the same features as that of waste collection.

2.3.2 Sector design for snow disposal operations

In urban areas snow alongside roads heap up and need to be loaded onto trucks and transported to disposal sites. In the approach followed by Perrier et al. [8], the area to be serviced is partitioned into smaller sub-networks called sectors. All sectors are treated simultaneously and each sector's demand is equal to a single vehicle's capacity. The sectors are designed in such a way that they don't have overlapping areas. Each sector is assigned to a single disposal site and there can be multiple sectors allocated to a single disposal site. The disposal site locations are fixed and have a certain capacity constraint. There are two types of resources, trucks and snowblowers. A snowblower picks up the snow and loads it unto a truck that is travelling alongside it. Once the truck has reached its full capacity, it has to unload the snow at the nearest disposal site. Empty trucks form queues at the disposal sites. Once a truck leaves a snowblower, the truck that is first in line immediately travels to that snowblower. The objective of this method is to prevent a snowblower from being idle. The approach is to first partition the service area into sectors and secondly to assign the various sectors to appropriate disposal sites. The objective is to minimize the number of trucks and the variable costs incurred during snow disposal operations. These costs include transportation cost, operating costs of disposal sites, cost of trucks, fuel expenses and labour costs.

There are many constraints to incorporate with the design and assignment of sectors. Sectors need to be balanced in workload, this implies that more or less the same amount of resources have to be allocated to them and their sizes need to be approximately the same. Balanced sectors will lead to a balance in service times which is desirable. *Basic units* are the units added to a sector in each iteration of sector building.

Two constructive methods are proposed. The basic units in these approaches are single street segments. The sectors are constrained with contiguity, shape, size, hourly and annual capacities of disposal sites and a sector can only be assigned to one disposal site. The unusual big size of the model prevents the model from being solved optimally and thus two constructive heuristics are proposed. The first method is called: *assign first - partition second.* In this method, street segments are assigned to disposal sites prior to the partitioning of the service area into balanced sectors. The second method is called: *partition first - assign second.* With this method, the service area is divided into balanced sectors prior to the assigning of sectors to disposal areas.

The objective of the *sector design for snow disposal operations* method doesn't correlate to that of waste collection, as waste collection focuses on the minimization of the travelling cost and that of snow disposal aims to minimize the number of trucks. It can be seen that heuristic procedures are applied and that it is feasible to sector such a large service area. The next approach to sectoring which is applied to waste collection and proposed by Mourao et al. [5], will be described in the following section.

2.3.3 Sectoring the arc routing problem with heuristics

In this paper, the *Sectoring Arc Routing Problem* (SARP) is defined on a mixed multigrah which models street networks of a service area. The SARP combines the sectoring problem and the arc routing problem. The SARP is applied to waste collection. All vehicles are assigned to one depot node. The objective is to partition the service area into sectors, each sector is assigned to one vehicle, and to solve the MCARP in each sector in such a way that the service time for all trips is minimized. In this approach, two two-phase heuristics (TPH) and one best insertion method (BIH) are proposed and tested. With the TPH, phase one constructs sectors by making use of two possible heuristics, whereas in phase two, the trips in each sector are computed by solving a MCARP. With the BIH, sectors and trips are built simultaneously. The three methods are compared by their cost gap (%), imbalance and dispersion measures.

Two two-phase heuristics

The SARP and partitioning problem is complex. Moreover, the service areas contain hundreds of segments with various demands and constraints. Thus it is difficult to solve the problem with exact algorithms. Due to the level of complexity, two heuristic methods are proposed. The first heuristic determines sectors and the second heuristic computes the routing in each sector. The two heuristics used with the TPH are (1) the *Circuit of Tasks Heuristic* (CTH) and (2) the *Single Task Heuristic* (STH).

The CTH and STH can be described as follows: A link is defined as an edge or arc that has known demands and service durations. Sectors are initialized by identifying a required link, called the *seed-task* in each sector, where the distance, called the *U-distance*, between the set of K seed-tasks are maximized. The first task is the task farthest from the depot. The K-1 other tasks are selected in such a way that the minimum distance to the already chosen seeds is maximized. Each sector has a maximum workload L. With the process of sector expansion, tasks are only allocated to sectors that have not reached their full capacity and will skip sectors that have reached maximum capacity. The only difference between these two methods is with the building of sectors. With the CTH, a *circuit* of tasks is added to a sector in each iteration whereas only a *single* task is added with the STH. Because of trips only built in the second phase, a workload estimate is used for sectoring in the first phase. It is updated once a trip is added to sector k according to the cost variation derived from the insertion of that task into a position on an imaginary trip of k. These trips are only valid for phase one and are therefore called *imaginary trips*. Sectors are built simultaneously in order to promote balanced sectors. Balanced sectors are sectors with approximately the same number of resources allocated to them and more or less the same time requirements for servicing. In each iteration, a task is added to the sector with the smallest workload estimate.

