universität wien

# DISSERTATION

Titel der Dissertation

## „Vehicle Routing with Multi-Dimensional Loading Constraints“

Verfasser

Mag. Günther Füllerer

angestrebter akademischer Grad

Doktor der Sozial- und Wirtschaftswissenschaften
Dr. rer. soc. oec.

Wien, im Juli 2008

| | |
|---|---|
| Studienkennzahl lt. Studienblatt | A 084 157 |
| Dissertationsgebiet lt. Studienblatt | Internationale Betriebswirtschaft |
| Betreuer | O.Univ.Prof. Dipl.-Ing. Dr. Richard F. Hartl |

# Contents

# List of Figures

# List of Tables

# List of Algorithms

# List of Abbreviations

1DBPP ......... One-Dimensional Bin Packing Problem

2|RN|L ......... Two-Dimensional Rear Non-Oriented Loading

2|RO|L ......... Two-Dimensional Rear Oriented Loading

2|UN|L ......... Two-Dimensional Unrestricted Non-Oriented Loading

2|UO|L ......... Two-Dimensional Unrestricted Oriented Loading

2D-1-VLP ...... Two Dimensional One Vehicle Loading Problem

2DBPP ......... Two-Dimensional Bin Packing Problem

2DSPP ......... Two-Dimensional Strip Packing Problem

2L-CVRP ...... Capacitated Vehicle Routing Problem with Two-Dimensional Loading Constraints

3D-1-VLP ...... Three Dimensional One Vehicle Loading Problem

3DBPP ......... Three-Dimensional Bin Packing Problem

3DSPP ......... Three-Dimensional Strip Packing Problem

3L-CVRP ...... Capacitated Vehicle Routing Problem with Three-Dimensional Loading Constraints

ACO ........... Ant Colony Optimization

BPP ........... Bin Packing Problem

CVRP .......... Capacitated Vehicle Routing Problem

CLP ........... Container Loading Problem

CVRPTW ...... Capacitated Vehicle Routing Problem with Time Windows

DARP .......... Dial-A-Ride Problem

DCVRP ........ Distance-Constrained VRP

DP ............. Dynamic Programming

GA ............. Genetic Algorithm

GAP ........... General Assignment Problem

GTS ........... Guided Tabu Search

$HL$ ............ Loading heuristic for the MP-VRP

L-VRP ......... Loading Vehicle Routing Problem

LIFO ........... Last In First Out

MA ............ Memetic Algorithm

MP-1-VLP ..... Multi Pile One Vehicle Loading Problem

MP-VRP ....... Multi-Pile Vehicle Routing Problem

$P||C_{max}$ ........ Parallel Processor Scheduling Problem

PDP ........... Pickup and Delivery Problem

PDVRP ........ Pickup and Delivery Vehicle Routing Problem

SA ............. Simulated Annealing

TS ............. Tabu Search

TSP ............ Traveling Salesperson Problem

VNS ........... Variable Neighborhood Search

VRPB .......... VRP with Backhauls

VRPCB ........ Vehicle Routing Problem with Clustered Backhauls

VRPDDP ...... Vehicle Routing Problem with Divisible Delivery and Pickup

VRPMB ....... Vehicle Routing Problem with Mixed linehauls and Backhauls

VRPPD ........ VRP with Pickup and Delivery

VRPSDP ....... Vehicle Routing Problem with Simultaneous Delivery and Pickup

VRPTW ....... VRP with Time Windows

VRPWLP ...... Vehicle Routing with Time Windows and Loading Problem

VRP ........... Vehicle Routing Problem

# List of Symbols

$a_i$ .............. total area of all items of customer $i$

$a_q$ .............. area of item $q$

$\alpha_1$ .............. weight of pheromone information

$\alpha_2$ .............. weight of saving values

$\alpha_3$ .............. weight of virtual rectangle

$\alpha_4$ .............. weight of virtual cuboid

$A(\overline{P})$ ........... denotes the set of edges $A(\overline{P}) = \{(p_1, p_2), (p_2, p_3), \dots (p_{|\overline{P}|-1}, p_{|\overline{P}|})\}$ belonging to path $\overline{P}$.

$a^{util}$ ............ average area utilization (2L-CVRP)

$\hat{b}_{q\overline{q}}$ ............. $\hat{b}_{q\overline{q}} = \begin{cases} 1 & \text{if item } q \text{ is positioned behind item } \overline{q} \\ 0 & \text{otherwise} \end{cases}$

$bot$ ............. $bot = \begin{cases} 1 & \text{vehicle bottom is added to touching area of an item} \\ 0 & \text{vehicle bottom is \textbf{not} added to touching area of an item} \end{cases}$

$d^{util}$ ............ average capacity utilization

$cb_d$ ............. chipboards for doors (MP-VRP)

$cb_h$ ............. heavy-use chipboards (MP-VRP)

$cb_l$ ............. long chipboards (MP-VRP)

$cb_s$ ............. short chipboards (MP-VRP)

$c_e$ .............. cost of traveling on edge $e \in \mathcal{E}$

$D$ .............. weight capacity of a vehicle

$d_i$ .............. total weight of all items of customer $i$

$\delta(\mathcal{S})$ ............ set of edges with one endpoint in $\mathcal{S} \subset \mathcal{V}$ and the other in $\mathcal{V} \backslash \mathcal{S}$

$\mathcal{E}$ ............... set of edges available in Graph $\mathcal{G}$

$\eta_1$ .............. weight of the capacity penalty

$\eta_2$ .............. weight of capacity/area utilization

$\mathcal{E}(\mathcal{S})$ ........... set of edges with both endpoints in $\mathcal{S}$

$\hat{f}_{q\overline{q}}$ ............. $\hat{f}_{q\overline{q}} = \begin{cases} 1 & \text{if item } q \text{ is position in front of item } \overline{q} \\ 0 & \text{otherwise} \end{cases}$

$F$ .............. number of elitists

Ⓕ$_q$ ............. Ⓕ$_q = \begin{cases} 1 & \text{if item } q \text{ is fragile} \\ 0 & \text{otherwise} \end{cases}$

$\mathcal{G}$ ............... $\mathcal{G} = (\mathcal{V}_0, \mathcal{E})$ complete, undirected graph

$g_{ij}$ .............. heuristic loading information for customers $i$ and $j$ (MP-VRP)

$\gamma_{ij}$ ............. amount of bulk material needed when pairing customer $i$ and $j$ (MP-VRP)

gap A-B ........ gap between two objective values in percent derived by $\frac{A-B}{B} \cdot 100$

$H$ .............. height of vehicle

$\overline{h_i}$ .............. minimum height of loading one customer alone (MP-VRP)

$\overline{h_{ij}}$ ............. minimum height of loading two customers forming one pair (MP-VRP)

$h^t$ .............. height of one single item of type $t$ (MP-CVRP)

$h_i^t$ .............. total height of all items of type $t$ ordered by customer $i$ plus the height of the pallet (MP-VRP)

$\mathcal{I}$ ............... set of items that contains all items of one problem instance

$\iota$ ............... pheromone update constant

$\mathcal{I}(\overline{P})$ ........... set of items that belong to all customers of one route

$\mathcal{I}_i$ .............. set of items that belong to customer $i$

$\mathcal{I}_i^l$ .............. set of long items that belong to customer $i$ (MP-VRP)

$\mathcal{I}_i^s$ .............. set of short items that belong to customer $i$ (MP-VRP)

$\tilde{K}$ .............. maximum number of selections per generation in a GA

$K$ .............. given number of vehicles to all serve customers

$k(\mathcal{S})$ ............ represents a lower bound on the minimum number of vehicles needed to serve a customer subset $\mathcal{S}$

$\kappa$ ............... number of clustering steps performed (2L-CVRP)

$\hat{l}_{q\overline{q}}$ .............. $\hat{l}_{q\overline{q}} = \begin{cases} 1 & \text{if item } q \text{ is positioned left of item } \overline{q} \\ 0 & \text{otherwise} \end{cases}$

$L$ ............... length of vehicle

$M$ .............. total number of items ordered by all customers

$m_i$ ............. number of items ordered by customer $i$

$m_i^t$ ............. number of items of type $t$ ordered by customer $i$ (MP-VRP)

$n$ ............... number of customers (without depot)

$\overline{NV}$ ............ average number of vehicles need for an instance over several trials

$o_q$ .............. $o_q = \begin{cases} 1 & \text{if item } q \text{ is oriented (not rotated)} \\ 0 & \text{if item } q \text{ is rotated with } w_q = l_q, l_q = w_q \end{cases}$

$O$ .............. initial ordering of items passed to loading heuristics

$\Omega_\Pi$ ............. set of $\Pi$ best customer combinations for merging two routes in the savings algorithm

$\Pi$ .............. size of the neighborhood $\Omega$

$\mathcal{P}$ .............. set of piles available on a vehicle

$P$ .............. number of ants

$\overline{P}$ .............. denotes a path $\overline{P} = (p_1, p_2, \dots, p_{|\overline{P}|})$ representing the ordered sequence of customers without depot

$\phi$ ............... maximum number of permutations tried for loading heuristics

$\Psi$ .............. length of loading (2L-CVRP)

$\hat{r}_{q\overline{q}}$ ............. $\hat{r}_{q\overline{q}} = \begin{cases} 1 & \text{if item } q \text{ is positioned right of item } \overline{q} \\ 0 & \text{otherwise} \end{cases}$

$r(q)$ ............ rank of item $q$ in $\overline{P}$. If $r(q) < r(\overline{q})$, item $q$ has be unloaded before item $\overline{q}$.

$\rho$ ............... pheromone trail persistency

$\tilde{r}$ ............... number of vehicles used in current solution

$\mathcal{S}$ ............... subset of $\mathcal{V}$

$s_{ij}$ .............. $s_{ij} = c_{i0} + c_{0j} - c_{ij}$ (savings)

$sec_h$ ............ time needed to find best solution by a deterministic metaheuristic

$\overline{sec}_h$ ............ average time in seconds needed by a non-deterministic metaheuristic to find best solution (among several trials on the same problem instance)

$sec_{tot}$ .......... time needed to terminate by a deterministic metaheuristic

$\overline{sec}_{tot}$ .......... average time in seconds needed by a non-deterministic metaheuristic to terminate (among several trials on the same problem instance)

$stack$ .......... integer value that is used to multiply the touching area of two items placed on top of each other to favor the stacking of items in the touching area heuristic

$n_c$ .............. number of clusters that the master solution is divided in

$\tau_0$ .............. initial pheromone

$T$ .............. maximum number of iterations (generations)

$\hat{u}_{q\overline{q}}$ ............. $\hat{u}_{q\overline{q}} = \begin{cases} 1 & \text{if item } q \text{ is positioned under item } \overline{q} \\ 0 & \text{otherwise} \end{cases}$

$u_q^{(p)}$ ............ $u_q^{(p)} = \begin{cases} 1 & \text{if item } q \text{ is assigned to pile } p \\ 0 & \text{otherwise} \end{cases}$

$\mathcal{V}$ ............... set of all vertices of the graph $\mathcal{G}$ without the depot

$\mathcal{V}_0$ .............. $\mathcal{V}_0 = \mathcal{V} \cup 0$ set of all vertices of the graph $\mathcal{G}$ with vertex 0 as the depot

$vol_i$ ............ total volume of all items of customer $i$

$vol_q$ ............ volume of item q

$W$ .............. width of vehicle

$\tilde{x}$ ............... solution of a (meta)heuristic

$x_q$ .............. integer position of left, bottom (front) corner of item $q$ on x-axis

$x_e$ .............. $x_e = \begin{cases} 1 & \text{if edge } e \text{ is part of the routing solution} \\ 0 & \text{otherwise} \end{cases}$

$y_q$ .............. integer position of left, bottom corner (front) of item $q$ on y-axis

$z$ ............... objective value found by a deterministic metaheuristic

$\overline{z}$ ............... average objective value found by a non-deterministic metaheuristic (among several trials on the same problem instance)

$z_{max}$ ........... worst objective value found by a non-deterministic metaheuristic (among several trials on the same problem instance)

$z_{min}$ ............ average objective value found by a non-deterministic metaheuristic (among several trials on the same problem instance)

$z_q$ .............. integer position of left, bottom corner (front) of item $q$ on z-axis

# 1 Introduction

Goods are needed everywhere and have to be transported therefore from the production facilities to depots or the end user directly. Although in the last decade there was a lot of discussion concerning global warming, greenhouse effect, etc. and the role of carbon dioxide in this context, more than 75 percent of all goods are transported on the European road network. EuroStat (2008) shows an increasing portion of road transportation for the EU 27 countries (ranging from 73.9% in 2000 to 76.7% in 2006). As a consequence railway, inland water navigation and carriage by air play an inferior role.
Due the significant relevance in day by day operations the so called Vehicle Routing Problem (VRP) has been of interest in academic research for the last five decades. Toth and Vigo (2002*b*, pp. 2–4) define typical characteristics for customers, shippers and objective functions: Typical characteristics of customers are:

- *coordinates* of the road graph where customers are located
- amount of goods that should be either delivered or collected at the customer (*demand*)
- periods of the day where the customer can be served (*time windows*)
- times required to deliver or collect the goods (*(un-)loading times*)
- subset of available vehicles that can be used to serve a certain customer (Sometimes in inner city regions too big trucks cannot pass narrow passages.)

On the other side dispatchers face given constraints for their operation plans:

- locations of the home *depot(s)* (possibility of ending a service at a depot different from the home depot)
- *capacity* of the vehicles (homogenous or heterogenous fleet)
- possible subdivision of the vehicle into compartments; each characterized by its own capacity and type of goods that can be loaded
- devices available to load and unload required goods
- subset of arcs that can be traversed by the vehicles
- *costs* associated

and different objectives can be considered:

- minimization of the global transportation cost, which in the simplest case is equal to the total travel distance of all vehicles. Apart from that, fixed costs concerning vehicles and drivers can be added.
- minimizations of the total number of vehicles used to serve all customers
- balancing of the routes concerning travel times and vehicle loads
- minimization of penalties associated with partial service of customers

Figure 1.1 gives a small overview of the basic problems of the VRP, which are explained in the following paragraph:

1. Capacitated Vehicle Routing Problem (CVRP)
   Customers order a certain demand (usually expressed in weight or volume units) and are served by a single depot. The number of vehicles is not fixed, but each vehicle has a given capacity $C$ that must not be exceeded.

2. VRP with Time Windows (VRPTW)
   This problem extends the CVRP. Apart from the capacity constraint each customer specifies a time interval, called *time window*. Additionally the travel time between two customers $i$ and $j$ and the service time for customer and the time when a vehicle leaves a depot are known in advance. The service for each customer must start within the given time window, moreover when vehicles arrive before the beginning of a time window they have to wait until the beginning of the time window to start their service.

3. VRP with Backhauls (VRPB)
   This extension of the CVRP deals with two sets of customers:

   - Delivery (Linehaul) customers are customers who require delivery of products from the depot.
   - Pickup (Backhaul) customers are customers who have products to be picked up by the driver of the vehicle to be returned to the depot.

   One vehicle can serve customers from the delivery and pickup sets as long as the capacity constraint is met. This problem class can be split into four subclasses (see Parragh, Doerner, and Hartl 2008*a*):

   a) In the Vehicle Routing Problem with Clustered Backhauls (VRPCB) customers are either delivery or pickup customers. All delivery customers have to be served before any pickup customer can be visited.

b) In the Vehicle Routing Problem with Mixed linehauls and Backhauls (VRPMB) again customers are either delivery or pickup customers but in contrast to the VRPCB the visiting sequences can be mixed. Therefore not all delivery customers have to be served before a pickup customer can be visited.

c) In the Vehicle Routing Problem with Divisible Delivery and Pickup (VRPDDP) each customer requires a pickup and a delivery. In addition two visits one for delivery and one for pickup are possible.

d) In the Vehicle Routing Problem with Simultaneous Delivery and Pickup (VRPSDP) each customer is associated with a linehaul and backhaul quantity. In contrast to the VRPDDP each customer can only be served once.

4. VRP with Pickup and Delivery (VRPPD)
In contrast to the VRPB goods are not transported from the depot to the delivery customers and from the pickup customers back to the depot, but between two customer $i$ and $j$ within one route. Therefore each customer specifies his delivery quantity and pickup quantity. It is assumed that at each customer location the delivery is performed before the pickup, which is of practical relevance. The load of each vehicle must not exceed the loading capacity in any stage of its route.
This problem class can divided into to two subclasses (see Parragh, Doerner, and Hartl 2008*b*):

a) Pickup and delivery locations are *unpaired*: In this class only one homogenous good is considered. Thus each unit picked up at any customer can be used to satisfy the demand of any subsequent customer. This problem is denoted as the Pickup and Delivery Vehicle Routing Problem (PDVRP).

b) Pickup and delivery locations are *paired*: In this subclass two problems can be distinguished: the (classical) Pickup and Delivery Problem (PDP) and the Dial-A-Ride Problem (DARP). In contrast to the unpaired case these two problems require that goods are transported from the required pickup location to the required delivery location. In the PDP goods are transported, while the DARP handles passenger transportation.

Figure 1.1 additionally shows an augmentation of the basic problems by the Loading - Vehicle Routing Problem (L-VRP). The L-VRP extends CVRP with loading constraints that are more complex than a simple weight or volume criterion. It consists of three different subgroups, the Multi Pile - Vehicle Routing Problem (MP-VRP), the CVRP with Two-Dimensional Loading Constraints (2L-CVRP) and the CVRP with Three-Dimensional

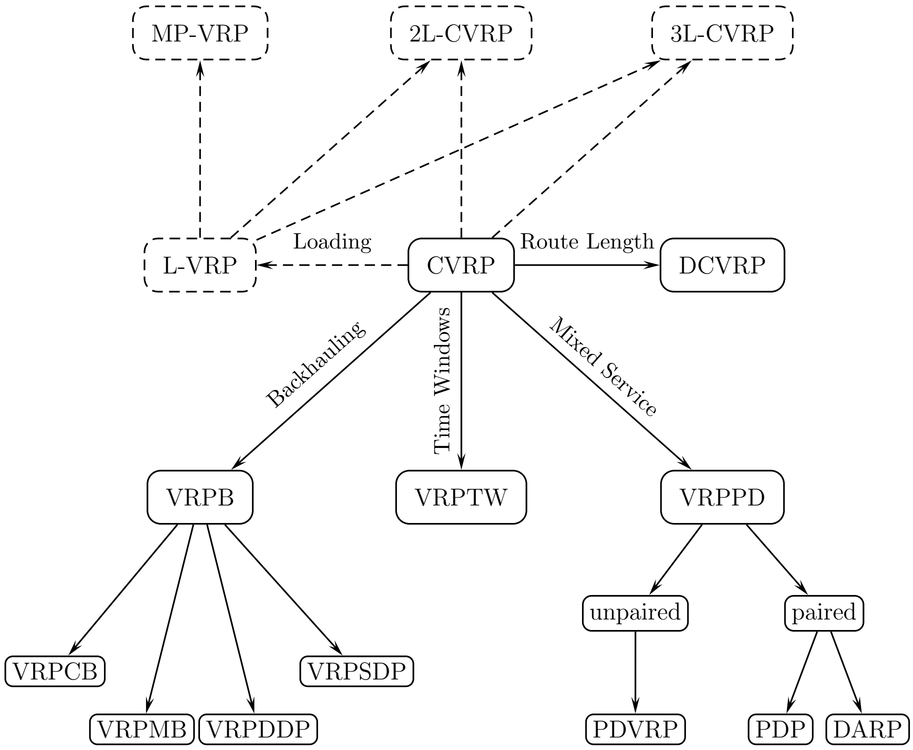



Figure 1.1: The Basic Problems of the VRP and Their Interconnections (see Toth and Vigo (2002*b*, p. 5), Parragh, Doerner, and Hartl (2008*a*) and Parragh, Doerner, and Hartl (2008*b*))

Loading Constraints (3L-CVRP), which are going to be described and discussed extensively in the next chapters.

Please note that the problem classes VRPB and VRPPD and all subclasses can or have been extend to problem variants with time windows and with the above defined loading constraints.