These methods are tested on waste collection, a real life application. It can be seen that heuristic procedures are used and that it is possible to implement such procedures in the sectoring of an ARP. These heuristics are applied to an area with one depot node and no Intermediate Facilities (IF). These methods will thus be adjusted to incorporate IFs.

2.4 Conclusion

It was found that there is an urgent need for improving the management of municipal solid waste in South Africa. Three methods were examined and it was decided that the methods used by Mourao et al. [5] are appropriate. Certain characteristics of the models used in Mourao et al. [5] will be adapted for compatibility with the characteristics of waste collection in South Africa. The approach used, as well as the changes needed for the method to collaborate with the addressed problem will be discussed in-depth in the next chapter.

Chapter 3

Models for the SARP

3.1 Problem Modelling and Notation

The sectoring problem can be modelled by a mixed multigraph $\boldsymbol{T}$. This graph contains m links (tasks) which can be divided into required and non-required tasks. A subset of τ required tasks have to be serviced. Service vehicles are constrained by a maximum working time L and a limited capacity W. The maximum working time is sufficient to allow any vehicle to reach a required link, service it, and return to the depot. The amount of required arcs are presented by α and the required edges by ϵ respectively, thus $\boldsymbol{T}$ contains $\tau = \alpha + \epsilon$ required links.

3.1.1 Network model

The solution approach of the SARP can be simplified by transforming $\boldsymbol{T}$ into a directed multigraph $\boldsymbol{G} = (\boldsymbol{N},\boldsymbol{A})$. Each required edge is presented as two seperate arcs [3]. The required arcs are listed from 1 to $|\boldsymbol{A}|$. Each arc u in $\boldsymbol{A}$ is defined by a begin-node *b(u)*, end-node *e(u)* and a deadheading time d_u. The arcs in $\boldsymbol{G}$ will be presented by $\boldsymbol{R} = \alpha + 2\epsilon$. Each arc has a collection time t_u, a demand q_u and a pointer $inv(u)$. Each edge is coded as two seperate arcs u and v, such that $q_u = q_v$, $t_u = t_v$, $d_u = d_v$, $inv(u) = v$, $inv(v) = u$. The pointers ensures that when a edge is serviced, both u and v are marked in the algorithm. The minimum travel time from u to v can be defined as U_{uv}.

3.1.2 Modelling of sectors

There is a number of K sectors and $\boldsymbol{R}_k$ is the set of tasks assigned to sector k, where $k = 1, 2, \ldots, K$. The cost of sector k, cost(k), is defined by the total duration of its trips. Each sector has one vehicle for servicing, the total cost of its trips must not exceed the time limit L.

The sectors are initialized with a single required arc of $\boldsymbol{R}$, called the *seed-task*. These tasks are selected through the MaxDist rule, as defined by Equation 3.1, where the U-distance between the K seed-tasks are maximized.

$$\begin{aligned} MaxDist = \max(\min\{&U_{uv}, U_{u,inv(v)}, U_{inv(u),v}, U_{inv(u),inv(v)}, U_{vu}, U_{v,inv(u)}, \\ &U_{inv(v),u}, U_{inv(v),inv(u)}\}) \end{aligned} \tag{3.1}$$

The first seed is the task furthest from the depot, where the subsequent seeds are selected in such a way that the minimum distance between the seeds already chosen is maximized. The maximum workload in a sector should not be exceeded. When a sector has reached maximum workload, this sector is *closed*, otherwise it is *open*. Sectors cannot be expanded when they are *closed*. The sector with the smallest workload is selected for expansion.

Workload Estimate

With the TPH, a workload estimate (WE) is used throughout the sector expansion. The WE is computed by estimating the insertion position of the selected task i into trip r after task b in such a way that the route cost is minimized. The trips in this phase are called imaginary trips as they are only valid during phase 1. The insertion position is estimated by making use of Equations 3.2 and 3.3.

$$C(r, i, b) = \text{Route cost of inserting task } i \text{ into trip } r \text{ after task } b \tag{3.2}$$

$$\text{Position}(b) = \text{The position of } b \text{ where C(r,i,b) is minimized} \tag{3.3}$$

The workload is computed by calculating the cost of trip r with a RouteCost Algorithm 1. This algorithm takes the sequence of tasks as in input and estimates the cost of the route by combining the service cost of each task (line 3) and the cost of travelling from task i to $i + 1$ (line 4). With the Exact Cost Heuristic and Single Task Heuristic, the exact workload is known at all times as it is computed throughout the sector-building process.