# 2 Problem Definition

## 2.1 MP-VRP

The Multi-Pile Vehicle Routing Problem (MP-VRP) has been introduced by Doerner et al. (2007). This problem is an NP-hard problem, since it generalizes the CVRP. The CVRP and MP-VRP are indeed equivalent if it is only allowed to load items into one pile.
The loading component of the MP-VRP is inspired by an Austrian wood retailer and is characterized by the following type of items:

- *short chipboards* ($cb_s$): are used mainly in furniture production
- *chipboards for doors* ($cb_d$): used for constructing doors and similar products
- *heavy-use chipboards* ($cb_h$): can be used at building sites as supporting material
- *long chipboards* ($cb_l$): are used at building sites as construction material or they are cut to produce short chipboards

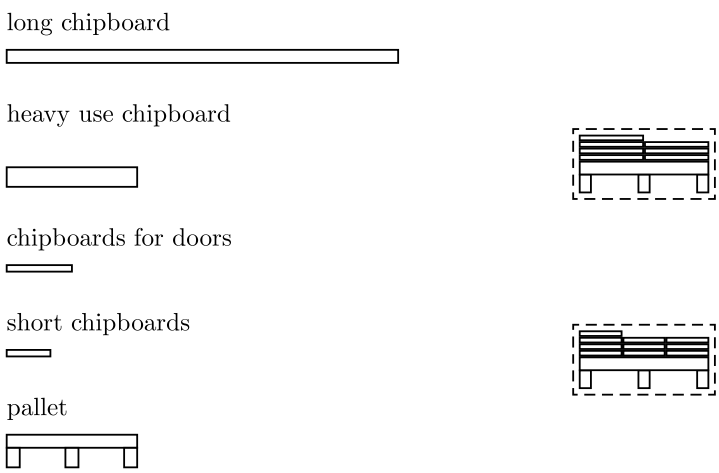
Figure 2.1 shows the dimensional relations between the different chipboards and a pallet. Figure 2.2 shows the grouping of chipboards and pallets of customer $i$ to items according to Equations (2.1) - (2.4): $m_i^t$ stands for the number of items of type $t$ ordered by customer $i$ and $h^t$ for the height of a single item of type $t$ and the pallet respectively. $h_i^t$ represents the total height of each type of chipboards ordered by customer $i$ plus the height of the pallet.

$$h_i^{cb_s} = \left\lceil \frac{m_i^{cb_s}}{3} \right\rceil h^{cb_s} + h^{pal} \tag{2.1}$$

$$h_i^{cb_d} = \left\lceil \frac{m_i^{cb_d}}{2} \right\rceil h^{cb_d} + h^{pal} \tag{2.2}$$

$$h_i^{cb_h} = m_i^{cb_h} h^{cb_h} + h^{pal} \tag{2.3}$$

$$h_i^{cb_l} = m_i^{cb_l} h^{cb_l} + h^{pal} \tag{2.4}$$




(a) Dimensions of Different Chip Board Types and a Pallet

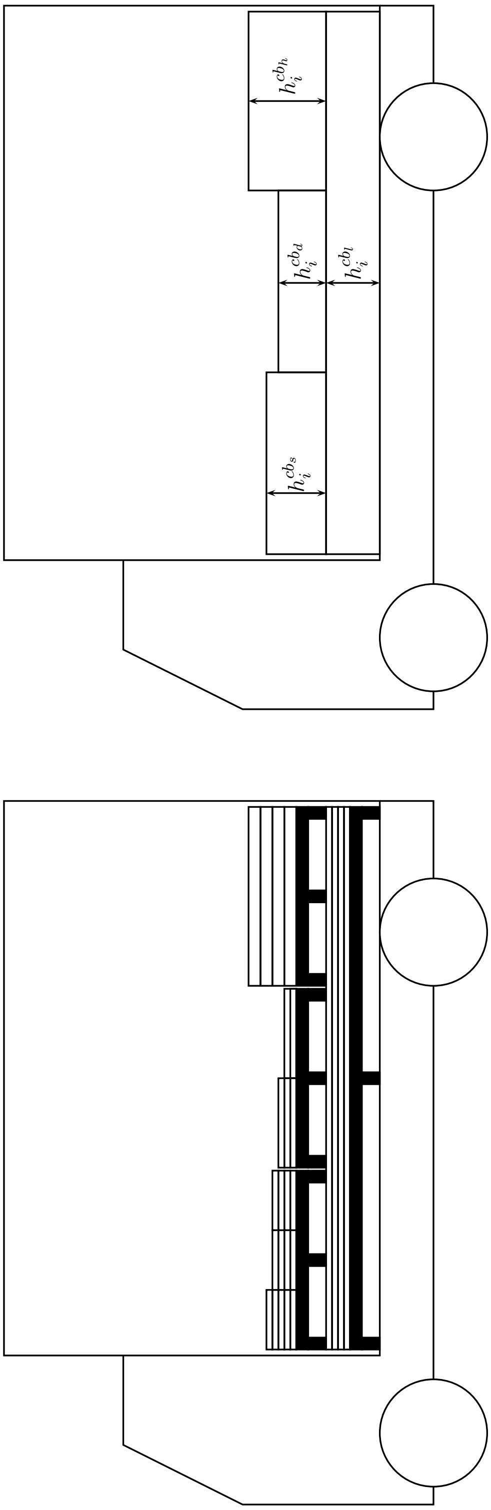
(b) Grouping of chipboards on pallets

Figure 2.1: Relation of Chipboards and Pallets

The loading of one vehicle has to fulfill the following requirements:

- The items must not overlap and the total height of the packing must not exceed the vehicle height.
- Chipboards of one type and one customer have to lie on one pallet.
- All items of one customer have to be transported with the same vehicle (no splitting deliveries).
- When customer $j$ is visited after customer $i$ all items of customer $i$ must be accessible from the unloading side of the vehicle without removing items of customer $j$. The unloading side is located at the right side of the vehicle. Unloading from the rear side of the vehicle is not possible.

Figure 2.3 gives an illustrative example of the MP-VRP and Figure 2.4 the corresponding loading. Note that whenever long items are not fully supported by the short items packed below, bulk material has to be used to ensure loading stability. The construction of feasible loadings will be explained in detail in Section 3.3.1.




(a) Lateral View of Pallets and Chipboards

(b) Lateral View of Items (Chipboards and Pallets Grouped to Items)

Figure 2.2: Relation of Chipboards and Items

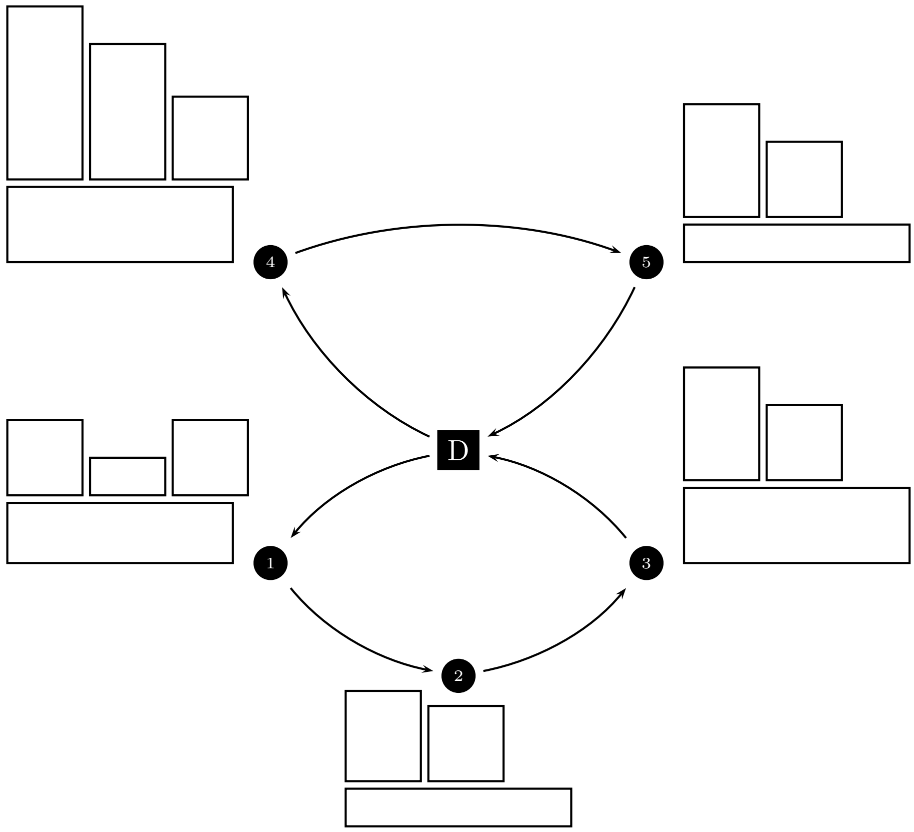
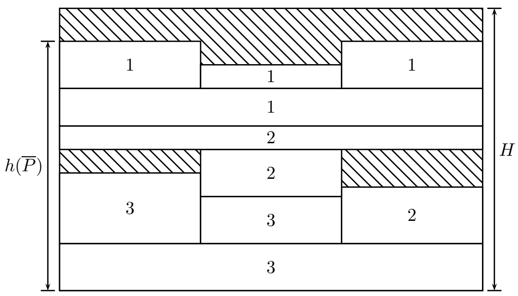
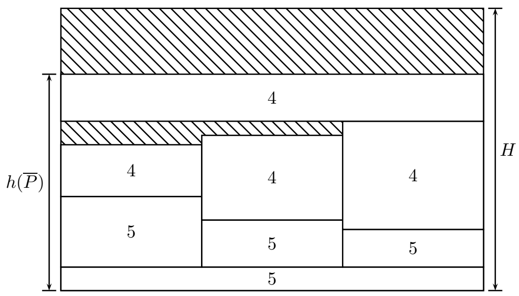



Figure 2.3: An Example of the MP-VRP.




(a) Feasible Loading for Route 1 - 2 - 3




(b) Feasible Loading for Route 4 - 5

Figure 2.4: Feasible Loadings for the Example from Figure 2.3

In order to clarify the problem description a LP model for the MP-VRP is presented: Let $\mathcal{G} = (\mathcal{V}_0, \mathcal{E})$ be a complete, undirected graph with costs of $c_{ij}$ on each edge $(i, j) \in \mathcal{E}$. Equivalently one can use costs of $c_e$ on each edge $e \in \mathcal{E}$ due to the fact that the MP-VRP is symmetric in distances and also symmetric in loading (Looking at Figure 2.4 is easy to see that if there exists a feasible loading for route $1-2-3$ then there is always also a feasible loading for the inverse route $3-2-1$). The vertex set $\mathcal{V}_0 = \mathcal{V} \cup \{0\} = \{v_0, v_1, \dots, v_n\}$ denotes the vertices of all customers and vertex 0 corresponds to the depot. Given a customer subset $\mathcal{S}$ the set of edges with one endpoint in $\mathcal{S}$ and the other in $\mathcal{V} \backslash \mathcal{S}$ is denoted by $\delta(\mathcal{S})$ and the set of edges with both endpoints in $\mathcal{S}$ is denoted by $\mathcal{E}(\mathcal{S})$. Furthermore $\overline{P} = (p_1, p_2, \dots, p_{|\overline{P}|})$ denotes a path representing the ordered sequence of customers without depot and $A(\overline{P}) = \{(p_1, p_2), (p_2, p_3), \dots (p_{|\overline{P}|-1}, p_{|\overline{P}|})\}$ denotes the set of edges belonging to path $\overline{P}$. Finally $k(\mathcal{S})$ represents a lower bound on the minimum number of vehicles needed to serve a customer subset $\mathcal{S}$. $x_e$ specifies if an edge $e \in \mathcal{E}$ is used or not.

$$\min \sum_{e \in \mathcal{E}} x_e c_e \tag{2.5}$$

s. t.

$$\sum_{e \in \delta(i)} x_e = 2 \qquad \forall i \in \mathcal{V} \tag{2.6}$$

$$\sum_{e \in \mathcal{E}(\mathcal{S})} x_e \leq |\mathcal{S}| - k(\mathcal{S}) \qquad \forall \mathcal{S} \subset \mathcal{V}, |\mathcal{S}| \geq 2 \tag{2.7}$$

$$\sum_{e \in A(\overline{P})} x_e \leq |A(\overline{P})| - 1 \qquad \forall \overline{P} \subset \mathcal{V} : \overline{P} \text{ is MP-1-VLP infeasible} \tag{2.8}$$

$$x_e \in \{0, 1\} \qquad \forall e \in \mathcal{E} \backslash \delta(0) \tag{2.9}$$

$$x_e \in \{0, 1, 2\} \qquad \forall e \in \delta(0) \tag{2.10}$$

The objective function is the minimization of the total travel cost according (2.5). The so-called degree constraints (2.6) ensure that each customer is visited exactly once. The generalized subtour elimination constraints (2.7) impose the connectivity of the routes. The infeasible path constraints (2.8) guarantee that only feasible routes are accepted in a solution. Constraint (2.9) imposes that all edges connecting two customers are binary (equal to one if edge $e$ is used and zero otherwise). Edges with one endpoint at the depot are not binary. According to (2.10) they can take values 0, 1 and 2 to allow single customer routes, which are routes that go from the depot to a customer and again back to the depot, e. g. $0 - i - 0$.

The MP-1-VLP in (2.8) refers to the Multi-Pile One Vehicle Loading Problem, which is explained in the following: The item set ordered by each customer is given by $\mathcal{I}_i = \mathcal{I}_i^l \cup \mathcal{I}_i^s$, which is derived from long and short items. $\mathcal{I}(\overline{P}) = \bigcup_{i=1}^{|\overline{P}|} \mathcal{I}(p_i)$ is the set of items corresponding to the beforehand defined path $\overline{P}$ of customers, where $\mathcal{I}(\overline{P}) = \mathcal{I}^l(\overline{P}) \cup \mathcal{I}^s(\overline{P})$ contains all items of path $\overline{P}$ derived from long ($\mathcal{I}^l(\overline{P})$) and short ($\mathcal{I}^s(\overline{P})$) chipboards. Furthermore $\mathcal{P}$ is the set of available piles on the vehicle (here $|\mathcal{P}| = 3$) and $H$ is the loading height of the vehicle. $y_q$ represents the vertical position of item $q$ above the loading floor of a vehicle and $h_q$ gives the height of item $q$. Finally $r(q)$ represents the rank of item $q$ in the path $\overline{P}$ meaning that all items belonging to customer $p_i$ get the same rank as customer $p_i$ has in path $\overline{P}$. $u_q^{(p)}$ is equal to 1, if item $q$ is assigned to pile $p$ and $\hat{u}_{q\overline{q}}$ is equal to 1, if item $q$ lies under item $\overline{q}$

$$\sum_{p \in \mathcal{P}} u_q^{(p)} = 1 \qquad \forall q \in \mathcal{I}^s(\overline{P}) \tag{2.11}$$

$$\sum_{p \in \mathcal{P}} u_q^{(p)} = |\mathcal{P}| \qquad \forall q \in \mathcal{I}^l(\overline{P}) \tag{2.12}$$

$$y_q + h_q \leq y_{\overline{q}} + H(3 - u_q^{(p)} - u_{\overline{q}}^{(p)} - \hat{u}_{q\overline{q}}) \qquad \forall q, \overline{q} \in \mathcal{I}(\overline{P}) \wedge q \neq \overline{q}, p \in \mathcal{P} \tag{2.13}$$

$$\hat{u}_{q\overline{q}} + \hat{u}_{\overline{q}q} \geq u_q^{(p)} + u_{\overline{q}}^{(p)} - 1 \qquad \forall q, \overline{q} \in \mathcal{I}(\overline{P}) \wedge q \neq \overline{q}, p \in \mathcal{P} \tag{2.14}$$

$$y_q \geq y_{\overline{q}} + h_{\overline{q}} - H(2 - u_q^{(p)} - u_{\overline{q}}^{(p)}) \qquad \forall q, \overline{q} \in \mathcal{I}(\overline{P}), r(q) < r(\overline{q}), p \in \mathcal{P} \tag{2.15}$$

$$y_q + h_q \leq H \qquad \forall q \in \mathcal{I}(\overline{P}) \tag{2.16}$$

$$u_q^{(p)} \in \{0, 1\} \qquad \forall q \in \mathcal{I}(\overline{P}), p \in \mathcal{P} \tag{2.17}$$

$$\hat{u}_{q\overline{q}} \in \{0, 1\} \qquad \forall q, \overline{q} \in \mathcal{I}(\overline{P}) \tag{2.18}$$

$$y_q \geq 0 \qquad \forall q \in \mathcal{I}(\overline{P}) \tag{2.19}$$

The objective of the loading subproblem MP-1-VLP is to find a feasible solution to constraints (2.11) - (2.19). Constraints (2.11) and (2.12) ensure that each short item occupies exactly one pile and each long item occupies all piles of the truck respectively. Constraint (2.13) guarantees that items of customers placed in the same pile do not overlap. If items $q$ and $\overline{q}$ are located in the same pile and $q$ lies under $\overline{q}$, the vertical position $y_q$ plus the height $h_q$ of item $q$ have to be less or equal to the vertical position $y_{\overline{q}}$ of item $\overline{q}$. (2.14) links $u_q^{(p)}$ and $\hat{u}_{q\overline{q}}$. If items $q$ and $\overline{q}$ are placed in the same pile, either $q$ has to be positioned above $\overline{q}$ or vice versa. The sequence constraint (2.15) is needed for the LIFO unloading policy, meaning that if customer $i$ is delivered item $q$ before customer $j$ is delivered item $\overline{q}$, item $q$ has to be placed above item $\overline{q}$ in the same pile or in different piles. According to constraint (2.16) the total height of the packing has to be smaller or equal to the loading height of the vehicle. Constraints (2.17) and (2.18) ensure that $u_q^{(p)}$ and $\hat{u}_{q\overline{q}}$ are binary. The last constraint (2.19) guarantees placements of items on or above the loading floor.

## 2.2 2L-CVRP

The Vehicle Routing Problem with Two-Dimensional Loading Constraints (2L-VRP) has been introduced by Gendreau et al. (2008) and has been solved by means of Tabu Search (TS). An exact approach by branch and cut for this problem has been addressed by Iori, Salazar-González, and Vigo (2006), whereas Zachariadis, Tarantilis, and Kiranoudis (2007) have solved the problem by Guided Tabu Search (GTS) and Fuellerer et al. (2007) by Ant Colony Optimization (ACO), which will be presented in detail in Section 3.4.

In the 2L-CVRP a complete, undirected graph $\mathcal{G} = (\mathcal{V}_0, \mathcal{E})$ with edge cost $c_e$ and $e = (i, j)$ between two customers $i$ and $j$ is given. $\mathcal{V}_0 = \mathcal{V} \cup 0 = \{v_0, v_1, \ldots, v_n\}$ is the set of $n + 1$ vertices corresponding to the depot (0) and customers. $\mathcal{I} = \bigcup_{q \in \{1,\ldots,M\}} (w_q, l_q)$ is the set of all items ordered by all customers. Each item is characterized by a certain width $w_q$ and a certain length $l_q$. The items ordered by one customer are denoted by the set $\mathcal{I}_i$. These items have a total weight of $d_i$ and a total area of $a_i = \sum_{q \in \mathcal{I}_i} w_q \cdot l_q$. Furthermore there are $K$ identical vehicles available at the depot with a loading surface of $W \times L$ and a loading capacity of $D$. The opening of the vehicle is placed at the $W$ edge and is $W$ units wide. The objective of the 2L-CVRP is the minimization of total routing cost while meeting the following constraints:

- All customers have to be visited exactly once (no splitting deliveries).
- The vehicle capacity of $D$ must not be exceeded.
- Orthogonal packing is required. This means that all items are packed parallel to the $W$ and $L$ edges of the vehicle, which has a practical reason: It is much safer and easier to unload goods with forklifts, when the fork and the item are parallel to each other.
- Additionally four different loadings configurations are investigated:
  - 2|RO|L: two-dimensional rear oriented loading
    Items must not be rotated by 90° meaning that the $w_q$ edge of an item is always parallel to the $W$ edge of the vehicle. Rear loading is required, so that at each customer site all items of customer $i$ can be unloaded without moving items of subsequent customers on the route (LIFO loading policy).
  - 2|UO|L: Two-Dimensional Unrestricted Oriented Loading
    Items must not be rotated but no LIFO loading policy is required.
  - 2|RN|L: Two-Dimensional Rear Non-Oriented Loading

Items may be rotated by 90° and rear loading is necessary.