Algorithm 1 ROUTECOST

```
INPUT THE SEQUENCE OF TASKS
for $i = 1$ to (length of sequence - 1) do
    service cost += service cost of task $i$
    route cost += $U_{i,i+1}$
end for
cost = service cost + route cost
```

3.2 The Two-phase heuristic

As the name states, the two-phase heuristic is divided into two phases. In the first phase, the required arcs are partitioned into sectors with a WE smaller than L. In the second phase, each sector's arcs are modelled and solved as a mixed CARP, this model is developed by Willemse and Joubert [12]. The sectoring method in the first phase is called the *Single Task Heuristic* (STH). It expands sectors by adding one task at a time.

Phase 1: Single Task Heuristic The sectors are expanded by adding a task that is closest to the seed-task. First, each sector is initialized by a seed-task and then the sector with the minimum WE is chosen for expansion. If the workload maximum is not exceeded when adding the selected task to the sector, it is expanded. Otherwise, the sector is closed and it can no longer be expanded. If all the sectors are closed and all tasks are not assigned, the number of sectors is incremented and all the steps are repeated. The algorithm stops when all tasks are allocated. The STH algorithm is displayed in Algorithm 2.

Algorithm 2 description:

In line 1 the initial number of sectors, K, is an input. The algorithm is repeated (lines 2 to 23) until all arcs are assigned to sectors. The for-loop from line 4 to 9 initializes a WE and a seed-task for each sector. The seed-task is selected by using the MaxDist function, as defined by equation 3.1. In line 8, the tasks selected as seed-tasks are removed from the required arc list ($\boldsymbol{R}(k)$), it is added to the appropriate sector's arc list ($\boldsymbol{R}$') and the WE of each sector is updated by calculating the cost of the trip when the seed-task is added. This cost is computed by a RA, developed by Willemse and Joubert [12]. After the sectors are initialized, they are expanded by adding one task at a time. The expansion

Algorithm 2 SINGLE TASK HEURISTIC

```
INITIALIZE K OPEN SECTORS
repeat
  Define R' as the set of required arcs not yet assigned
  for $k = 1$ to K do
    initialize a WE(k) for sector(k)
    R(k) = 0
    select a seed-task for sector(k)
    update R(k), R' and WE(k)
  end for
  EXPAND SECTORS
  Define a set of open sectors
  repeat
    select the sector with the minimum WE
    select the task closest to the seed-task
    if WE of selected sector + cost of adding the selected task $\geq L$ then
      add the selected task to sector(k)
      update R(k), R' and WE(k)
    else
      close the selected sector
    end if
  until no more required arcs to assign or all sectors are closed
  increment K if all tasks are not assigned
until all tasks are assigned
```

is executed by repeating lines 2 to 23 until all arcs are assigned or all sectors are closed. In line 13, the sector with the smallest WE is selected for expansion and in line 14, the task closest to the selected sector's seed-task is picked. The WE is computed by inserting the selected task in a best insertion manner (Section 3.3) and computing the route cost (same as line 8 cost calculation). If the WE is smaller than L (maximum workload), then the task is added to the sector and again, the $\boldsymbol{R}(k)$, $\boldsymbol{R}$' and WE(k) are updated as with line 8. If the expansion of the sector causes the WE(k) to exceed the maximum cost, the sector is *closed* and can no longer be expanded. If there are no more *open* sectors and there are still unassigned tasks, K is incremented and the loop will restart at line 2, otherwise the algorithm stops.

Algorithm 3 TWO-PHASE HEURISTIC

```
input K
repeat
   PHASE 1: SECTORING
   call: Single Task Heuristic
   PHASE 2: TRIP CONSTRUCTION
   repeat
      for $k = 1$ to K do
         call the routing algorithm
         if cost(k) $>$ L then
            failure = true
         else
            k = k + 1
         end if
      end for
   until k = K + 1 or failure
   if failure then
      increment K
   end if
until not failure
```

Phase 2: Routing In the second phase, the arcs within each sector is routed. The routing algorithm (RA) used in this phase is developed by Willemse and Joubert [12]. Phase 2 of the TPH is presented in Algorithm 3.

Algorithm 3 description:
Algorithm 2 runs in Phase 1 (lines 1 to 4), where K sectors are built. In Phase 2 (lines 5 to 15), each sector is routed (RA developed by Willemse and Joubert [12]). If the cost of a sector exceeds the limit, K is incremented. This algorithm will run until no sector cost exceeds K.

3.3 Best Insertion Heuristic

The Best Insertion Heuristic (BIH) method builds sectors and trips simultaneously. The exact workload is estimated by the same method used to build the imaginary trips of the STH. The BIH is exactly the same as the STH in the TPH. The only difference between the TPH and the BIH is that the imaginary trips of the TPH is the actual trips of the BIH.