- 2|UN|L: Two-Dimensional Unrestricted Non-Oriented Loading
  Neither fixed orientation, nor LIFO loading policy is required.

Figures 2.5 and 2.6 show the routing and loading solution for one of the smallest instances for this problem with 15 customers. It is interesting to observe, that with the loosest loading requirement 2|UN|L it is not only possible to generate a solution with lowest total routing cost but additionally save one vehicle for delivery. Furthermore looking at Figure 2.6b one can easily see that vehicle 3 with delivery route $6-11-19-10$ is not rear loaded: Assuming that the opening of the vehicle is at the bottom of the picture, the driver cannot unload the items of customer 6 before unloading the wide item of customer 11. One could argue, that traveling the delivery route in the inverse direction $10-19-11-6$ (which would lead to identical routing costs due to Euclidian distances) could be feasible. However this is false because after unloading customers 10 and 19 the items of customer 11 cannot be unloaded without moving the item of customer 6. It is quite obvious that when a route is rear loading feasible in one direction it is also feasible in the inverse direction. A more detailed view on the managerial implications of different loading configurations is given in Section 4.2.

For modeling the problem it is possible to reuse the LP model of the MP-VRP. In fact objective and constraints (2.20) - (2.25) are identical to the MP-VRP, except for the Two-Dimensional One Vehicle Loading Problem (2D-1-VLP), which is formulated in the following paragraph.

$$\min \sum_{e \in \mathcal{E}} x_e c_e \tag{2.20}$$

s. t.

$$\sum_{e \in \delta(i)} x_e = 2 \qquad \forall i \in \mathcal{V} \tag{2.21}$$

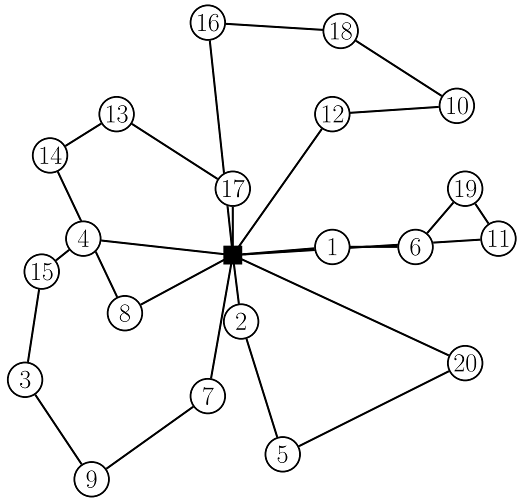
$$\sum_{e \in \mathcal{E}(\mathcal{S})} x_e \leq |\mathcal{S}| - k(\mathcal{S}) \qquad \forall \mathcal{S} \subset \mathcal{V}, |\mathcal{S}| \geq 2 \tag{2.22}$$

$$\sum_{e \in A(\overline{P})} x_e \leq |A(\overline{P})| - 1 \qquad \forall \overline{P} \subset \mathcal{V} : \overline{P} \text{ is 2D-1-VLP infeasible} \tag{2.23}$$

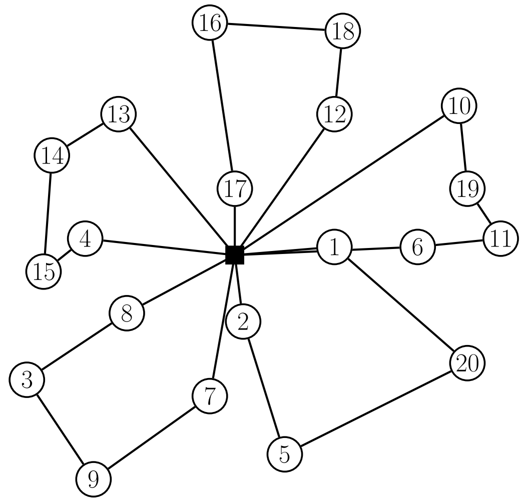
$$x_e \in \{0,1\} \qquad \forall e \in \mathcal{E} \backslash \delta(0) \tag{2.24}$$

$$x_e \in \{0,1,2\} \qquad \forall e \in \delta(0) \tag{2.25}$$

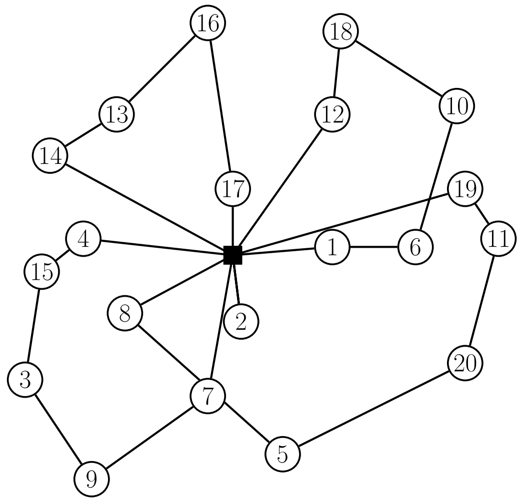
The 2D-1-VLP aims at determining a feasible loading for a given set of items $\mathcal{I}(\overline{P})$ into a vehicle. Chen, Lee, and Shen (1995) introduced an analytical model for the Container



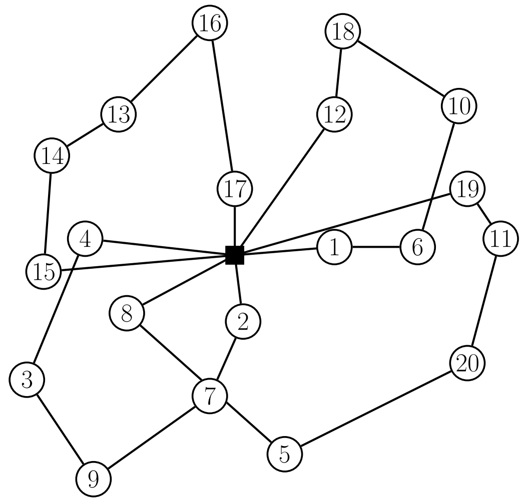
(a) 2|RO|L: $z = 403.93$



(b) 2|UO|L: $z = 387.70$

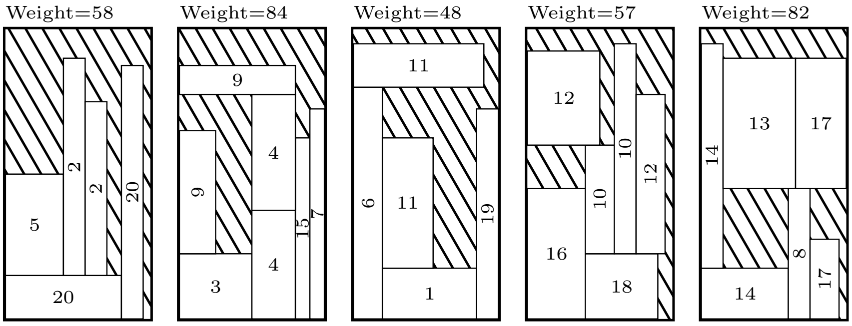


(c) 2|RN|L: $z = 384.93$

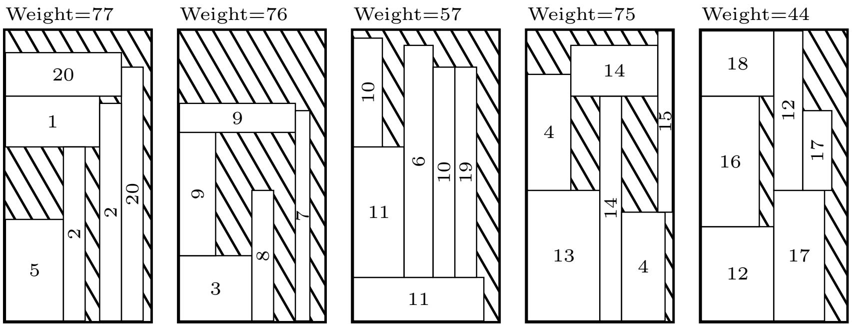


(d) 2|UN|L: $z = 380.35$

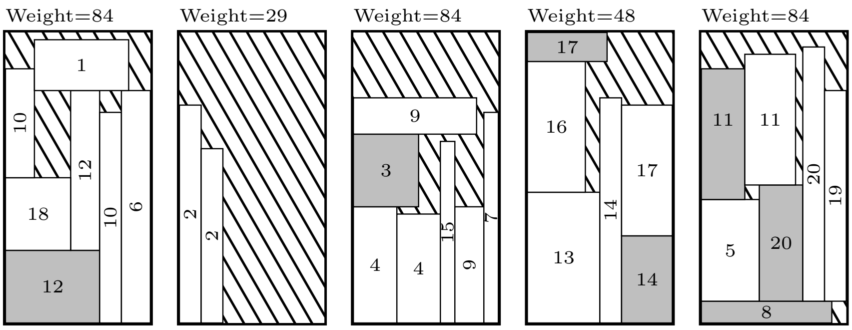
Figure 2.5: Routing Solutions to Problem Instance `2L-CVRP0302`



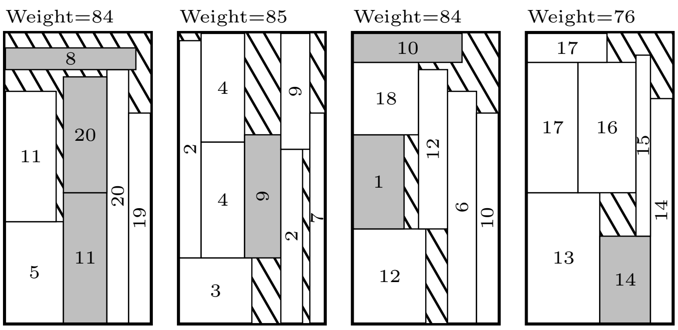

(a) 2|RO|L




(b) 2|UO|L




(c) 2|RN|L




(d) 2|UN|L

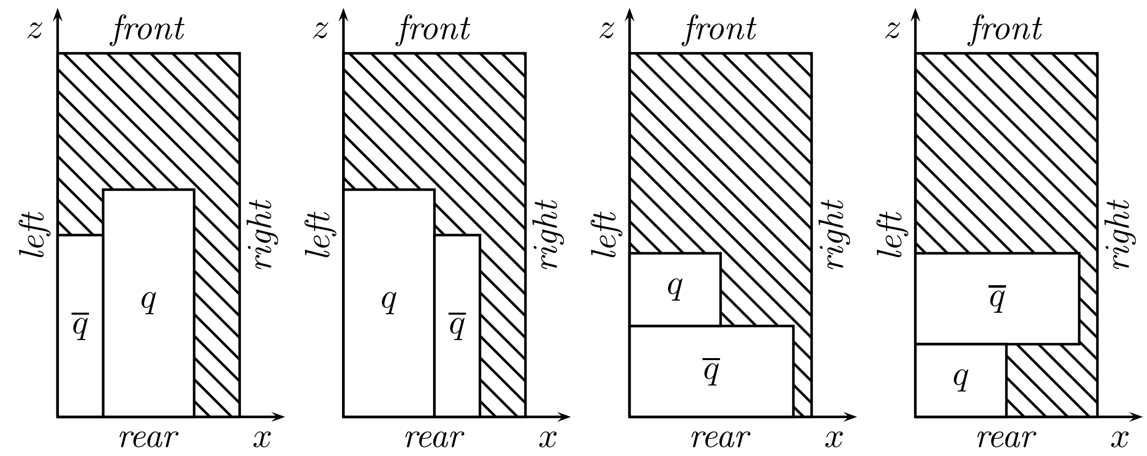
Figure 2.6: Loading Solutions Corresponding to Figure 2.5. The Numbers in the Items Refer to the Customer IDs and Gray Items Have Been Rotated by 90°.

Loading Problem (CLP), which is partly used here. $\mathcal{I}(\overline{P})$ is the set of items to be loaded in one truck with $\mathcal{I}(\overline{P}) = \bigcup_{i=1}^{|\overline{P}|} \mathcal{I}(p_i)$. $x_q$ and $z_q$ denote the x and z coordinate of the left front corner of item $q$. $o_q$ represents the orientation of item $q$ and is equal 1 if item $q$ is rotated and 0 otherwise. The relation variables $\hat{r}_{q\overline{q}}$, $\hat{l}_{q\overline{q}}$, $\hat{b}_{q\overline{q}}$ and $\hat{f}_{q\overline{q}}$ set the relation of two items:

- If $\hat{r}_{q\overline{q}}$ is 1, item $q$ is placed right of item $\overline{q}$ and $x_q \geq x_{\overline{q}} + w_{\overline{q}}$ (see Figure 2.7a).
- If $\hat{l}_{q\overline{q}}$ is 1, item $q$ is placed left of item $\overline{q}$ and $x_q + w_q \leq x_{\overline{q}}$ (see Figure 2.7b).
- If $\hat{f}_{q\overline{q}}$ is 1, item $q$ is placed in front of item $\overline{q}$ and $z_q \geq z_{\overline{q}} + l_{\overline{q}}$ (see Figure 2.7d).
- If $\hat{b}_{q\overline{q}}$ is 1, item $q$ is placed behind item $\overline{q}$ and $z_q \leq z_{\overline{q}} + l_{\overline{q}}$ (see Figure 2.7c).

$$x_q + o_q w_q + (1 - o_q) l_q \leq x_{\overline{q}} + (1 - \hat{l}_{q\overline{q}}) W \quad \forall q, \overline{q} \in \mathcal{I}(\overline{P}) \wedge q < \overline{q} \tag{2.26}$$
$$x_{\overline{q}} + o_{\overline{q}} w_{\overline{q}} + (1 - o_{\overline{q}}) l_{\overline{q}} \leq x_q + (1 - \hat{r}_{q\overline{q}}) W \quad \forall q, \overline{q} \in \mathcal{I}(\overline{P}) \wedge q < \overline{q} \tag{2.27}$$
$$z_q + o_q l_q + (1 - o_q) w_q \leq x_{\overline{q}} + (1 - \hat{f}_{q\overline{q}}) L \quad \forall q, \overline{q} \in \mathcal{I}(\overline{P}) \wedge q < \overline{q} \tag{2.28}$$
$$z_{\overline{q}} + o_{\overline{q}} l_{\overline{q}} + (1 - o_{\overline{q}}) w_{\overline{q}} \leq z_q + (1 - \hat{b}_{q\overline{q}}) L \quad \forall q, \overline{q} \in \mathcal{I}(\overline{P}) \wedge q < \overline{q} \tag{2.29}$$
$$\hat{l}_{q\overline{q}} + \hat{r}_{q\overline{q}} + \hat{f}_{q\overline{q}} + \hat{b}_{q\overline{q}} \geq 1 \quad \forall q, \overline{q} \in \mathcal{I}(\overline{P}) \wedge q < \overline{q} \tag{2.30}$$
$$x_q + o_q w_q + (1 - o_q) l_q \leq W \quad \forall q \in \mathcal{I}(\overline{P}) \tag{2.31}$$
$$z_q + o_q l_q + (1 - o_q) w_q \leq L \quad \forall q \in \mathcal{I}(\overline{P}) \tag{2.32}$$
$$\hat{l}_{q\overline{q}} + \hat{r}_{q\overline{q}} + \hat{b}_{q\overline{q}} \geq 1 \quad \forall q \in \mathcal{I}(\overline{P}), \overline{q} \in \mathcal{I}(\overline{P}), r(q) < r(\overline{q}) \tag{2.33}$$
$$\tag{2.34}$$
$$\hat{r}_{q\overline{q}}, \hat{l}_{q\overline{q}}, \hat{b}_{q\overline{q}}, \hat{f}_{q\overline{q}} \in \{0, 1\} \quad \forall q, \overline{q} \in \mathcal{I}(\overline{P}) \tag{2.35}$$
$$x_q \geq 0 \quad \forall q \in \mathcal{I} \tag{2.36}$$
$$z_q \geq 0 \quad \forall q \in \mathcal{I} \tag{2.37}$$
$$o_q \in \{0, 1\} \quad \forall q \in \mathcal{I}(\overline{P}) \tag{2.38}$$

The constraints (2.26) - (2.30) ensure that no two items can overlap. Constraint (2.30) ensures that at least one of the relation variables is 1, meaning that item $q$ lies either right (left) and/ or front (behind) of item $\overline{q}$. Constraint (2.31) guarantees that the vehicle width $W$ is not exceed, whereas constraint (2.32) allows no placements of items exceeding the vehicle length $L$. The following constraint (2.33) is needed to ensure the LIFO-Loading policy: If two items $q, \overline{q}$ belong to different customers $i, j$ and $q$ has to be unloaded before $\overline{q}$, $q$ has to be placed either left, right or behind $\overline{q}$ - it must not be placed in front of $\overline{q}$. The relation variables (2.35) and the orientation variable (2.38) $o_q$ must be binary. The placement variables $x_q$ and $z_q$ are integer and have to be greater or equal to 0.




(a) $\hat{r}_{q\overline{q}} = 1$ (b) $\hat{l}_{q\overline{q}} = 1$ (c) $\hat{f}_{q\overline{q}} = 1$ (d) $\hat{b}_{q\overline{q}} = 1$

Figure 2.7: Relation Variables

## 2.3 3L-CVRP

The Vehicle Routing Problem with Three-Dimensional Loading Constraints has been introduced by Gendreau et al. (2006) and generalizes the 2L-CVRP by introducing a third dimension in the loading problem. Oliveira and Moura (2008) present a combination of CVRPTW with the Container Loading Problem (CLP), which is denoted as Vehicle Routing with Time Windows and Loading Problem (VRPWLP). The main differences between these two problems are the time window aspect on the one hand and different loading constraints on the other hand. More details are going to follow after the problem description of the 3L-CVRP.

In the 3L-CVRP a complete, undirected graph $\mathcal{G} = (\mathcal{V}_0, \mathcal{E})$ with edge cost $c_e$ and $e = (i, j)$ between two customers $i$ and $j$ is given. $\mathcal{V}_0 = \mathcal{V} \cup 0$ is the set of $n + 1$ vertices corresponding to the depot (0) and customers ($\mathcal{V} = \{v_1, \ldots, v_n\}$). The total set of items ordered by all customers is given by $\mathcal{I} = \bigcup_{q \in \{1,\ldots,M\}} (w_q, h_q, l_q, \textcircled{F}_q)$. Each item is characterized by a certain width $w_q$, height $h_q$ length $l_q$ and fragility status ($\textcircled{F}_q = 1$ for fragile items and $\textcircled{F}_q = 0$ for non-fragile ones). Each customer $i$ orders items $\mathcal{I}_i$ with a total volume $vol_i = \sum_{q \in \mathcal{I}_i} w_q h_q l_q$ and a total weight of $d_i$. Furthermore there are $K$ identical vehicles available at the depot with a loading space of $W \times H \times L$ and a loading capacity of $D$. The opening of the vehicle is placed at the $W \times H$ surface and is supposed to be $W$ units wide and $H$ units high. Figure 2.8 gives an example of a 3L-CVRP instance with eight customers and Figure 2.9 shows the according loading of different customers. The items of customers belonging to the same route have different colors (white, light gray and gray).