3.4 Exact Cost Heuristic

As the BIH, the Exact Cost Heuristic (ECH) builds trips and sectors simultaneously. The only difference between the BIH and the ECH is the way the cost of the sector is computed when adding a task. With the BIH, an algorithm developed by Willemse and Joubert [12] is used to compute the trip cost when the sequence of tasks is given as an input. The trip cost of the ECH is computed by making use of the routing algorithm, also developed by Willemse and Joubert [12], to work out the best route with the given tasks as an input (not the sequence of the tasks) and provides the trip cost as an output. The ECH is described in Algorithm 4.

3.5 Intermediate Facilities

The methods used by Mourao et al. [5]'s algorithms only computes sectors and routes for service areas containing a single depot node. In South Africa, service areas contain Intermediate Facilities (described in Section 1.2). Willemse and Joubert [12] developed a RA incorporating IFs. This algorithm takes a list of tasks as an input and gives the route, cost and number of trips to and from the IFs as an output. This RA will be tested on Phase 2 of the TPH and the cost calculations and final routing

Algorithm 4 EXACT COST HEURISTIC

```
INITIALIZE K OPEN SECTORS
repeat
    Define R' as the set of required arcs not yet assigned
    for $k = 1$ to K do
        initialize a cost(k) for sector(k)
        R(k) = 0
        select a seed-task for sector(k)
        update R(k), R' and cost(k)
    end for
    EXPAND SECTORS
    Define a set of open sectors
    repeat
        select the sector with the minimum cost
        select the task closest to the seed-task
        if Cost of selected sector + the cost of adding the selected task to the sector $\geq L$ then
            add the selected task to sector(k)
            update R(k), R' and cost(k)
        else
            close the selected sector
        end if
    until no more required arcs to assign or all sectors are closed
    increment K if all tasks are not assigned
until all tasks are assigned
```

of the ECH. BIH will also be tested on IFs. This will be achieved by adding the IF arc into the sequence of tasks when the service vehicle has reached full capacity. With the lpr-IF problems, the vehicle has to visit an IF before returning to the depot. An IF arc will thus be added at the end of the sequence of tasks.

3.6 Conclusion

Three different approaches to sector a service area were developed and examined. These models will be tested on benchmark problems and the results will be presented and evaluated in the next chapter.

Chapter 4

Computational Results

4.1 Evaluation criteria

The TPH, BIH and ECH will be evaluated by certain criteria. The main evaluation will be to compare the total duration of trips over the K sectors. There are also secondary criteria that will be used in the selection of the most appropriate method, these criteria can be described as follows:

1. **Cost gap** The cost gap of solution X can be defined as the deviation of cost(X) to the lower bound (LB). The LB, developed by Belenguera et al. [2], is primarily used for the MCARP, although it might be weaker when used for the SARP, it remains valid. The LB is the theoretical optima of the problem, thus is a smaller gap preferable. The cost gap in percentage is defined as:

$$gap(X) = ((cost(X) - LB)/LB) * 100 \tag{4.1}$$

2. **Diameter** This criteria concerns the shape of sectors. The diameter of a sector is calculated by the maximum distance (U-distance) between two of its tasks:

$$diam(k) = max\{U_{uv} | u, v \epsilon k\} \tag{4.2}$$

The diameter of a partition(S) is measured by the maximum diameter of its sectors:

$$diam(S) = max\{diam(k) | k \epsilon S\} \tag{4.3}$$

3. **Dispersion measure** This measure defines the compactness of sectors by evaluating the mean distance from all arcs in the sector to the seed-task. It can be defined as follows:

$$u_k = (1/|R_k|) \sum_{u \in R_k} U_{ua} \tag{4.4}$$

$$\tag{4.5}$$

4. **Imbalance** This measures the imbalance of cost over all sectors in solution X. It is defined as the difference between the minimum and the maximum sector costs:

$$imbal(X) = max\{cost(k)|k \epsilon S\} - min\{cost(k)|k \epsilon S\} \tag{4.6}$$

4.2 Instances and parameters

Mourao et al. [5] tests the developed models by solving 15 small benchmark problems, called *lpr* files. One of these files are shown in Appendix A as an example. The raw data of these problems were reworked with algorithms developed by Willemse and Joubert [12] to produce some of the input data required for sectoring. The algorithms described in Chapter 3 will be tested on these benchmark problems and the input data acquired by Willemse and Joubert [12] will be incorporated. The mathematical symbols used in the algorithms are presented in Table 4.1.

Table 4.1: Mathematical symbols

Symbol	**Description**
R'	Set of required arcs not yet assigned
K	No of sectors
k	Sector index
$WE(k)$	Workload estimate used in the STH and BIH
$R(k)$	Set of arcs allocated to sector k
L	The maximum service cost allowed per sector
W	Maximum load per vehicle
cost(k)	The service cost of sector k
u, v	Arc indexes
U_{uv}	The minimum cost for travelling from arc u to v

The benchmark problems consist of three groups of data sets, namely: Group *a* that represents modern towns where a majority of the streets are edges and two sides are serviced separately; Group *b* represents historical town centres where there are a lot of one-way streets; and Group *c* is low-traffic suburban areas that contain a large amount of two-way streets where both sides can be serviced simultaneously. The features of these benchmark problems are presented in Table 4.2.