Items with dashed border lines are fragile (while solid border lines indicate non-fragile items). The objective of the 3L-CVRP is the minimization of total routing cost while meeting the following constraints:

- All customers have to served exactly once (no splitting deliveries).
- Items must not overlap and must not exceed the vehicle loading space.
- Non-fragile items may only be stacked on non-fragile ones, they must not be stacked *directly* onto fragile ones. In Figure 2.9c all fragile items are either placed on the vehicle floor or on non-fragile items. Figure 2.10 clarifies the meaning of direct contact.
- Each item that is not placed on the vehicle floor has to be supported by at least 75% of its base area. Referring to Figure 2.9a the large item of customer 1 and all other items are either fully supported by the vehicle floor or other items. The long item of customer 1 is only partly supported due to the fact that the item of customer 3 is not as high as the large item of customer 1.
- LIFO Loading: This constraint is equivalent to the MP-VRP and 2L-CVRP: It must be possible to unload all items of customer $i$ (through the opening in the rear of the vehicle) without moving items of customer $j$, whenever $i$ is visited before $j$. The loading in Figures 2.9a - 2.9c are LIFO loading feasible. Figure 2.11 clarifies the meaning of LIFO loading.

For the sake of simplicity and ability to address which set of constraints is active and which not the following notation is introduced: $3|LIFO \in \{R(ear), U(unrestricted)\}|$® $\in \{O(oriented), N(on-oriented)\}|$Ⓕ $\in \{0,1\}|\underline{A} \in [0,100]|L$. Therefore $3|R|N|1|75|L$ stands for: LIFO loading and fragility has to be accounted for, items may be rotated by 90° on the x-z plane and a supporting area of 75% is required.

This set of constraints raises one interesting question: *Are loadings still symmetric?* Alternatively formulated, if route $1-2-3$ is feasible, is route $3-2-1$ also feasible. To answer this question one can consider Figure 2.9a: Assuming that the rear and front of the vehicle have been changed, the first customer to visit is customer 1. It is not possible to get the gray item of customer 1 without moving the on top placed item of customer 2. Therefore it is necessary to rotate the whole cargo by 180° around the z-axis. This allows to unload the gray item first and it is possible to unload the items of the subsequent customers according to the LIFO constraint. However either the supporting area or fragility constraint may be violated, which is indeed true in this case. The long, thin item of customer 1 does not provide enough supporting area for the large item of the same customer when the whole loading space is rotated by 180° along the z-axis. Although for this example there exists a feasible loading also for the inverse route $3-2-1$ meeting all defined constraints,

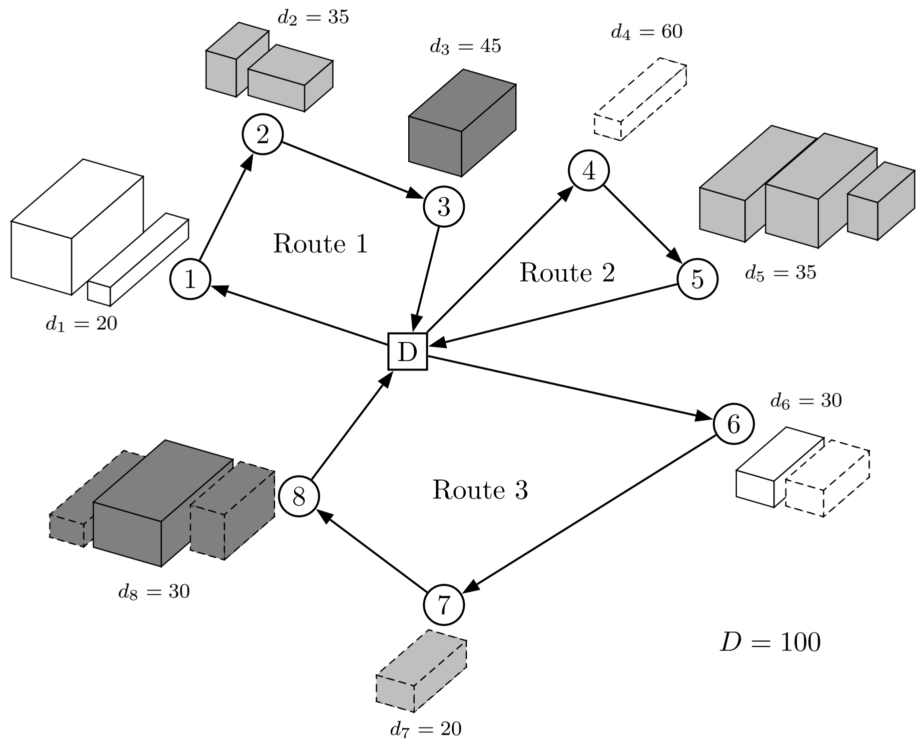
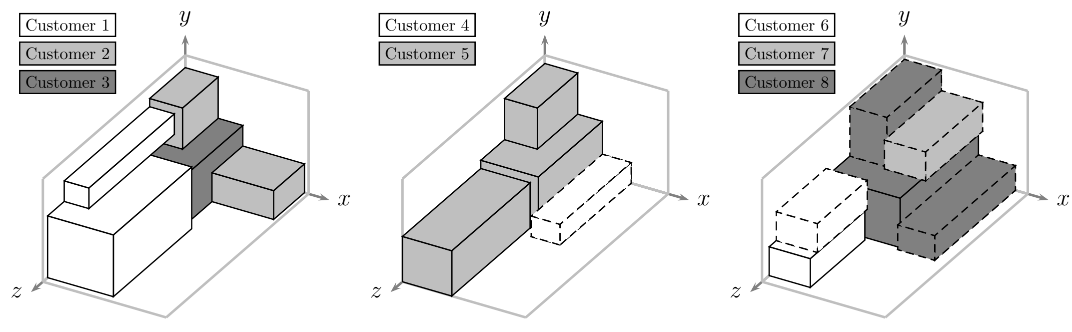
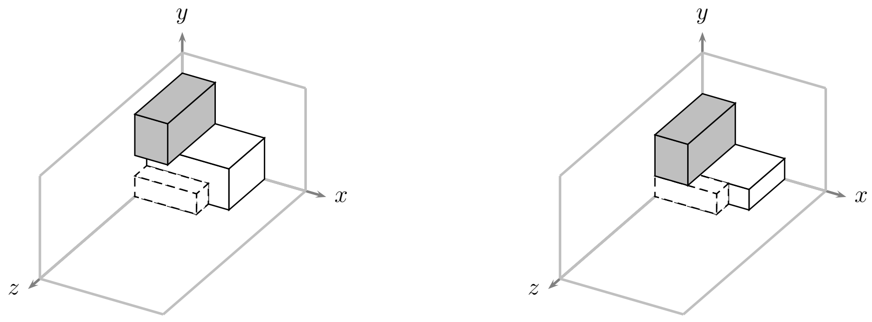



Figure 2.8: Example for the 3L-CVRP with $n = 8$




(a) Loading of Route 1 (b) Loading of Route 2 (c) Loading of Route 3

Figure 2.9: Loading for the Example from Figure 2.8




(a) Allowed Placement of Gray Item due to No Direct Contact between Non-fragile and Fragile Item

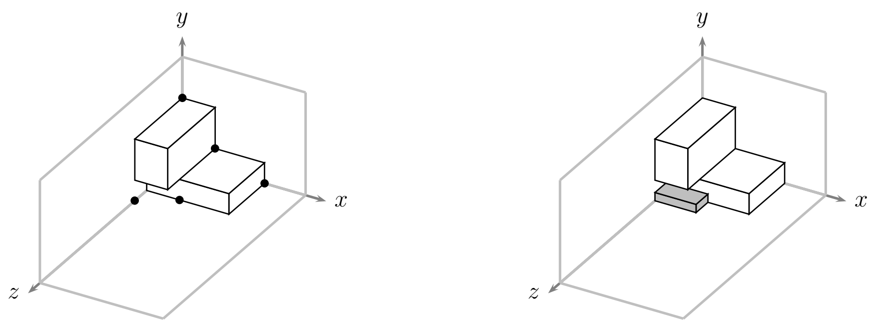
(b) Forbidden Placement of Gray Item due to Direct Contact between Non-fragile and Fragile Item

Figure 2.10: Fragility Constraint

it quite easy to construct examples where no feasible loadings exist for the inverse route. Considering two customers $i$ with an item $w_{i1} = W, h_{i1} = \frac{H}{2}, l_{i1} = L, Ⓕ_q 1 = 0$ and $j$ with an item $w_{j1} = 1, h_{j1} = 1, l_{j1} = 1, Ⓕ_{\overline{q}} 1 = 1$ only the route $i - j$ can be feasible. Due to the fact that the item of customer $i$ is as large as the vehicle floor, items of other customers can only placed below or above the item of customer $i$, which in this case leads to an supporting area and fragility violation (see Figure 2.12).

For the 3L-CVRP no LP formulation like for the MP-VRP or 2L-CVRP will be given. On the one hand the supporting area constraint can only be handled with quadratic constraints and on the other hand the model for the loading subproblem would get too complicated to be of any help to state the problem more precisely.

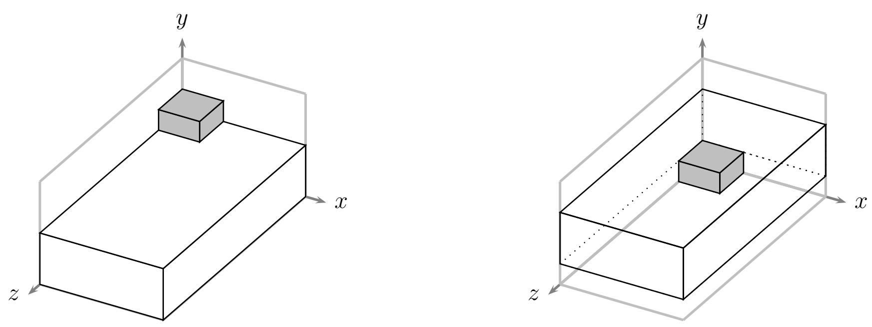
The interested reader is referred to Oliveira and Moura (2008) where a LP model for the VRPWLP is represented. However the constraint of supporting area is not taken into account. The main difference between the 3L-CVRP and the VRPWLP concerns the loading. Both problems postulate LIFO loading. In the CLP each customer usually orders several items of the same box type. Items of one box type have the same dimensions and can be rotated in the same directions. In the VRPWLP a supporting area of 100% is required and for stability reasons each item has to be surrounded by at least three sides by other items or the container walls. For the 3L-CVRP a supporting area of 100% can only be satisfied when splitting deliveries are allowed because otherwise for some instances single customers cannot be loaded onto one vehicle.




(a) Allowed Placing Points for Gray Item

(b) Forbidden Placement of Gray Item

Figure 2.11: LIFO Loading Constraint (Gray Item Has to be Unloaded before White Items)




(a) Supporting Area constraint Met

(b) Supporting Area Constraint Not Met (<50%))

Figure 2.12: Supporting Area Constraint with Two Customers and One Item Per Customer

# 3 Solution Methods

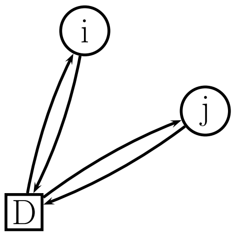
## 3.1 Brief Overview on (Meta)Heuristics and Exact Approaches for the CVRP

### 3.1.1 Classical Construction and Improvement Heuristics

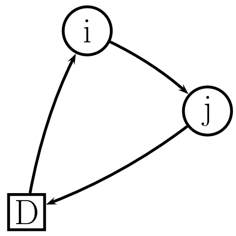
The ideas of construction heuristics are well known and well documented (see Laporte and Semet 2002; Cordeau et al. 2002). This section will only provide a very limited and short overview on the different approaches, the interested reader is referred to the given references.

#### Savings Algorithm

The savings algorithm by Clarke and Wright (1964) is based on the notion of savings given by $s_{ij} = c_{i0} + c_{0j} - c_{ij}$ illustrated by Figure 3.1 and is explained in Algorithm 3.1. The algorithm starts with serving each customer with one vehicle, thus the first solution consists of $n$ routes. By calculating the $s_{ij}$ for each pair of customers, the most profitable, feasible combination is chosen and the two routes $(0, \ldots, i, 0)$ and $(0, j, \ldots, 0)$ are merged. For the solution with $n-1$ vehicles again the $s_{ij}$ are calculated and the most profitable, feasible combination is chosen. The algorithm stops when no two routes can be profitably and feasibly merged.




(a) Visiting Customers $i$ and $j$ Separately




(b) Savings due to Route Merge

Figure 3.1: Savings Algorithm

**Algorithm 3.1** Savings Algorithm

Calculate $s_{ij}$ for all customer pairs $i, j = 1, \dots, n$ and $i \neq j$ and create a non-increasing ordered list with $s_{ij}$
Initialize $k = n$ routes, each serving exactly one customer $i$.
Select first entry of the list $s_{ij}$
**while** End of list with $s_{ij}$ is not reached **do**
    **if** Merge of routes $(0, \dots, i, 0)$ and $(0, j, \dots, 0)$ is feasible **then**
        Merge these two routes.
    **end if**
    Check next $s_{ij}$ entry.
**end while**

#### Sweep Algorithm

The idea of the sweep algorithm by Gillett and Miller (1974) is building clusters by rotating a ray centered at the depot. The concept is illustrated by Figure 3.2 and Algorithm 3.2.

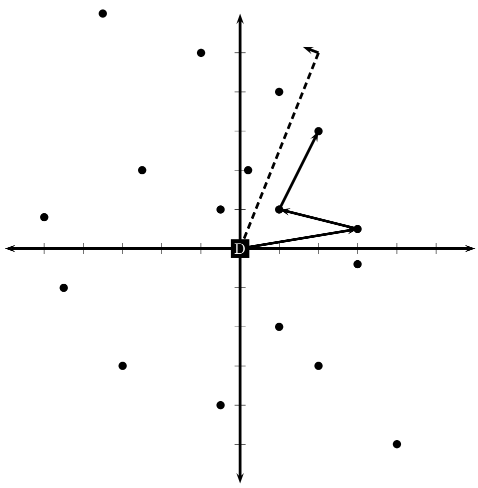



Figure 3.2: Sweep

#### Fisher and Jaikumar Algorithm

This algorithm by Fisher and Jaikumar (1981) forms clusters by solving a Generalized Assignment Problem (GAP) and can be described by Algorithm 3.3

**Algorithm 3.2** Sweep

```
Calculate the angle of all customer vertices to a reference line (e. g. x-axis) and create
  a non-increasing sorted list
Initialize first vehicle with first customer of list.
while End of list is not reached do
    if Assignment of customer i to current route is feasible then
        Assign customer i to current route.
    else
        Assign customer i to new route.
    end if
    Check next customer from the list.
end while
```

**Algorithm 3.3** Fisher and Jaikumar (see Laporte and Semet 2002, p. 117)

1. Chooses seed vertices $j_k \in \mathcal{V}$ to initialize each cluster $k$. ▷ seed selection
2. Compute the cost $d_{ik}$ of allocating each customer $i$ to each cluster $k$ as $d_{ik} = \min\{c_{0i} + c_{ij_k} + c_{j_k0}, c_{0j_k} + c_{j_ki} + c_{i0}\} - (c_{0j_k} + c_{j_k0})$ ▷ Allocation of customers to seeds
3. Solve a GAP with costs $d_{ik}$, customer weights $d_i$ and vehicle capacity $D$ ▷ General Assignment
4. Solve a TSP for each cluster corresponding to the GAP solution. ▷ TSP solution

#### Route-First, Cluster-Second Methods

Beasley (1983) proposed this algorithm, which builds a giant TSP tour disregarding side constraints and decomposes this tour into feasible vehicle routes in the second phase. However according to Laporte and Semet (2002) this approach is not competitive to the other presented construction heuristics.

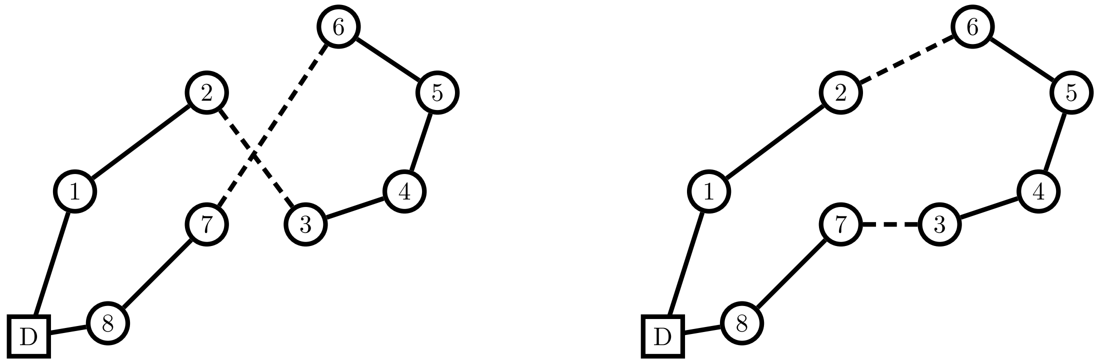
#### Improvement Heuristics

All improvement heuristics known from the TSP problem can be used for the VRP, because each route represents a TSP problem. Apart from this intra-route improvement heuristics, there are also inter-route improvement heuristics.

1. *Intra-route Improvement Heuristics*
    - **$\lambda$-opt** with $\lambda \in [2, \ldots, n]$(Lin 1965)
      Most common are 2-opt and 3-opt. In the 2-opt two edges of one route are removed and the route fragments are reconnected. In the example of Figure 3.3 the edges from customer 2 to 3 and 6 to 7 are removed $(0-1-2 \times 3-4-5-6 \times 7-8-0)$ and the tour is reconnected with customer 2 to 6 and 3 to 7

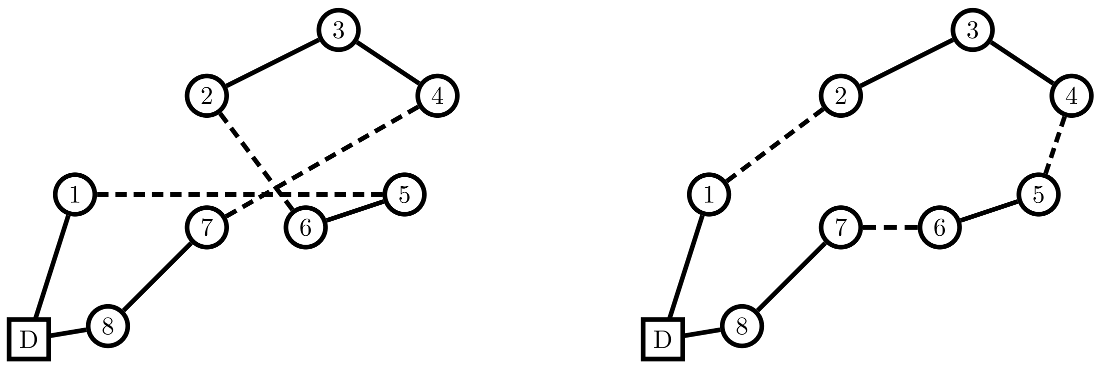
$(0-1-2-6-5-4-3-7-8-0)$. Note that the 2-opt removes all crossings when the triangular inequality holds ($c_{ij} \leq c_{ik} + c_{kj}$).

The 3-opt removes 3 edges and reconnects the route fragments. Figure 3 shows an example. Note that if inverting route parts is allowed, it is also possible to form $(0-1-4-3-2-5-6-7-8-0)$ or $(0-1-6-5-4-3-2-7-8-0)$. However one always has to assess if the additional computational effort is really reflected in solution quality.