Table 4.2: Benchmark features

File	**Links**	**Required links**	**Required edges**	**Required arcs**	**Depot**
lpr-a-01	94	52	0	52	C
lpr-a-01	94	52	0	52	C
lpr-a-02	169	104	5	99	P
lpr-a-03	469	304	33	271	C
lpr-a-04	651	503	34	469	P
lpr-a-05	1056	806	58	748	P
lpr-b-01	63	50	5	45	C
lpr-b-02	117	101	9	92	C
lpr-b-03	361	305	26	279	C
lpr-b-04	582	501	8	493	P
lpr-b-05	876	801	37	764	P
lpr-c-01	52	50	39	11	P
lpr-c-02	101	100	77	23	P
lpr-c-03	316	302	241	61	P
lpr-c-04	604	504	362	142	C
lpr-c-05	841	803	287	416	C

The depot is located in either a central (C) or peripheral (P) manner. The demand of a certain arc is measured in kg and the collecting and deadheading times are given in seconds. All benchmark groups consist of the following attributes: a maximum vehicle capacity of 10,000 kg; a dumping time of 300 seconds and a maximum working time of 21,600 seconds per day.

4.3 Comparisons between the heuristics

The average and worst values of the cost gap (%), imbalance, diameter and U_k values for the three heuristics are presented in Tables 4.3 and 4.4 (IF files). In Tables 4.5

Table 4.3: Evaluation criteria results of lpr files

		TPH	**BIH**	**ECH**
Cost gap (%)	Avg	7.31	27.04	**5.90**
	Worst	18.00	47.90	**14.90**
Imbalance	Avg	274.40	**264.07**	287.40
	Worst	632.00	**613.00**	647.00
Diameter	Avg	457.20	**441.87**	457.73
	Worst	736.00	736.00	736.00
U_k	Avg	**159.58**	161.46	159.87
	Worst	**266.00**	273.50	274.90

Table 4.4: Evaluation criteria results of lpr-IF files

		TPH	**BIH**	**ECH**
Cost gap (%)	Avg	7.36	24.77	**6.14**
	Worst	17.40	42.70	**14.41**
Imbalance	Avg	1145.70	262.67	**232.60**
	Worst	2638.00	670.00	**591.00**
Diameter	Avg	457.20	**322.07**	457.70
	Worst	736.00	736.00	736.00
U_k	Avg	159.60	**147.13**	158.30
	Worst	266.00	275.40	**261.35**

and 4.6, the total cost, imbalance, diameter and number of sectors of each benchmark problem (IF presented in Table 4.6) is displayed for the three heuristics separately, where 1 represents the TPH, 2 represents the BIH and 3 represents the ECH.

Cost gap

The gap percentages are presented in Tables 4.3 and 4.4. When looking at the lpr-files, the ECH has the lowest average and worst percentages. The TPH's percentages are close to that of the ECH. The BIH's percentages are very high, for both lpr and lpr-IF files, which indicates that the cost of its sectors are far from optimal. With the lpr-IF files, the ECH still has the lowest percentage and the TPH differs with only 1.22%. The TPH and ECH's worst values are also close. The small average cost gap % of the sectors constructed with the ECH and TPH indicates that the sector costs are close to optimal.

Table 4.5: Results for lpr files

File	**Cost**			**Imbalance**			**Diameter**			**No. Sectors**		
	1	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
a-01	13484	15172	13484	0	0	0	182	94	182	1	1	1
a-02	29664	36386	**29516**	22	22	38	309	297	309	2	2	2
a-03	81759	93126	**79664**	407	315	242	488	471	488	5	5	4
a-04	139869	164364	**137527**	425	407	387	500	503	500	8	8	7
a-05	230844	281593	**226078**	430	313	333	723	701	723	13	13	11
b-01	14839	17946	14839	0	0	0	197	120	197	1	1	1
b-02	29464	35512	**29699**	314	314	125	334	339	334	2	2	2
b-03	83541	101443	**81732**	316	316	348	553	541	544	5	5	4
b-04	145641	182669	**140464**	303	344	486	613	613	613	9	9	7
b-05	247440	310453	**241054**	399	309	647	736	736	736	15	15	13
c-01	18808	21245	18808	0	0	0	175	177	175	1	1	1
c-02	37390	43116	**37261**	6	6	125	301	293	301	2	2	2
c-03	120106	136508	**117751**	289	400	492	506	501	506	7	7	6
c-04	180921	206686	**179753**	632	602	577	556	589	567	10	10	9
c-05	284931	328372	**279641**	573	613	520	685	653	691	16	16	14

Imbalance

The imbalance, displayed in Tables 4.3 and 4.4, between sectors is an important criterion as there needs to be a workload balance between sectors to reduce idle time or the overworking of service vehicles. When looking at the lpr-files, the differences between the average and worst imbalance values of the three methods are minor. The BIH has the smallest average and worst value. This indicates that the sectors' costs of the three methods are close to being balanced. The lpr-IF files solved with the TPH have exceptionally high values, whereas with the BIH and ECH, it is approximately the same as with the lpr-files. The ECH has the lowest imbalance when solving the IF problems.