(a) Solution before 2-opt (b) Solution after 2-opt

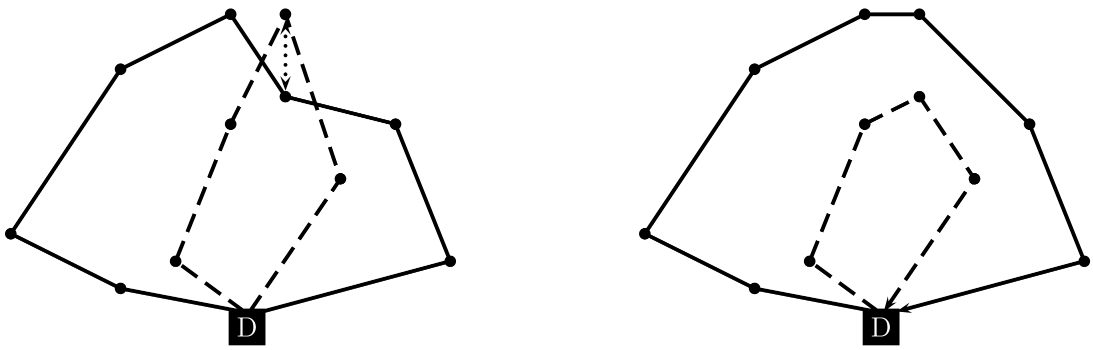
Figure 3.3: Example for 2-opt




(a) Solution before 3-opt (b) Solution after 3-opt

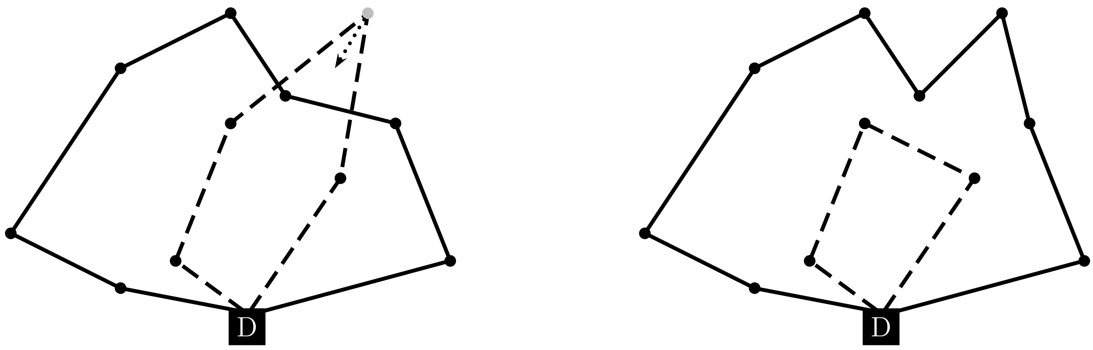
Figure 3.4: Example for 3-opt

- Or-opt (Or 1976)
  Or-opt displaces strings of consecutive customers with size 1,2 or 3 and puts them into another position of the route.




(a) Solution before Swap (b) Solution after Potential Swap

Figure 3.5: Example for Swap




(a) Solution before Move (b) Solution after Potential Move

Figure 3.6: Example for Move

2. *Inter-route Improvement Heuristics*
   Kindervater and Savelsbergh (1997) give a detailed overview on this type of improvement heuristics. As the feasibility check of the loading for one route requires a lot of computational time only inter-route improvement heuristics with simple and small neighborhoods are used within local search:

   - Swap
     The swap is characterized by swapping (exchanging) two customers of two different routes. Figure 3.5 illustrates the concept.
   - Move
     The move tries to move one customer from one route to another. Figure 3.6 illustrates the concept.

For all intra and inter-route improvement methods two different types can be distinguished:

- *First Improvement*: Whenever the improvement procedure finds an improvement, the solution is updated and the improvement procedure is re-initialized. When the complete neighborhood of an improvement heuristic (e. g. all 2-opt sequences of a route) does not lead to a better solution the improvement phase is stopped.
- *Best Improvement*: The complete neighborhood of an improvement heuristic is explored and the best improvement is realized by updating the solution and re-initializing the improvement procedure. Again, when no further improvements are found, the improvement procedure is stopped.

### 3.1.2 Metaheuristics

This section is dedicated to very short and not too detailed description of the main metaheuristics. First of all, what are metaheuristics?
The word metaheuristic consists of two words: meta and heuristic (Greek).

> Heuristics are criteria, methods, or principles for deciding which among several alternative courses of action promises to be me most effective in order to achieve some goal. They represent compromises between two requirements: the need to make such criteria simple and, at the same time, the desire to see them discriminate and correctly between good and bad choices (see Pearl 1984, p. 3).

Meta means above or beyond. Glover and Kochenberger (2003) define metaheuristics as follows:

> Metaheuristics, in their original definition, are solution methods that orchestrate an interaction between local improvement procedures and higher level strategies to create a process capable of escaping from local optima and performing a robust search for a solution space.

Figure 3.7 illustrates the basic intent of metaheuristics. Heuristics alone are often trapped in local optima, whereas metaheuristics should be able and are able to overcome this fault to find the optimal solution. The following metaheuristics have been successfully applied to the VRP:

- **Simulated Annealing (SA)**
  This method is inspired by annealing in metallurgy, a technique involving heating and controlled cooling of a material to increase its solidity and decrease defects. In simulated annealing a starting solution $\tilde{x}_0$ is generated by an appropriate construction heuristic. Then at each iteration $it$ a solution $\tilde{x}$ is drawn randomly in the

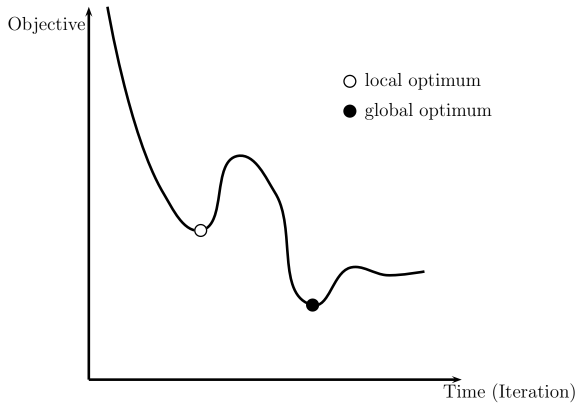



Figure 3.7: Basic Intent of Metaheuristics: Overcoming of Local Optima

neighborhood of $\tilde{x}_{it}$ denoted by $N(\tilde{x}_{it})$. $f(\tilde{x})$ represents the objective function and has to be minimized. If $f(\tilde{x}) \leq f(\tilde{x}_{it})$, then $\tilde{x}_{it+1}$ is set equal to $\tilde{x}$, otherwise

$$\tilde{x}_{it+1} = \begin{cases} \tilde{x} & \text{with probability } p^{it} \\ \tilde{x}_{it} & \text{with probability } 1 - p^{it} \end{cases}$$

where $p^{it}$ is a decreasing function of $it$, $f(\tilde{x}) - f(\tilde{x}_{it})$ and $\theta_{SA}^{it}$, where $\theta_{SA}^{it}$ is the temperature at iteration $it$. Usually $\theta_{SA}^{it}$ is defined by a decreasing step function, so that that the acceptance probability of worse solutions decreases with $it$ increasing. The basic idea of simulated annealing is explained by Kirkpatrick, Gelatt, and Vecchi (1983). Approaches for the VRP are given by Alfa, Heragu, and Chen (1991); Robuste, Daganzo, and Souleyrette (1990) and Osman (1993).

- **Tabu Search (TS)**
  In contrast to SA in tabu search the next move is made to the best neighbor solution of solution $\tilde{x}_{it}$. To avoid cycling a tabu list is kept. Cycling can easily occur when $f(\tilde{x}_{it}) > f(\tilde{x}_{it+1})$ and $\tilde{x}_{it}$ is the best neighbor of $\tilde{x}_{it+1}$. Hence $\tilde{x}_{it+2}$ and $\tilde{x}_{it}$ would be identical and the solution would cycle between $\tilde{x}_{it+1}$ and $\tilde{x}_{it}$. Therefore the best neighbor moves of the $\theta_{TS}$ last iterations are kept tabu (e. g.: move customer 1 between customers 2 and 3). This means that the moves in the tabu list are only performed if a new best global solution is found. Further details are given by Taillard (1993); Gendreau, Hertz, and Laporte (1994); Glover and Laguna (1997) and Toth and Vigo (2003).
- **Genetic Algorithms (GA)** The basic idea of genetic algorithms is inspired by the evolutionary mechanisms of all living beings on earth: crossover, inheritance,

mutation and selection. A very basic scheme of this method is given in Algorithm 3.4. $T$ stands for the number of generations and $\tilde{K}$ for the number of selections per

**Algorithm 3.4** Simple scheme of a genetic algorithm (see Gendreau, Laporte, and Potvin 2002, pp. 140–141)

```
Generate an initial random population X^0 = {x^01, ..., x^0N} and it := 0   ▷ initialization
while it < T do
    k := 0
    while k < K̃ do
        Select two parent chromosomes from X^it.   ▷ reproduction
        Generate two offspring from the two parent chromosomes using a crossover operator.   ▷ recombination
        Apply a random mutation to each offspring (with a small probability)   ▷ mutation
        k := k + 1
    end while
    Create X^(it+1) from X^it by removing the N worst solutions from X^it and adding the new N offspring.   ▷ generation replacement
    it := it + 1
end while
```

generation. Algorithms for the VRP and its variants have been proposed by Berger and Barkaoui (2003) and Potvin and Bengio (1996).

- **Memetic Algorithms (MA)**
  Moscato and Cotta (2003) give a very detailed description of MAs. The major difference to GAs is that the recombination and the mutation are both followed by a local search operator. Prins (2004) and Fallahi, Prins, and Calvo (2008) present memetic algorithms for the CVRP and a variant respectively.
- **Ant Colony Optimization (ACO)**
  See Section 3.2.
- **Variable Neighborhood Search (VNS)**
  VNS has been first proposed by Hansen and Mladenovic (1997) and makes use of different neighborhoods to avoid being trapped in local optima. The basic idea is represented in Algorithm 3.5.
  To the author's knowledge there is no publication on the CVRP with VNS but there is a very successful approach to the Multi-Depot Vehicle Routing Problem with Time Windows by Polacek et al. (2004).

**Algorithm 3.5** Basic Scheme of Variable Neighborhood Search (see Hansen and Mladenovic 2001, p. 451)

```
Find an initial solution.
Select a set of neighborhood structures N_k with k = 1, ..., k_max.
while Stopping condition is not met. do
    k := 1
    while k ≤ k_max do
        Generate a point x̃' at random from the k^th neighborhood of x̃ with x̃ ∈ N_k(x̃)   ▷ shaking
        Apply some local search method to x̃' to gain x̃'' as the local optimum of x̃'   ▷ local search
        if f(x̃'') ≤ f(x̃) then                                  ▷ Move or move not
            x̃ := x̃''
        else
            k := k + 1
        end if
    end while
end while
```

### 3.1.3 Exact Approaches

In the context of faster computers, more efficient commercial Linear Problem Solvers (like CPLEX 11.0 from ILOG and XPRESS 2007b from DashOptimization) researchers are also interested in providing exact solutions to the CVRP. According to the book of Toth and Vigo (2002*b*) and more recent approaches exact methods can be divided in

- branch-and-bound algorithms (see Toth and Vigo 2002*a*)
- branch-and-cut algorithms (see Naddef and Rinaldi 2002)
- set-covering-based algorithms (see Bramel and Simchi-Levi 2002)
- one and two-commodity formulation algorithms (see Letchford and Salazar-González 2006)
- branch-and-cut-and-price algorithms (see Fukasawa et al. 2006)

Letchford and Salazar-González (2006) analyze these different approaches theoretically and conclude that theoretical indications and practical evidence favor branch-and-cut-and-price algorithms. Indeed Fukasawa et al. (2006) are able to solve a lot of instances to optimality. They use 109 instances[1] out of 8 different well known benchmark test sets and can solve 102 instances to optimality. The largest instance solved (to optimality) has 261 (134) customers. The average gap to the lower bound of the remaining 6 instances (for one instance no gap and optimality status are reported) is about 7.1%.

[1]The problem instances and solutions are available online: `http://www.branchandcut.org/VRP/data`

## 3.2 Ant Colony Optimization for the CVRP

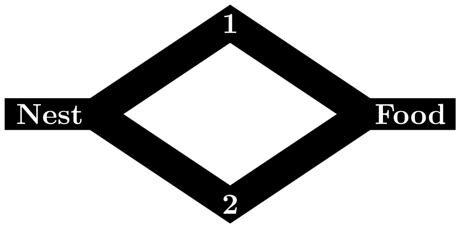
ACO has been used for a wide range of applications, among them are routing problems, assignment problems, scheduling problems, subset problems, etc.. Dorigo and Stützle (2004) give a very detailed description of this metaheuristic and its performance for the above mentioned optimization problems.

ACO is inspired by real ants and the experiments of Deneubourg et al. (1990) and Goss et al. (1989). Three of the conducted experiments are worth mentioning to get a notion of the meaning of pheromone for real and artificial ants. In the nature ants want to find the shortest distance between their nest (home) and potential food sources. Therefore the before mentioned scientists offered ants two equally long paths from their nest to the food (see Figure 3.8a). In the great majority of experiments the ants started to use only one branch after a certain amount of time. In the next step the experiment was extended as depicted in Figure 3.8b. Now the ants were offered to different choices to get to the food: a short and a long branch. In this setting in almost all experiments the ants start to use the short branch after a certain amount of time. The answer for this behavior lies in the pheromone concentration. Whenever an ant goes on its way it looses pheromone on the traveled path. Subsequent ants use the already deposited pheromone as information source to decide which way to go. As a consequence the pheromone concentration on the short branch is higher than on the long branch due to the fact that more ants can traverse the short branch compared to the long branch in the same amount of time. Apart from that scientists discovered that although the pheromone concentration on the short branch gets quite high there are still some ants that use the long branch as if they were willing to continue the exploration of the search space to possibly find a shorter way.

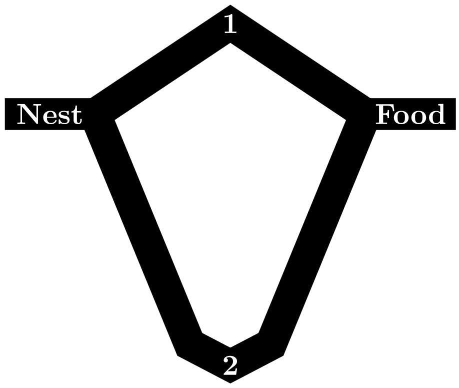
The last experiment shown in Figure 3.9 consists of two stages: First the ants have only one option to get to the food source: the long branch, meaning that pheromone is only deposited there. After 30 minutes the experiment setting is changed and the ants now have a second possibility to go: the short branch. Although a rather small number of ants starts to use the new shorter branch, they cannot overcome the pheromone concentration on the long branch and the majority of the population continues to use the long branch.

There are two papers that are most relevant for the CVRP and ACO: Reimann, Stummer, and Doerner (2002) and Reimann, Doerner, and Hartl (2004) which use the savings algorithm from Section 3.1. A previous approach by Bullnheimer, Hartl, and C. (1999) used a nearest neighbor heuristic and had a worse performance. Concerning these papers the basic scheme of ACO is illustrated in Algorithm 3.6.

The attractiveness values $\xi_{ij}$ representing the attractiveness combining customers $i$ and




(a) Experiment 1: Equal Branches




(b) Experiment 2: Unequal Branches

Figure 3.8: Basic Convergence Behavior of Real Ants

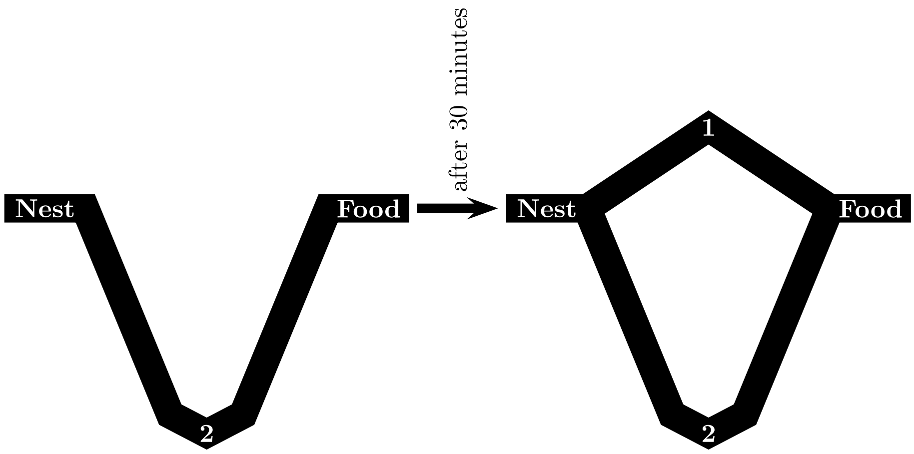



Figure 3.9: Experiment 3: Change of Setting after 30 Minutes

**Algorithm 3.6** Basic Scheme of Savings Based Ant Colony Optimization

**for** $it := 0$, $it < T$, $it := it + 1$ **do** ▷ Iterations
  Determine $\xi_{ij}$ by (3.1) and sort them in non-increasing manner
  **for** $a := 0$, $a < P$, $a := a + 1$ **do** ▷ Population size
    Initialize ant $a$.
    **while** Profitable and feasible combinations $(i, j)$ exist **do**
      Determine $\Omega_\Pi$
      Select two entries in $\Omega_\Pi$ randomly and merge the corresponding partial routes.
    **end while**
    Apply local search to ant $a$.
  **end for**
  Determine elitist solutions and perform pheromone update according to (3.3).
**end for**

$j$ are determined at each iteration. Note that these values are influenced by the static information of savings values (3.1), which do not change from iteration to iteration and $\tau_{ij}$, which represents pheromone information that is updated from iteration to iteration. Additionally there are two parameters $\alpha_1, \alpha_2$ to emphasize the heuristic and pheromone information properly. In each iteration an ant population of $P$ ants searches for feasible solutions according to the same knowledge base (pheromone and heuristic information). The solutions are constructed by a modified savings algorithm. In the basic version of this algorithm at each decision the most profitable customer pair is combined, whereas here in the neighborhood $\Omega_\Pi$ one potential combination is chosen randomly by a roulette wheel selection for the $\Pi$ most attractive combinations. The probabilities of the roulette wheel are determined by (3.2), meaning that potential combinations with higher attractiveness have a higher probability to be chosen than less attractive combinations. Note that in the real ant experiments ants did not always automatically follow the branch with the highest pheromone concentration, which is imitated by the roulette wheel selection. The restriction to $\Pi$ alternatives focuses the search. Whenever no more customers and thus routes can be merged, local search is applied to the current ant solution. At the end of each iteration the elitist solutions, which are the best solutions among all ant solutions of the current iteration, are allowed to update the pheromone information according to (3.3). On all edges the amount of $1 - \rho$ pheromone is evaporated, all elitist solutions are allowed to update the traversed edges by their rank: The best solution found so far updates each edge $i, j$ belonging to that solution by $\Delta\tau^*_{ij} = F/z$, where F is the number of elitists and z is the objective value of this solution. Furthermore the elitists of the current population are allowed to update each edge $i, j$ belonging to the elitist solution by $\Delta\tau^q_{ij} = (F - q)/z_q$.

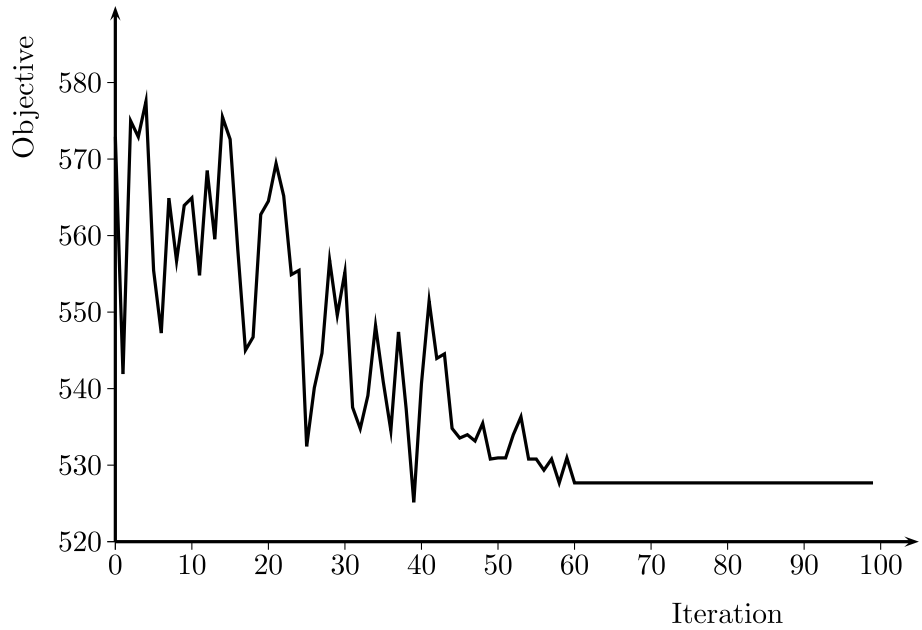



Figure 3.10: Convergence Behavior of Artificial Ants

Thus edges belonging to good solutions are amplified more than those belonging to worse solutions.

$$\xi_{ij} = (\tau_{ij})^{\alpha_1}(s_{ij})^{\alpha_2} \tag{3.1}$$

$$P_{ij} = \frac{\xi_{ij}}{\sum_{(h,l)\in\Omega_\Pi} \xi_{hl}} \tag{3.2}$$

$$\tau_{ij} = \rho\tau_{ij} + \sum_{q=1}^{F-1} \Delta\tau_{ij}^{q} + \Delta\tau_{ij}^{*} \tag{3.3}$$

Figure 3.10 shows the typical evolution of the objective during the execution of the algorithm. Note that the convergence is always close to the best solution found but not necessarily at the best solution found.

## 3.3 Ant Colony Optimization for the MP-VRP

Knowing of the good performance of the ACO for the classical CVRP, the natural question arose if it could also be tailored to solve problems with loading constraints. The first step was to find a quick and simple heuristic to check if a loading is feasible or not, which is explained in the following section:

### 3.3.1 Solution of the Loading Subproblem (MP-1-VLP)

Recalling that there are two types of items: short and long ones, where the latter need the length of the whole vehicle and the first only need one pile of the vehicle. This loading problem is related to two well known problems, namely the Bin Packing Problem (BPP) and the Parallel Processor Scheduling Problem ($P||C_{max}$).
The BPP calls for packing items characterized by weight or volume into the minimum number of homogenous bins with certain weight or volume capacity. The BPP has been addressed by branch and bound by Martello and Toth (1990) and Scholl, Klein, and Juergens (1997) and column generation by Vanderbeck (1999). Metaheuristic approaches have been proposed by Brugger et al. (2004) with ACO, Alvim et al. (2004) by TS, Fleszar and Hindi (2002) by VNS and Singh and Gupta (2007) by a GA.
The $P||C_{max}$ calls for minimizing the total makespan of $n$ jobs on $p$ parallel machines with $n > p \geq 2$. An exact approach has been presented by Dell'Amico and Martello (1995) with a branch-and-bound algorithm and by Chen and Powell (1999) with column generation. A multi-exchange neighborhood algorithm has been presented by Frangioni, Necciari, and Scutella (2004).
Determining the minimum height of the 1-VLP assuming that no long items have been ordered is equal to determining the minimum make span for the $P||C_{max}$ problem. However apart from the presence of long items there is also the LIFO loading constraint. The latter however reduces the complexity of the problem because the long items produce cuts in the packing (i. e. it is not possible to load all long items and determine the minimum makespan of the $P||C_{max}$ problem afterwards).
The basic idea of the loading heuristic $HL$ for the loading problem of the MP-VRP is based on the fact that nearly all customers order long items. This fact is used by forming pairs of customers when determining if the height $H$ of the vehicle is respected or not. The term pair defines a consecutive couple of customers $p_i$ and $p_{i+1}$. The items of these two customers are loaded in the vehicle such that the long items of $p_i$ and $p_{i+1}$ embrace the short items of these two customers in such way that the total height is minimized. This minimum height of one pair is represented by $\overline{h_{ij}}$. $\overline{h_i}$ represents the minimum height of loading one customer $i$ alone. $\overline{h_{ij}}$ and $\overline{h_i}$ can be computed by the following steps in a preprocessing phase:

1. Load the long items of customer $i$ (if there are any).
2. Load the short items of customers $i$ and $j$ by the branch-and-bound by Dell'Amico and Martello (1995). If the number of piles and the total number of short items of the two customers in consideration is small (like in this case) this can also be done

by enumerating all possible combinations.

3. Load the long items of customer $j$.

The obtained values are then used in Equation (3.4) if the number of customers in $\overline{P}$ is even and in Equation (3.5) if the number of customers is odd. In the latter Equation two cases are distinguished: either the first or the last customer is packed alone.

$$h(\overline{P}) = \sum_{i=1}^{|\overline{P}|/2} \overline{h}_{p_{2i-1},p_{2i}} \tag{3.4}$$

$$h(\overline{P}) = \min \left\{ \overline{h}_{p_1} + \sum_{i=1}^{(|\overline{P}|-1)/2} \overline{h}_{p_{2i},p_{2i+1}}; \sum_{i=1}^{(|\overline{P}|-1)/2} \overline{h}_{p_{2i-1},p_{2i}} + \overline{h}_{p_{|\overline{P}|}} \right\} \tag{3.5}$$

The worst case performance ratio of any Algorithm $A$ can be defined as the minimum value $WCP(A)$ such that $WCP(A) \geq UB(I)/z(I)$. $I$ stands for any instance $I$ of a problem and $z(I)$ is the optimal and $UB(I)$ the heuristic solution respectively. Doerner et al. (2007) prove the worst case performance ratio for $HL$ for the following two cases:

1. If all customers order long items, the $WCP(HL) \leq 2$
   This value is tight, which can be shown by the following example: $\overline{P} = 4$ and each customer orders a long item with height $\epsilon_1$. The first and the forth customer order a short item each with height $\epsilon_2$, whereas the second and third customer demand a short item each with a height of 1. $HL$ forms the following pairs $\underbrace{1-2}-\underbrace{3-4}$ with $h(\overline{P}) = 2 + 4\epsilon_1$. The optimal solution however is achieved by pairing customer $1-\underbrace{2-3}-4$ with $h^*(\overline{P}) = 1 + 4\epsilon_1 + 2\epsilon_2$. If $\epsilon_1$ and $\epsilon_2$ converge to zero $\frac{2+4\epsilon_1}{1+4\epsilon_1+2\epsilon_2} \to 2$.
2. If *not* all customers order long items $WCP(HL) \leq |\mathcal{P}|$. This means that the $WCP(HL)$ depends on the number of piles available on the vehicle. Again this value is tight, which can be shown by a different example: $|\overline{P}| = 2|\mathcal{P}|$ and no customer requests long items. Furthermore each customer in even position orders one short item with height 1 and each customer in odd position one item with height $\epsilon$. Assuming that $|\mathcal{P}| = 3$ $HL$ builds the following pairs $\underbrace{1-2}-\underbrace{3-4}-\underbrace{5-6}$. Each of this pairs only occupies two of the three available piles (one item with height $\epsilon$ and the other one with height 1. Therefore the $h(\overline{P}) = |\mathcal{P}| \cdot 1 = 3$. In contrast to that the loading pattern of the optimal solution is constructed by putting all items with height 1 (those of the even customers) side by side on the vehicle floor and the remaining items on top of them. The height of the optimal solution $h^*(\overline{P}) = 1 + \epsilon$ and $WCP(HL) = |\mathcal{P}|$.

| n | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 15 | 20 | 25 | 30 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $f_n$ | 1 | 1 | 2 | 2 | 3 | 4 | 5 | 7 | 9 | 12 | 49 | 200 | 816 | 3329 |

Table 3.1: Number of Customers and Corresponding Number of Possible Loadings

Of course there more possibilities of forming pairs than in the $HL$ heuristic, but how many of them are reasonable?

**Lemma 1** *The number of reasonable loading patterns, $f_n$, of $n$ customer demands can be computed by the recursion*

$$f_{n+1} = f_{n-2} + f_{n-1} \tag{3.6}$$

*with starting values $f_1 = 1,\ f_2 = 1$.*

**Proof.** Clearly, $f_1 = 1$ and $f_2 = 1$ since loading one customer is trivial and for loading two customers only one pair can be formed (which is always optimal). $f_3 = 2$, since with three customers the following pairs can be formed: $\underbrace{1-2}-3$ and $1-\underbrace{2-3}$
$f_4 = 2$, since the following pairs are possible $\underbrace{1-2}-\underbrace{3-4}$ and $1-\underbrace{2-3}-4$
$f_5 = 3$ with $\underbrace{1-2}-\underbrace{3-4}-5$, $1-\underbrace{2-3}-\underbrace{4-5}$ and $\underbrace{1-2}-3-\underbrace{4-5}$
The recursion from Lemma 1 can be derived by induction. Whenever a new customer $p_{n+1}$ is added to $\overline{P} = (p_1, p_2, \ldots, p_n)$ two cases can be distinguished:

a) Customer $p_n$ and customer $p_{n+1}$ do not form a pair. This only makes sense if $p_{n-1}$ and $p_n$ build a pair. Hence there are $f_{n-2}$ possibilities for arranging the first $n-2$ customers (see Figure 3.11a).
b) Customer $p_n$ and customer $p_{n+1}$ form a pair. Analogously there are $f_{n-1}$ possibilities for arranging the first $n-1$ customers (see Figure 3.11b).

Hence, in total there exist $f_{n+1} = f_{n-2} + f_{n-1}$ loading patterns. ■
Table 3.1 shows the series $f_{n+1}$ grows exponentially, but only when $n$ is sufficiently large. However during the search of the proposed metaheuristic the loading heuristic has to be called very often, which increases the runtime significantly (see also Section 4.1).
Apart from the simple heuristic $HL$ Doerner et al. (2007) introduce two dynamic programming approaches for solving the 1-VLP. The basic version $DP$ exploits the natural step structure within the problem but ignores the fact the some customers do not order long items (like the $HL$ heuristic).

**Proposition 2** *A better loading for a route with $n$ customers can be computed in linear time with effort*

$$2\,(n-2) \times additions + (n-2) \times comparisons \tag{3.7}$$

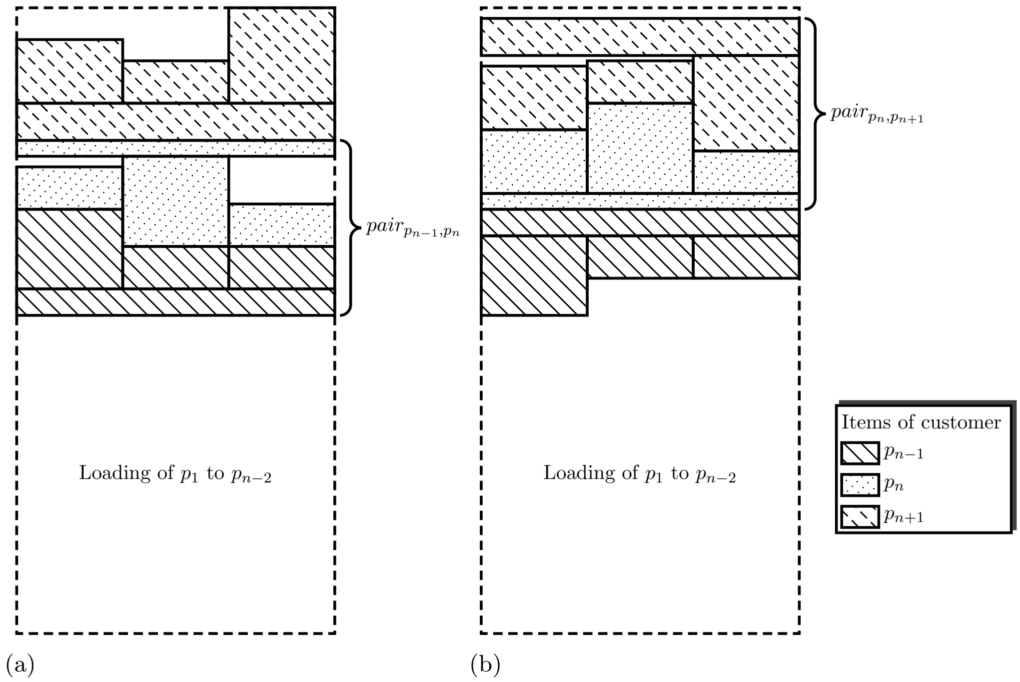



Figure 3.11: Possible Combinations of Customers

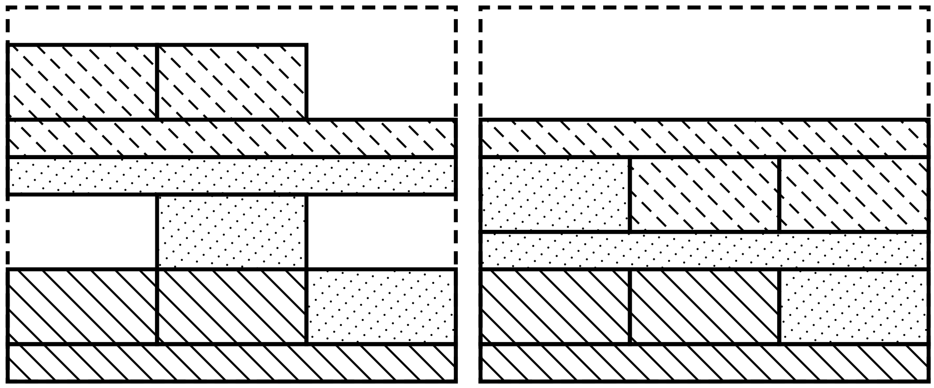
**Proof.** [2] This is done by induction. Let $L(p_1, \ldots, p_n)$ be the best loading height of the partial route $(p_1, \ldots, p_n)$. We start with $L(p_1) = \overline{h}_{p_1}$ and $L(p_1, p_2) = \overline{h}_{p_1,p_2}$. With 3 customers, the middle customer is either combined with the top or the bottom customer, i.e., $L(p_1, p_2, p_3) = \min\{L(p_1)+\overline{h}_{p_2,p_3}, L(p_1, p_2)+\overline{h}_{p_3}\}$. For extending the route to customer $p_{n+1}$, we again have to consider the 2 cases from Figure 3.11. Either the loading of the first $n$ customers remains unchanged and customer $p_{n+1}$ is added on top (not combined) leading to total height $L(p_1, \ldots, p_n) + \overline{h}_{p_{n+1}}$, or customers $p_n$ and $p_{n+1}$ are combined to form a pair giving total height $L(p_1, \ldots, p_{n-1}) + \overline{h}_{p_n,p_{n+1}}$. Taking the lower value of these two heights gives

$$L(p_1, ..., p_n) = \min\{L(p_1, \ldots, p_{n-2}) + \overline{h}_{p_{n-1},p_n},\ L(p_1, \ldots, p_{n-1}) + \overline{h}_{p_n}\}.$$

Summing up, for each of the customers from $p_3$ to $p_n$, two additions and one comparison have to be performed. ■

Note that $DP$ can find the optimal solution for the first example analyzing $WCP(HL)$, however it would also fail for the second example. $DP2$ makes use of the exact approach for the $P||C_{max}$ to deal with sequences where no long items are present. Let us first

[2]see Doerner et al. (2007), p. 299



(a) (b)

Figure 3.12: Example where $DP2$ Fails to Find the Optimal Solution

consider the simplest case with only one sequence of customers not demanding long items $(p_i, \ldots, p_j)$.

1. Use $DP$ to obtain a valid upper bound for $p_1, \ldots, p_{i-2}$.
2. The minimum loading height for customers $(p_{i-1}, p_i, \ldots, p_j, p_{j+1})$ is obtained by loading the long items of customer $p_{i-1}$, use the exact method by Dell'Amico and Martello (1995) to load the short items of $(p_{i-1}, p_i, \ldots, p_j, p_{j+1})$ and finally load the long items of customer $p_{j+1}$.
3. Use $DP$ to obtain a valid upper bound for $p_{j+2}, \ldots, p_n$.

This concept can be easily extended to cases where more sequences of customers do not order long items: For each sequence use the procedure described above and optimize the remaining parts with $DP$. $DP2$ is able to find the optimal solutions for both examples used in the explanation of $WCP(HL)$, however it does not always find the optimal solution. In Figure 3.12 three customers are considered. Each of them orders long items and two short items. Figure 3.12a shows the solution produced by $DP2$ and Figure 3.12b shows the optimal solution. The trick is to avoid forming pairs: One short item is placed below the long items of the customer in the middle and the second one item is positioned above.

### 3.3.2 Adaption of the ACO for the MP-VRP

The standard savings-based ACO from Section 3.2 has been adapted to solve the MP-VRP by three major elements:

1. Introduction of a measure for the loading part.
2. Introduction of second pheromone matrix for the loading part.
3. Update of the pheromone update mechanism for the loading part.

The additional visibility for the loading is realized by combining $\overline{h_{ij}}$ and $\gamma'_{ij}$. $\gamma'_{ij}$ represents the amount of bulk material needed when pairing customers $i$ and $j$ and $\gamma'_{ij}$ is given by Equation (3.8). Remember that bulk material is needed if long items are not fully supported by either short items below or the vehicle floor. $\gamma^{\max}$ is the maximum amount of bulk material needed when combining any customers $i$ and $j$ of the given problem instance which can be calculated in a preprocessing step.

$$\gamma'_{ij} = \gamma^{\max} - \gamma_{ij} \tag{3.8}$$

The new visibility $g_{ij}$ for the loading is given by Equation (3.9). If $g_{ij}$ is high, the combination of customers $i$ and $j$ leads to a high $\overline{h_{ij}}$ value with few bulk material needed. Similar to the First Fit Decreasing Heuristic for the BPP customer pairs with a higher $g_{ij}$ are preferred to the ones with lower $g_{ij}$. Creating $g_{ij}$ by multiplication has yielded better results than summation.

$$g_{ij} = \gamma'_{ij} \cdot \overline{h_{ij}} \tag{3.9}$$

Additionally the determination of the $\xi_{ij}$ values originally given by (3.1) is modified by incorporating the $g_{ij}$ values and a secondary pheromone matrix $\tau^p_{ij}$ for the loading according to Equation (3.10). The interpretation of this equation is quite simple: Whenever $\xi^{MP}_{ij}$ is high it is reasonable to combine customers $i$ and $j$. $\delta \in [0,1]$ is used to put either more weight on the loading or on the routing respectively. A sensitivity analysis in Section 4.1 will clarify its meaning. $\alpha_1, \alpha_2$ are parameters used for weighting the heuristic $(s_{ij}, g_{ij})$ and the pheromone $(\tau^r_{ij}, \tau^p_{ij})$ information respectively.

$$\xi^{MP}_{ij} = \delta\left[(s_{ij})^{\alpha_2}(\tau^r_{ij})^{\alpha_1}\right] + (1-\delta)\left[(g_{ij})^{\alpha_2}(\tau^p_{ij})^{\alpha_1}\right] \tag{3.10}$$

The last modification concerns the pheromone update: The $F$ routing elitists are allowed to update the routing pheromone matrix according to Equation (3.3). Moreover also $F$ loading elitists determined by calculating the total loading height of all solutions of an ant population are allowed to update the loading pheromone matrix by Equation (3.3). Note that $\Delta\tau^q_{ij} = (F-q)\iota$ for the $F-1$ best ants and $\Delta\tau^*_{ij} = F\iota$ for the best solution found so far, where $\iota$ is a small update constant. Additionally the loading elitists are only allowed to perform an pheromone update, when they do not use more vehicles than in the current best solution found so far. The parameter setting given in Reimann, Stummer, and Doerner (2002) and Reimann, Doerner, and Hartl (2004) proved to be a good choice also

| | | | |
|---|---|---|---|
| Number of ACO iterations ($T$) | $2n$ | Weight of pheromone ($\alpha_1$) | 5 |
| Number of ants (population size) ($P$) | $n/2$ | Weight of heuristic ($\alpha_2$) | 5 |
| Neighborhood size ($\Pi$) | $n/4$ | Trail persistency ($\rho$) | 0.95 |
| Number of elitists ($F$) | 6 | Update constant ($\iota$) | 0.0005 |
| Initial pheromone routing ($\tau_0^r$) | 2 | Initial pheromone packing ($\tau_0^p$) | 2 |

Table 3.2: Best Parameter Setting for ACO

for the MP-VRP and is summarized in Table 3.2. Concerning the local search mentioned in Algorithm 3.6 in line 9 a 2-opt, move and swap (see Section 3.1.1) have been used.