Diameter

The diameter results, presented in Tables 4.3 and 4.4, shows a minor difference between the three heuristics for both lpr and lpr-IF files. The BIH's sectors has the best average values, especially for the lpr-IF files.

Table 4.6: Results for lpr_IF files

File	**Cost**			**Imbalance**			**Diameter**			**No. Sectors**		
	1	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
a-01	13589	14945	13589	0	0	0	182	94	182	1	1	1
a-02	**29403**	35453	29408	897	23	16	309	297	309	2	2	2
a-03	82202	92582	**80240**	513	355	155	488	471	488	5	5	4
a-04	141149	163282	**138257**	1285	477	261	500	503	500	8	8	7
a-05	229348	270378	**224996**	2638	344	304	723	701	723	13	13	11
b-01	14876	17646	14876	0	0	0	197	120	197	1	1	1
b-02	**29609**	35089	29749	199	201	137	334	339	334	2	2	2
b-03	83117	100041	**81776**	1772	136	333	553	541	544	5	5	4
b-04	145422	177763	**140815**	1617	491	362	613	613	613	9	9	7
b-05	246241	299512	**240024**	1574	334	516	736	736	736	15	14	12
c-01	18770	20763	18770	0	0	0	175	177	175	1	1	1
c-02	37108	42201	37108	56	5	56	301	293	301	2	2	2
c-03	119130	132335	**117307**	3051	374	320	506	501	506	7	7	6
c-04	181326	205723	**180498**	1538	670	438	556	589	567	10	10	9
c-05	290172	333338	**286758**	2045	530	591	685	653	691	16	16	14

Dispersion measures

The differences in average and worst values of the three methods for both lpr and lpr-IF files are insignificant. The sectors for all the methods are very compact, as their worst values only range between 266.00 and 275.40.

Results

The total cost, imbalance, diameter, dispersion measures and the number of sectors are presented in Tables 4.5 and 4.6. In almost all instances, the ECH shows to result in the lowest cost. The BIH's sectors have the highest costs for both lpr and lpr-IF files, where most of the sectors constructed by the TPH's costs were close to that of the ECH. The ECH produced the least sectors, where the TPH and BIH resulted in the same number of sectors for all the benchmark problems. This indicates that less service vehicles will be needed when using the ECH.

4.4 Fabricated Case Study

The research question, defined in Section 1.3, stated that the collection area has to be partitioned into collection days. To accomplish this, another sectoring algorithm was developed (Algorithm 5). This algorithm will only be tested on the lpr-05 problems, as the other problems are not big enough to partition into days. In comparison to the other methods, the ECH built sectors which resulted in the least cost. South African waste collection areas contains IFs, thus the CDA will only be tested on the results achieved by the ECH-IF. These results are shown in Table 4.8.

Table 4.7: CDA: Results

File	**Day**	**No. sectors**	**Cost**
lpr-a-05	1	3	61148
	2	2	40853
	3	2	41127
	4	2	41108
	5	2	40903
lpr-b-05	3	3	59907
	2	3	60081
	3	2	40040
	4	2	40044
	5	2	39733
lpr-c-05	1	3	60948
	2	3	61605
	3	3	61022
	4	3	61029
	5	2	40814

Collection Days Algorithm This algorithm assigns the sectors computed with the ECH to a specific collection day. The first collection day service area is initiated by selecting the sector with the minimum cost. The other collection day areas are then selected in such a way that the distances between their seed-tasks are maximized. These areas are then expanded by adding one sector at a time. The sectors are added in such a way that the U-distance between the seed-tasks are minimized. The algorithm can be described as follows:

Algorithm 5 description:

In line 1, five collection days are initiated and their costs and seed-tasks is an input (line 2). The collection day areas are initialized by adding the sector with the minimum cost to the first day. The four other days are initiated by maximizing the minimum distance between the sector's seed-tasks. The cost, sector list and seed-task of each sector is initialized by the for loop (lines 4 to 8). In lines 10 to 14, the collection day areas are expanded by adding the sector that minimizes the distance to the seed-tasks of the sectors already added to that day. This is repeated until all sectors are assigned to days, then the algorithm stops.

Algorithm 5 COLLECTION DAYS ALGORITHM

```
 1: INITIALIZE 5 COLLECTION DAYS
 2: Input the costs and seed-tasks of the sectors
 3: Select the sector with the minimum cost
 4: for $k = 1$ to No. of sectors do
 5:     initialize a cost and sector list for day(k)
 6:     select a seed-task for day(k)
 7:     update the cost and sector list
 8: end for
 9: EXPANSION
10: repeat
11:     select the day with the minimum cost
12:     expand by adding the sector closest to the seed-tasks of the sectors already
        added to day(k)
13:     update the cost and sector list of each day
14: until all sectors are assigned to days
```

4.4.1 CDA Evaluation

The CDA results are presented in Table 4.7. The sectors computed with the ECH are assigned to specific collection days. The amount of sectors assigned to each day, as well as the total cost per day is shown. These results also give an indication of the number of service vehicles needed per day and per week as a sector is serviced by one vehicle in a day.

4.5 Conclusion

The aim of this project is to partition a waste collection service area into more manageable steps. Three appropriate heuristic methods were identified and developed. These methods were tested on benchmark problems and the results evaluated by certain criteria. The objective of sectoring is to minimize overall costs and to acquire balanced sectors which subsequently means that the two most important criterions are the total cost per sector and the sector imbalance. The evaluation showed that the areas solved by the Exact Cost Heuristic results in the least cost, least number of sectors and the most balanced sectors. Thus the ECH is the most appropriate method for the sectoring of a waste collection area. Another algorithm was developed to partition the service area into collection days and was tested on three benchmark problems. When comparing the current approach used by local municipalities, to sector service areas, with the method proposed, the following improvements was achieved: sectors are more balanced; service vehicles will experience minimum overwork and idle time; the total service cost is less due to the vehicle drivers knowing the exact route to service and the number of service vehicles required per day is known. This project will thus be beneficial to municipalities.

Table 4.8: ECH-IF: Results

File	**No. sectors**	**Sector Costs**													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
a-05	11	20366	20487	20506	20621	20416	20333	20430	20338	20605	20577	20317			
b-05	12	19969	20041	20071	20022	19764	19797	20160	19983	20280	20155	19829	19953		
c-05	14	20335	20612	20416	20278	20479	20577	20271	20714	20356	20862	20274	20552	20634	20398

Bibliography

[1] Abdelkader, S. and Eglese, R. W. (2010). Combinatorial Optimization and Green Logistics. *Annual Operations Research*, 175(1):159–175.

[2] Belenguera, J., Benaventa, E., Lacommeb, P., and Prins, C. (2006). Lower and upper bounds for the mixed capacitated arc routing problem. *Computers & Operations Research*, 33(12):3363–3383.

[3] Lacomme, S., Prins, C., and Ramdane-Cherif, W. (2001). A genetic algorithm for the capacitated arc routing problem and its extensions. *Applications of Evolutionary Computing, Lecture Notes in Computer Science*, 2037(4):473 – 483.

[4] Mogale City (2003). Solid Waste Management in Mogale City Local Municipality. Technical report, Mogale City Local Municipality.

[5] Mourao, M. C., Nunes, A. C., and Prins, C. (2009). Heuristic methods for the sectoring arc routing problem. *European Journal of Operational Research (Netherlands)*, 196:856–868.

[6] Muyldermans, L., Cattrysse, D., Van Oudheusden, D., and Lotan, A. (2002). Districting of salt spreading. *European Journal of Operational Research*, 139(3):521–532.

[7] Palmer Development Group (2007). *Assessment of the state of waste service delivery and capacity at the local government level. Available online form http://www.info.gov.za/otherdocs/2007/waste-assessment.pdf.*

[8] Perrier, N., Langevin, A., and Campbell, J. F. (2008). The sector design and assignment problem for snow disposal operations. *European Journal of Operational Research*, 189(2):508–525.

[9] Randin, K. B. and Maynard, H. B. (2001). *Maynard's Industrial Engineering Handbook*. McGraw-Hill Professional, 5th edition.

[10] Tavares, G., Zsigraiova, Z., Semiao, V., and da Graca Carvalho, M. (2008). A case study of fuel savings through optimisation of MSW transportation routes. *Management of Environmental Quality: An International Journal*, 19:444–454.

[11] Toth, P. and Vigo, D. (2002). *The Vehicle Routing Problem*. PA: SIAM.

[12] Willemse, E. J. and Joubert, J. W. (2010). Waste collection routing optimisation. Technical report, CSIR: Built Environment, Logistics and Quantitative Methods.

[13] Winston, L. W. and Venkataramanan, M. (2003). *Introduction to Mathematical Programming*, volume 1. Curt Hinrichs, 4th edition.