## 3.4 Ant Colony Optimization for the 2L-CVRP

This section gives an overview on the different approaches used to tailor the loading subproblem and the adoptions of the ACO to handle the more general problem (compared to the MP-VRP) with two dimensional loading constraints. It is also worth mentioning that the loading procedures for the 2L-CVRP can also be used to solve the MP-VRP. Nevertheless practical evidence has shown that the loading algorithms of the 2L-CVRP are much slower than the simple $HL$ procedure. This is mainly caused by the fact, that the simple $HL$ heuristic relies on values calculated in the preprocessing, why the 2L-CVRP loading algorithms start from cratch each time they are called.

### 3.4.1 Solution of the Loading Subproblem (2D-1-VLP)

Remember that the loading subproblem 2L has to deal with the four different cases from Section 2.2:

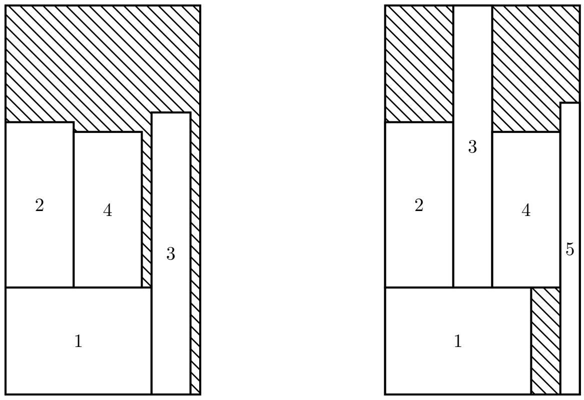
- 2|RO|L: rear loading, no rotation
- 2|UO|L: unrestricted loading, no rotation
- 2|RN|L: rear loading, with rotation
- 2|UN|L: unrestricted loading with rotation

2|UO|L can be solved with all available heuristics and exact approaches known from the Two-Dimensional Bin Packing Problem (2DBPP) and from the Two-Dimensional Strip Packing Problem (2DSPP).

The 2DBPP calls for packing two-dimensional items (rectangles with width $w$ and height $h$) into the minimum number of bins with width $W$ and height $H$ ($w \leq W$ and $h \leq H$). The 2DSPP calls for the minimization of the total height of packing two-dimensional items into a bin with given width $W$ and infinite height $H$. The 2DSPP has been addressed by an exact method by Martello, Monaci, and Vigo (2003) and very recently by Alvarez-Valdes, Parreño, and Tamarit (2008) with a GRASP algorithm. On the one hand the 2DBPP has been solved by (meta-)heuristic procedures by Lodi, Martello, and Vigo (1999) and Crainic, Perboli, and Tadei (2006). On the other hand it has been addressed by Martello and Vigo (1998) and Fekete, Scherpers, and Veen (2007) with exact approaches. A survey has been published by Lodi, Martello, and Monaci (2002). 2|UN|L corresponds to the orthogonal stock-cutting problem, which has been addressed by Burke, Kendall, and Whitwell (2004).

For solving the loading problem a combination of lower bounds, heuristics and a truncated branch and bound has been used. The lower bounds obtained by Martello and Vigo (1998) have been adapted to deal with rotation by the procedure proposed by Dell'Amico and Martello (2002). This simple procedure iteratively chooses an item and divides it into a list of smaller items, each of which has square dimensions. Assuming the procedure faces an item with $w = 5, l = 3$ it divides it into four items $\{(3,3),(2,2),(1,1),(1,1)\}$, which are used for determining the lower bound. If the obtained lower bound for a particular route is greater than one, no feasible packing exists.
If lower bounds do not indicate infeasibility, simple heuristics are applied to find a feasible loading. These heuristics pack one item at a time. The initial sorting of the items depends on the loading constraints. For 2|RO|L the items are sorted by scanning the customers associated with a particular route in reverse order of visit to meet the LIFO loading constraint. Additionally the items are sorted customerwise by non-increasing width, breaking ties by non-increasing length. For 2|RN|L the second sorting criteria for each customer items set is replaced by non-increasing area. Since for the remaining two cases the order of visit is irrelevant, the items are sorted by non-increasing width, breaking ties by non-increasing length for 2|UO|L and by non-increasing area for 2|UN|L. The following two heuristics have been implemented:

- *bottom-left fill* (see Burke, Kendall, and Whitwell 2004):
  The list of potential placing positions is sorted in non-increasing length breaking ties by non-increasing width. If items may be rotated (2|UN|L, 2|RN|L) first the oriented placement is checked and the rotated afterwards only if the oriented position is infeasible. The current item to pack is placed in the lowest leftmost position.
- *touching perimeter algorithm* (see Lodi, Martello, and Vigo 1999):
  Similar to the the first heuristic a list of potential placing points is kept. The next item to pack is placed in that position that maximizes the touching perimeter of the item with other items and the vehicle borders. Again if rotation is allowed both placements (oriented and rotated) are checked.




(a) Bottom-left Fill Heuristic

(b) Touching Perimeter Heuristic

Figure 3.13: Loading Heuristics Used for 2L-CVRP

The interested reader is referred to Section 3.5.1 for more details on the heuristics because all information there can be used one-to-one for the two-dimensional case here. Assuming that these two routines have to find a loading for the 2|UN|L case with the following items: $I = \{(15,11),(7,17),(4,29),(7,16),(2,30)\}$ and vehicle dimensions $(20,40)$. These items are already sorted by non-increasing area. Figure 3.13a shows the infeasible loading produced by bottom-left fill ( Item 5 cannot be placed). The procedure starts by placing item 1 with its left bottom corner at coordinates $(0,0)$. Item 2 cannot be placed right of Item 1 without exceeding $W$ and is placed at $(0,11)$. Item 3 could be placed at $(7,11)$ or at $(15,0)$ and is put at the latter position due to the lower y-coordinate. Item 4 can only be placed at $(7,11)$ and there is no feasible placing point left for the last item. Figure 3.13b shows a feasible loading for this instance obtained by the touching perimeter heuristic. The major difference is the placement of item 3 in $(7,11)$ due to the highest touching perimeter (bottom and top edge plus the length of item 2, which is longer than item 1). Item 4 and 5 can only be placed at $(11,11)$ and $(18,0)$ respectively.

If these heuristics fail to provide a feasible loading solution, local search is applied by switching the positions of two items in the input order given to the heuristics. Whenever two items have exchanged their positions the new item sequence is again passed to the loading heuristics. If LIFO loading is required, in the first phase only switchings of items belonging to same customers are allowed. whereas in the second phase exchanges of items belonging to different customers are also allowed. The rule for switching is the same for both phases. Fix one position and exchange the according item with all subsequent positions. Assuming the initial item sequence consists of three customers (indicated by brackets) and nine items: $\underbrace{1-2-3-4}-\underbrace{5-6}-\underbrace{7-8-9}$ the first phase (intra customer) gives:

$$\underbrace{2-1-3-4}-\underbrace{5-6}-\underbrace{7-8-9}$$
$$\underbrace{3-2-1-4}-\underbrace{5-6}-\underbrace{7-8-9}$$
$$\underbrace{4-2-3-1}-\underbrace{5-6}-\underbrace{7-8-9}$$
$$\underbrace{1-3-2-4}-\underbrace{5-6}-\underbrace{7-8-9}$$
$$\underbrace{1-4-3-2}-\underbrace{5-6}-\underbrace{7-8-9}$$
$$\underbrace{1-2-4-3}-\underbrace{5-6}-\underbrace{7-8-9}$$
$$\underbrace{1-2-3-4}-\underbrace{6-5}-\underbrace{7-8-9}$$
$$\underbrace{1-2-3-4}-\underbrace{5-6}-\underbrace{8-7-9}$$
$$\underbrace{1-2-3-4}-\underbrace{5-6}-\underbrace{9-8-7}$$
$$\underbrace{1-2-3-4}-\underbrace{5-6}-\underbrace{7-9-8}$$

and for the second phase (inter customer):

$$\underbrace{5-2-3-4}-\underbrace{1-6}-\underbrace{7-8-9}$$
$$\underbrace{6-2-3-4}-\underbrace{5-1}-\underbrace{7-8-9}$$
$$\underbrace{7-2-3-4}-\underbrace{5-6}-\underbrace{1-8-9}$$
$$\vdots$$
$$\underbrace{1-2-3-4}-\underbrace{5-9}-\underbrace{7-8-6}$$

If LIFO loading is not required the first and the second phase are mixed, meaning that item 1 is exchanged with item 4, followed by the exchange with item 5, 6, etc.. The number of permutations investigated is limited to $\phi$, which is a multiple of the number of customers of the investigated route ($\phi = 5 \cdot |\overline{P}|$). Investigating all possible permutations is far too time consuming.

If at this stage no feasible solution has been obtained a truncated branch-and-bound is passed the initial item sequence. The approach from Iori, Salazar-González, and Vigo (2006) has been adapted for the unrestricted, non-oriented case. At each node of the enumeration tree, the remaining set of placing points is limited to the so-called corner points (see Martello and Vigo 1998). Backtracking is performed when an item cannot enter any of the remaining corner points. The rotation is considered by enlarging the enumeration tree to be able to consider both orientations in each corner point. The branch-and-bound is halted after a certain number of backtracking steps (1000), or after the maximum CPU time allowed has been reached (10 seconds), or when a feasible solution has been found. Note that the set of placing points used within the heuristics is larger than the set of corner points within the branch-and-bound. This is due empirical observations of Gendreau et al.

(2008) which also tried corner points for the heuristics without convincing results. It is also interesting that Gendreau et al. (2008) additionally tried to construct a TS algorithm for the loading using the simple local search approach mentioned above instead of the branch-and-bound. Since the branch-and-bound outperformed the TS approach, it has been discarded.

Apart from that keeping a pool of checked routes that have been regarded feasible or infeasible has led to a dramatic reduction in run times (around 70%). This pool is sorted by the routing cost of the different routes. If the route of interest matches a route in the solution pool, it is not necessary to call the packing routines.

The pseudo code for the overall packing procedure CHECKLOADING is given in Algorithm 3.7. It returns true if a feasible loading could be found, false otherwise.

**Algorithm 3.7** Loading Procedure

**procedure** CHECKLOADING($\mathcal{I}(\overline{P}), \phi$) ▷ $\mathcal{I}(\overline{P})$ = set of items in route
    Found := CHECKPOOL($S(k)$,Feas) ▷ Feas = **true** if route is feasible
    **if** Found=**true then return** Feas
    **end if**
    L := $LB_{MV}(\mathcal{I}(\overline{P}))$ ▷ Lower bounds from Martello and Vigo 1998
    **if** $L > 1$ **then**
        Store $\mathcal{I}(\overline{P})$ in solution pool
        **return false**
    **end if**
    Create initial order $O$
    $\Psi$ := BOTTOMLEFT$_{2L}$($\mathcal{I}(\overline{P}), O$) ▷ $\Psi$ = Length of the loading
    $\Psi$ := min{$\Psi$, TOUCHINGPERIMETER$_{2L}$($\mathcal{I}(\overline{P}), O$)}
    $it := 1$
    **while** $\Psi > L$ **and** $it \leq \phi$ **do**
        Select two items and switch their positions in $O$
        $\Psi$ := min{$\Psi$, BOTTOMLEFT$_{2L}$($\mathcal{I}(\overline{P}), O$)}
        $\Psi$ := min{TOUCHINGPERIMETER$_{2L}$($\mathcal{I}(\overline{P}), O$)}
        $it := it + 1$
    **end while**
    **if** $\Psi > L$ **then** $\Psi$ := min{$\Psi$, BRANCHANDBOUND($\mathcal{I}(\overline{P}), O$)}
    **end if**
    Store $\mathcal{I}(\overline{P})$ in solution pool
    **if** $\Psi \leq L$ **then return true**
    **else return false**
    **end if**
**end procedure**

### 3.4.2 Adaption of the ACO for the 2L-CVRP

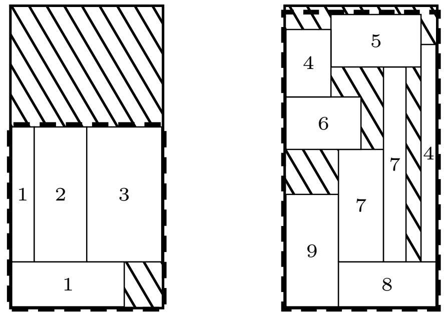
In contrast to the MP-VRP the number of vehicles available at the depot is not free but limited to $K$. This limit can be very strict according to the average capacity $d^{util} = \frac{\sum_{i \in \mathcal{V}} d_i}{DK}$ (up to 96%) and area utilization $a^{util} = \frac{\sum_{i \in \mathcal{V}} a_i}{WLK}$ (up to 90%) per vehicle. Therefore two main modifications are implemented to be able to generate feasible and good quality solutions for the 2L-CVRP. On the one hand a heuristic measure has been introduced to bias the search in the construction phase towards loading and on the other hand the local search applied to the ant solutions has been modified, penalizing the excess of the vehicle capacity as well as the given number of vehicles.

In the solution construction phase the creation of $\xi_{ij}$ from (3.1) is modified so as to obtain $\xi_{ij}^{2L}$ given in (3.11).

$$\xi_{ij}^{2L} = (\tau_{ij})^{\alpha_1} (s_{ij})^{\alpha_2} \left( \frac{\sum_{g \in \overline{P}} a_g}{\widetilde{a}_{\overline{P}}} \right)^{\alpha_3} \tag{3.11}$$

$\overline{P}$ is the set of customers generated by combining the two partial routes containing customers $i$ and $j$. $a_g$ is the total area of all items belonging to customer $g$ and $\widetilde{a}_{\overline{P}}$ is the area created by a *virtual rectangle*, i. e. the smallest rectangle containing all items currently loaded into the vehicle as depicted in Figure 3.14. Note that the loading algorithms described before have not been trimmed to find loadings minimizing the total virtual rectangle, therefore the virtual rectangle of the first feasible loading is taken. The idea behind the virtual rectangle can be easily explained with an example. In Figure 3.14 the algorithm has to decide wether to combine the partial routes $(0-3-2-0)$ and $(0-1-0)$ or to combine $(0-4-5-6-0)$ and $(0-7-8-9-0)$. Assuming for the sake of simplicity that the two alternatives have identical savings and pheromone values (thus $s_{21} = s_{67}, \tau_{21} = \tau_{67}$) it is clear that the decision for one alternative is only influenced by the loading information obtained by the virtual rectangle. The covered ratio $\frac{\sum_{g \in \overline{P}} a_g}{\widetilde{a}_{\overline{P}}}$ is 94% for Figure 3.14a and only 79% for Figure 3.14b. For runtime reasons the initial sorted list obtained at the beginning of each iteration of the algorithm is determined using the basic Equation (3.1), but within $\Omega_\Pi$ the $\xi_{ij}^{2L}$ values obtained by (3.11) are used. The virtual rectangle is a new fast measure for evaluating the quality of the loading of a vehicle. The envelope or contour as defined in Martello, Pisinger, and Vigo (2000) was also used to evaluate the quality of the loading, which led to higher run times while the solution quality did not improve.

However there were still few instances that could not be solved. Therefore the objective function was modified, so as to obtain (3.12).




(a) (b)

Figure 3.14: Virtual Rectangles Used to Bias Search towards Better Loadings

$$z'(\tilde{x}) = z(\tilde{x}) + \eta_1 q(\tilde{x}) + \eta_2 u(\tilde{x}) \tag{3.12}$$

$$z(\tilde{x}) = \sum_{k=1}^{\tilde{r}} c_r(k) \tag{3.13}$$

$$q(\tilde{x}) = \sum_{k=1}^{\tilde{r}} \left[ \sum_{i \in P(k)} d_i - D \right]^+ \tag{3.14}$$

$$u(\tilde{x}) = \begin{cases} \frac{KWLD}{\sum_{k=1}^{\tilde{r}} \left[ \sum_{i \in \overline{P}(k)} d_i \cdot \sum_{i \in \overline{P}(k)} a_i \right]} & \text{if } \tilde{r} > K \\ 0 & \text{otherwise} \end{cases} \tag{3.15}$$

The three terms from (3.12) are explained in Equations (3.13) - (3.15): $c_r(k)$ represents the total routing cost of route $k$. The next term is a penalty for exceeding the vehicle capacity $D$. Infeasible solutions with respect to vehicle capacity are only allowed if the average weight capacity utilization is greater or equal than 90% ($d^{util} = \frac{\sum_{i \in \mathcal{V}} d_i}{KD} \geq 0.9$). The penalty for exceeding the vehicle capacity is a constant ($\eta_1$), which is set to 100. The last term in the modified objective function is the most important one and comes only into play, if the ant solution exceeds the available number of vehicles $K$. Using this term the local search to a certain extent can accept worse solutions concerning $z(s)$ but with a better capacity/area utilization. Assume that in the current solution $K+1$ vehicles are used. Local search has to move to a solution with $K$ vehicles. Unfortunately this is most often impossible by just moving one customer from one route to another. Therefore local search must be able to decide which of the various solutions with $K+1$ vehicles is "nearer" to a solution with $K$ vehicles. This is ensured by $u(\tilde{x})$, which can be illustrated by a simple example: Assuming that $K = 1$ and an ant has generated a solution $\tilde{x}_a$ with two routes and the following characteristics $\sum_{i \in \overline{P}_{(1)}} d_i = 5$, $\sum_{i \in \overline{P}_{(1)}} a_i = 5$, $\sum_{i \in \overline{P}_{(2)}} d_i = 5$ and $\sum_{i \in \overline{P}_{(2)}} a_i = 5$. $L = 5$, $W = 2$ and $D = 10$. Then $u(\tilde{x})$ is equal to $\frac{1 \cdot 2 \cdot 5 \cdot 10}{5 \cdot 5 + 5 \cdot 5} = 2$. Supposing that, local search changes the solution to $\tilde{x}_b$, so as to obtain $\sum_{i \in \overline{P}_{(1)}} d_i = 7$, $\sum_{i \in \overline{P}_{(1)}} a_i = 8$, $\sum_{i \in \overline{P}_{(2)}} d_i = 3$ and $\sum_{i \in \overline{P}_{(2)}} a_i = 2$. $u(\tilde{x})$ then becomes $\frac{1 \cdot 2 \cdot 5 \cdot 10}{7 \cdot 8 + 3 \cdot 2} \approx 1.61$. If the increase in routing costs is smaller than the difference in $u(\tilde{x})$ meaning $z(\tilde{x}_a) - z(\tilde{x}_b) < [u(\tilde{x}_a) - u(\tilde{x}_b)]\eta_2$, $\tilde{x}_b$ is accepted. $\eta_2$ is initialized with 1000 and adapted dynamically from iteration to iteration according to the following rule: If more than half of the locally optimized solutions of a population exceed the given number of vehicles $\eta_2$ is doubled, otherwise $\eta_2$ is halved. Additionally $\eta_2$ is bounded in the interval $[10, 10^9]$. Algorithm 3.8 represents the whole algorithm in a pseudo code fashion. Note that the pheromone update is identical to the one used for the MP-VRP, where $\Delta\tau_{ij}^{q} = (F - q)\iota$ for the $F - 1$ best ants and $\Delta\tau_{ij}^{*} = F\iota$ for the best solution found so far, where $\iota$ is a small update constant.

**Algorithm 3.8** Standard Savings Based ACO

```
procedure AntAlgorithm(V, T, P, Π)                                   ▷ see Table 3.3
    for it := 0, it < T, it := it + 1 do                              ▷ Iterations
        Determine ξ_ij from (3.1) and sort them in non-increasing manner
        for a := 0, a < P, a := a + 1 do                              ▷ Population size
            Initialize ant a
            while Profitable and feasible combinations (i, j) exist do
                Determine Ω_Π and update ξ_ij^2L from (3.11)
                Select two entries in Ω_Π and merge the corresponding partial routes
            end while
            Apply local search to ant a
        end for
        Determine elitist solutions and perform pheromone update according to (3.3)
    end for
end procedure
```

Reimann, Doerner, and Hartl (2004) observed that problem instances with more than 100 customers should be divided into several subproblems to accelerate the search. Since the benchmark instances incorporate up to 255 customers the following approach inspired by Taillard (1993) and outlined in Algorithm 3.9 has been implemented. First a feasible solution to the problem as a whole, the so-called *master solution* must be found. This master solution is used for decomposing the problem into subproblems with approximately the same number of customers $\lambda$. First the number of clusters $n_c$ is determined, then Algorithm 3.8 is executed to get a master solution. (If no feasible master solutions has been found the process is re-initiated. In all conducted experiments three iterations at maximum were necessary to provide a feasible master solution). For each route of the master solution the center of gravity is determined according to Miehle's Algorithm (see Domschke and Drexl 1996, pp. 167–172). The routes are then sorted by their center of gravities using the sweep algorithm. Afterwards the problem is clustered into $n_c$ clusters on the basis of routes (the starting route for the first cluster is chosen randomly). This guarantees that each cluster has at least one feasible solution. Each subproblem is then optimized with Algorithm 3.8. All subproblems are recombined to form a new master solution and the clustering step is repeated with this new master solution. Consequently the master problem as a whole is only solved at the beginning.