Appendix A

NAME : Lpr-a-01.dat

NODES : 28

REQ_EDGES : 0

NOREQ_EDGES : 0

REQ_ARCS : 52

NOREQ_ARCS : 42

VEHICLES : 2

CAPACITY : 10000

DUMPING_COST : 300

MAX_TRIP : 28800

LIST_REQ_ARCS :

(2,14) serv_cost 274 trav_cost 34 demand 240

(2,3) serv_cost 156 trav_cost 19 demand 137

(2,8) serv_cost 299 trav_cost 20 demand 279

(4,3) serv_cost 284 trav_cost 20 demand 264

(4,5) serv_cost 287 trav_cost 22 demand 265

(4,9) serv_cost 346 trav_cost 21 demand 325

(5,4) serv_cost 50 trav_cost 22 demand 28

(6,11) serv_cost 304 trav_cost 19 demand 285

(7,14) serv_cost 172 trav_cost 22 demand 150

(7,8) serv_cost 124 trav_cost 42 demand 82

(7,12) serv_cost 64 trav_cost 17 demand 47

(8,13) serv_cost 210 trav_cost 23 demand 187

(9,4) serv_cost 122 trav_cost 21 demand 101

(9,10) serv_cost 143 trav_cost 21 demand 122

(9,15) serv_cost 171 trav_cost 18 demand 153

(10,5) serv_cost 106 trav_cost 20 demand 86

(11,10) serv_cost 323 trav_cost 18 demand 305

(11,17) serv_cost 84 trav_cost 18 demand 66

(12,7) serv_cost 214 trav_cost 17 demand 197

(12,13) serv_cost 103 trav_cost 39 demand 64

(12,18) serv_cost 174 trav_cost 22 demand 152

(13,12) serv_cost 280 trav_cost 39 demand 241

(1,3) serv_cost 246 trav_cost 41 demand 205

(1,13) serv_cost 325 trav_cost 19 demand 306
(15,21) serv_cost 319 trav_cost 18 demand 301
(16,10) serv_cost 109 trav_cost 21 demand 88
(16,15) serv_cost 400 trav_cost 21 demand 379
(16,17) serv_cost 391 trav_cost 19 demand 372
(17,11) serv_cost 395 trav_cost 18 demand 377
(17,16) serv_cost 418 trav_cost 19 demand 399
(17,23) serv_cost 405 trav_cost 15 demand 390
(18,19) serv_cost 383 trav_cost 35 demand 348
(18,24) serv_cost 259 trav_cost 21 demand 238
(19,13) serv_cost 234 trav_cost 18 demand 216
(19,18) serv_cost 410 trav_cost 35 demand 375
(19,20) serv_cost 45 trav_cost 19 demand 26
(19,25) serv_cost 50 trav_cost 18 demand 32
(20,1) serv_cost 221 trav_cost 19 demand 202
(20,21) serv_cost 332 trav_cost 19 demand 313
(20,26) serv_cost 153 trav_cost 14 demand 139
(21,20) serv_cost 286 trav_cost 19 demand 267
(21,27) serv_cost 381 trav_cost 17 demand 364
(22,21) serv_cost 31 trav_cost 20 demand 11
(22,23) serv_cost 168 trav_cost 21 demand 147
(22,27) serv_cost 124 trav_cost 26 demand 98
(24,25) serv_cost 435 trav_cost 41 demand 394
(25,24) serv_cost 327 trav_cost 41 demand 286
(25,26) serv_cost 77 trav_cost 18 demand 59
(26,20) serv_cost 97 trav_cost 14 demand 83
(26,27) serv_cost 282 trav_cost 20 demand 262
(27,22) serv_cost 424 trav_cost 26 demand 398
(27,28) serv_cost 423 trav_cost 39 demand 384
LIST_NOREQ_ARCS : (14,2) cost 34
(14,7) cost 22
(3,2) cost 19
(3,4) cost 20
(3,1) cost 41
(5,6) cost 21
(5,10) cost 20

(6,5) cost 21

(8,2) cost 20

(8,7) cost 42

(10,9) cost 21

(10,11) cost 18

(10,16) cost 21

(11,6) cost 19

(13,8) cost 23

(13,1) cost 19

(13,19) cost 18

(1,15) cost 14

(1,20) cost 19

(15,9) cost 18

(15,1) cost 14

(15,16) cost 21

(15,20) cost 27

(17,22) cost 26

(18,12) cost 22

(20,15) cost 27

(20,19) cost 19

(20,25) cost 24

(21,15) cost 18

(21,22) cost 20

(22,17) cost 26

(23,17) cost 15

(23,22) cost 21

(23,28) cost 19

(24,18) cost 21

(25,19) cost 18

(25,20) cost 24

(26,25) cost 18

(27,21) cost 17

(27,26) cost 20

(28,23) cost 19

(28,27) cost 39

DEPOT : 1 ; DUMPING_SITES : 14,28