**Algorithm 3.9** Main Algorithm

```
procedure MAIN(𝒱, λ, κ)
    if n > 100 then
        if n/λ − ⌊n/λ⌋ < 0.5 then n_c := ⌊n/λ⌋                ▷ Number of Clusters
        else n_c := ⌈n/λ⌉
        end if
        T := 1; P := 10; Π := max{|V|, 50}/4
        ANTALGORITHM'(V, T, P, Π)                               ▷ Master Solution
        for k := 0, k < κ, k := k + 1 do                        ▷ Clustering Steps
            Compute the centers of gravity for the routes in the best solution found
            Sort the routes by their centers of gravity through the Sweep Algorithm
            Cluster the problem in n_c clusters (C_1, ..., C_{n_c})
            for l := 0, l < n_c, l := l + 1 do
                ñ = max{50, |C_l|}; T := 2ñ; P := ñ/2; Π := ñ/4
                ANTALGORITHM'(C_l, T, P, Π)
            end for
            Combine partial solutions of clusters into a new candidate global solution
        end for
    else
        ñ := max{50, |V|}; T := 2ñ; P := ñ/2; Π := ñ/4
        ANTALGORITHM'(V, T, P, Π)
    end if
end procedure
```

Figure 3.15 illustrates the concept of clustering. Assuming that we want to decompose the whole problem consisting of 15 customers into clusters of 7 customers, we get $n_c = 2$. The routes sorted by their center of gravity gives I - II - III - IV. Since the master solution

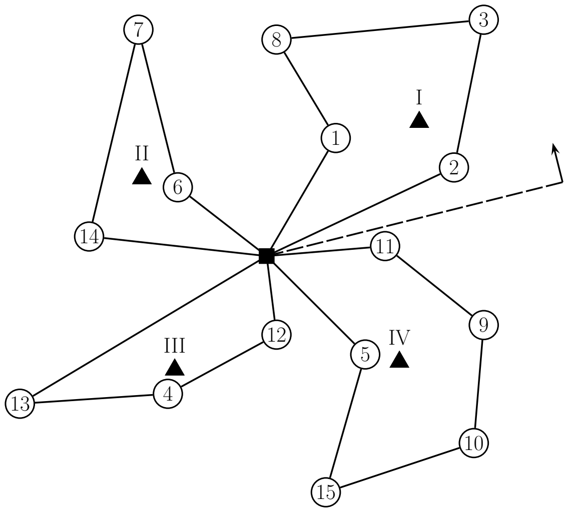



Figure 3.15: Clustering of an Instance through Sweep Algorithm.

consists of 4 routes, it is decomposed in two clusters with 2 routes each. If the starting route that is chosen randomly is III, cluster 1 consist of route III and IV, cluster 2 of I and II respectively.

The whole algorithm is halted when

- either $T$ iterations haven been performed (unclustered case, $n \leq 100$) or the problem has been clustered and optimized for $\kappa = 10$ times;
- or the total runtime limit of 3 CPU hours has been reached.

Table 3.3 shows the parameter setting for the ACO. Note that the parameters found by Reimann, Stummer, and Doerner (2002) and Doerner et al. (2007) proved to be a good choice for the 2L-CVRP too. Additionally the maximum number of permutations for the loading heuristics from Algorithm 3.7 was set to five times the number of customers in the particular route. The branch-and-bound procedure was allowed to perform 1000 backtracking steps with a time limit of 10 seconds of CPU time. These choices were all motivated by computational evidence, since they represent a good balance between solution quality and computational run time. Concerning the clustering of problems two questions had to be answered: At which problem size clustering should be applied and how large should the clusters be chosen? Both questions were answered with computational evidence: Clustering problems smaller than 100 customers ($n < 100$) yielded worse results. Choosing the cluster size $\lambda$ with 20, 30, 40 and 50 customers per cluster did not lead to significant performance differences, therefore $\lambda$ was set to 30. Note that in Fuellerer et al. (2007) the last term in (3.11) has not been equal $\alpha_3 = 5$ but equal to $\alpha_3 = 1$. However computational evidence in Section 4.2 will show slightly improved results in combination

| | | | |
|---|---|---|---|
| Number of ACO iterations ($T$) | see Algorithm 3.9 | Weight of pheromone ($\alpha_1$) | 5 |
| Number of ants ($P$) | see Algorithm 3.9 | Weight of savings ($\alpha_2$) | 5 |
| Neighborhood size ($\Pi$) | see Algorithm 3.9 | Weight of virt. rect. ($\alpha_3$) | 5 |
| Number of elitists ($F$) | 6 | Trail persistency ($\rho$) | 0.95 |
| Initial pheromone ($\tau_0$) | 2 | Update constant ($\iota$) | 0.0005 |
| Weight of capacity penalty ($\eta_1$) | 100 | | |
| Initial Weight of capacity/ area utilization | | ($\eta_1^0$) | 1000 |
| max. number of permutations in local search - loading heuristic ($\phi$) | | | $5 \cdot \lvert\overline{P}\rvert$ |
| max. number of backtracking steps of loading branch-and-bound | | | 1000 |
| max. time for loading branch-and-bound | | | 10 sec. |

Table 3.3: Best Parameter Setting for ACO ($\tilde{n} = \max\{50, |V|\}$).

with slightly shorter runtime.

## 3.5 Ant Colony Optimization for the 3L-CVRP

### 3.5.1 Solution of the Loading Subproblem (3D-1-VLP)

For the loading problem in the 2L-CVRP lower bounds, two heuristics and a truncated branch-and-bound have been used. For the the 3L-CVRP only the adapted version of the two heuristics has been applied, which has the following reasons: Firstly, in a preliminary test 10 million potential item sets (respecting the continuous lower bound, where the volume of all items does not exceed the vehicle volume) for one route have been tested with the current best available bounds by Boschetti (2004). If rotation is not allowed around 23% of these item sets are proven to be infeasible for $3|U|O|0|0|L$, if the lower bounds are tested on $3|U|N|0|0|L$ less than 1% can be excluded. Additionally the complexity of these lower bounds is $O(n^5)$ if the lower bound for the 1DBPP introduced by Boschetti (2004) is used and $O(n^4)$ if the continuous lower bound for 1DBPP is used. Apart from that Martello and Vigo (1998), Fekete and Schepers (1997), Fekete and Scherpers (1998) and Fekete and Schepers (2004) present lower bounds with a complexity of $O(n^2)$. However these bounds are both dominated by the bounds of Boschetti. Therefore there is no reason for using them because still for the rotation case only very few item sets can be disregarded. Secondly, there are indeed exact approaches for the 3DBPP that could be adapted to solve the loading problem of the 3L-CVRP. Nevertheless there is clear evidence that this is not that easy. The most recent approach for the 3DBPP is based on so-called interval graphs from graph theory by Fekete, Scherpers, and Veen (2007) and Fekete, Koehler, and Teich

(2001). The latter additionally takes precedence constraints into account, which would allow to handle the LIFO loading constraint. Unfortunately this approach is not capable for dealing with the supporting area constraint because the interval graph representation only depicts relations between items. Therefore one can say that item $q$ lies above item $\overline{q}$ for instance but the corresponding overlapping of these two items which is needed to calculate the supporting area cannot be derived from the graphs. Apart from that, there is the approach by Martello, Pisinger, and Vigo (2000); Martello et al. (2007). In fact these two publications present two different algorithms solving either the robot packing or the general packing.

> A *robot packing* is a packing which can be achieved by successively placing items starting from the bottom-left-behind corner, and such that each item is in front of, right of, or above each of the previously placed items. The name comes from the fact that robots used for packing boxes in the industry are equipped with a rectangular hand parallel to the base of the bins, which is covered with vacuum cells for lifting the boxes. To avoid collisions, it is demanded that no already packed box is positioned in front of, right of, or above the destination of the current box (see Martello et al. 2007, p. 1).

The robot packing is solved by extending of the idea of corner points being introduced for the 2DBPP by Martello and Vigo (1998). Nevertheless this approach is not capable for the 3L-CVRP: The corner point approach has a drawback, which unfortunately is also true for the heuristics that are going to be presented in the next paragraphs. Assume that the route $1-2-3-4$ (four customers and each orders exactly one item) has to satisfy the loading configuration $3|R|N|1|75|L$. In Figure 3.16a the loading solution for customers 2, 3 and 4 is shown achieved by using the corner point approach. For the remaining item of customer 4, only one feasible corner point (black dot) is allowed but it is infeasible due to supporting area. Even if allowing the item to be placed at the white dot, the solution is not feasible because the item would exceed $H$. However there exists a feasible solution by shifting the item of customer 2 backwards to gain enough space for placing the item of customer 1 (see Figure 3.16b).
The last exact approach mentioned by Martello et al. (2007) should be capable of solving the loading subproblem of the 3L-CVRP. Nevertheless there are many adaptions necessary concerning constraints and the algorithm has to be tuned for dealing with rotation which would again increase the runtime dramatically. In combination with the poor performance of lower bounds for the rotation case and the fact that only using heuristics leads to quite high computational runtimes compared to the 2L-CVRP case implementing an exact ap-

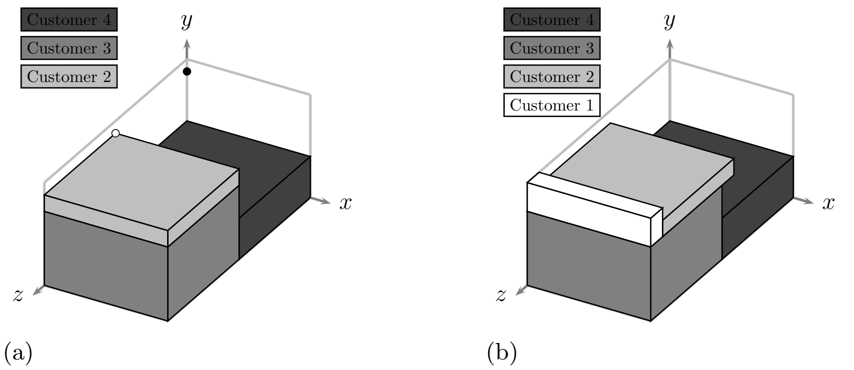
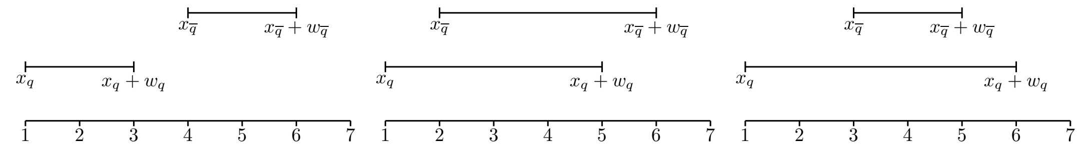



Figure 3.16: Placing Point Problem Illustrated by 4 Items of 4 different Customers

proach has been abandoned. However this leaves open questions for future research.

For solving the loading subproblem two heuristics are used that only differ in the criterion for choosing the next placing point. First of all for placing an item all combinations of x, y and z-coordinates are considered that can be derived from the endpoints of already placed items and the origin. Assuming that $\mathcal{I}(\hat{\overline{P}}) \subset \mathcal{I}(\overline{P})$ is the set of already placed items, for placing the next item all points that can be derived from the following three sets are checked: $\mathcal{X} = \{0\} \cup \bigcup_{q \in \mathcal{I}(\hat{\overline{P}})} x_q + w_q$, $\mathcal{Y} = \{0\} \cup \bigcup_{q \in \mathcal{I}(\hat{\overline{P}})} y_q + h_q$ and $\mathcal{Z} = \{0\} \cup \bigcup_{q \in \mathcal{I}(\hat{\overline{P}})} z_q + l_q$. For each combination out of these three sets one potential placing point $(x_q, y_q, z_q)$ is generated, which is checked for feasibility by Algorithms 3.10 and 3.11. The latter just returns the lengths along the x ($ol_x$), y ($ol_y$) and z ($ol_z$) axis concerning the overlapping of two items. Figure 3.17 illustrates the three possible cases that can occur when calculating the overlapping of two items:
(a) The items do not overlap.
(b) The items partially overlap.
(c) One item is completely overlapped.
Algorithm 3.10 checks if item $\overline{q}$ can be feasibly placed at $(x_q, y_q, z_q)$ in relation to the all items $q \in \mathcal{I}(\hat{\overline{P}})$ already placed. If $r(q) < r(\overline{q})$, item $q$ has to be unloaded before item $\overline{q}$. Note that $ol_x, ol_y$ and $ol_z$ can additionally be used for calculating the touching area between any two items.
For generating feasible loading patterns for particular routes the heuristics introduced in the 2L-CVRP are used:

- *Bottom-left-front*: For the next item $q$ to place the placing point with smallest $y_q$, breaking ties by smallest $x_q$, breaking ties by $z_q$ is chosen.
- *Touching area*: The next item is placed at that placing point that maximizes the




(a) $ol_x = 0$ (b) $ol_x = 3$ (c) $ol_x = 2$

Figure 3.17: Determination of Overlapping on x-coordinate Axis

touching area of this item with all already placed items and the vehicle walls. In Section 4.3 computational evidence will show the impact of ignoring the vehicle floor when calculating the touching area.

The initial sorting of all items belonging to all customers of one route depends on the set of constraints that is required:

- $3|R|$® $\in \{O, N\}|1|\underline{A} \in [0, 100]|L$: The items are sorted in reverse order of visit. The items of each customer are sorted by their fragility status, the non fragile ones are listed first, breaking ties by non-increasing volume.
- $3|R|$® $\in \{O, N\}|0|\underline{A} \in [0, 100]|L$: The items are sorted in reverse order of visit, the items of one customer are sorted by non-increasing volume.
- $3|U|$® $\in \{O, N\}|$Ⓕ $\in \{0, 1\}|\underline{A} \in [0, 100]|L$: All items are sorted by non-increasing volume.

Afterwards the sorted item list is passed to the bottom-left-front heuristic, if no feasible loading can be generated, it is passed to the touching area heuristic. While no feasible loading can be found, the initial item sequence is changed according to the procedure mentioned in Section 3.4.1 and the new item sequence is again passed to the two heuristics. After $\phi = 5 \cdot |\overline{P}|$ local search steps and no feasible loading solution the loading routine is stopped and returns infeasible. Additionally the solution pool of (in)feasible loading patterns as introduced in the loading procedure for the 2L-CVRP is also used here.

**Algorithm 3.10** Feasibility Check of a Placing Point for the Next Item to Place

**procedure** CHECKFEASPLACINGPOINT($\mathcal{I}(\hat{\overline{P}}), \overline{q}$)
  $\underline{a} := 0$
  **if** $x_{\overline{q}} + w_{\overline{q}} > W$ OR $y_{\overline{q}} + h_{\overline{q}} > H$ OR $z_{\overline{q}} + l_{\overline{q}} > L$ **then**
    **return** infeasible ▷ bin exceeded
  **end if**
  **for** $q \in \mathcal{I}(\hat{\overline{P}})$ **do**
    GETOVERLAPPING($x_q, y_q, z_q, x_{\overline{q}}, y_{\overline{q}}, z_{\overline{q}}, w_q, h_q, l_q, w_{\overline{q}}, h_{\overline{q}}, l_{\overline{q}}, ol_x, ol_y, ol_z$)
    **if** $ol_x > 0$ AND $ol_y > 0$ AND $ol_z > 0$ **then**
      **return** infeasible ▷ items intersect
    **end if**
    **if** $ol_x > 0$ AND $ol_z > 0$ **then** ▷ vertical relation
      **if** ($y_q + h_q = y_{\overline{q}}$ AND $\textcircled{F}_q = 1$ AND $\textcircled{F}_{\overline{q}} = 0$) OR ($y_q = y_{\overline{q}} + h_{\overline{q}}$ AND $\textcircled{F}_{\overline{q}} = 1$ AND $\textcircled{F}_q = 0$) **then**
        **return** infeasible ▷ fragility constraint violated
      **end if**
      **if** ($y_q < y_{\overline{q}}$ AND $r_{(q)} < r_{(\overline{q})}$) OR ($y_q > y_{\overline{q}}$ AND $r_{(q)} > r_{(\overline{q})}$) **then**
        **return** infeasible ▷ LIFO constraint violated
      **end if**
    **end if**
    **if** $ol_x > 0$ AND $ol_y > 0$ **then** ▷ relation along z axis
      **if** ($z_q < z_{\overline{q}}$ AND $r_{(q)} < r_{(\overline{q})}$) OR ($z_q > z_{\overline{q}}$ AND $r_{(q)} > r_{(\overline{q})}$) **then**
        **return** infeasible ▷ LIFO constraint violated
      **end if**
    **end if**
    **if** $y_q + h_q = y_{\overline{q}}$ **then**
      $\underline{a} := \underline{a} + (ol_x \cdot ol_z)$
    **end if**
  **end for**
  **if** $\frac{\underline{a}}{w_{\overline{q}} \cdot l_{\overline{q}}} \cdot 100 \geq \underline{A}$ OR $y_{\overline{q}} = 0$ **then**
    **return** feasible
  **else**
    **return** infeasible
  **end if**
**end procedure**

**Algorithm 3.11** Get Lengths of Overlapping of Two Items

```
procedure GetOverlapping(x_q, y_q, z_q, x_q̄, y_q̄, z_q̄, w_q, h_q, l_q, w_q̄, h_q̄, l_q̄, ol_x, ol_y, ol_z)
    if x_q̄ > x_q then                                ▷ Overlapping on x-axis
        ol_x = x_q + w_q − x_q̄
    else
        ol_x = x_q̄ + w_q̄ − x_q
    end if
    if ol_x > min(w_q, w_q̄) then
        ol_x = min(w_q, w_q̄)
    end if
    ol_x = max(0, ol_x)
    if y_q̄ > y_q then                                ▷ Overlapping on y-axis
        ol_y = y_q + h_q − y_q̄
    else
        ol_y = y_q̄ + h_q̄ − y_q
    end if
    if ol_y > min(h_q, h_q̄) then
        ol_y = min(h_q, h_q̄)
    end if
    ol_y = max(0, ol_y)
    if z_q̄ > z_q then                                ▷ Overlapping on z-axis
        ol_z = z_q + l_q − z_q̄
    else
        ol_z = z_q̄ + l_q̄ − z_q
    end if
    if ol_z > min(l_q, l_q̄) then
        ol_z = min(l_q, l_q̄)
    end if
    ol_z = max(0, ol_z)
end procedure
```

### 3.5.2 Adaption of the ACO for the 3L-CVRP

The modifications of the standard savings based ACO presented in Section 3.4.2 have proven to lead to very good solution quality and have been therefore directly taken over to the 3L-CVRP. The main difference is the extension of the virtual rectangle to the *virtual cuboid*, which again biases the search in the construction phase to good loading combinations weakening the pure influence of the $s_{ij}$ values according to Equation (3.16). Apart from that, also $u(\tilde{x})$ had to be adapted and is given by (3.17). In Section 4.3 the influence of $\alpha_4$ on the solution quality of the algorithm and computational time will be investigated. Moreover the parameter setting given in Table 3.3 except for the settings of the branch-and-bound is also used for the 3L-CVRP.

$$\xi_{ij}^{2L} = (\tau_{ij})^{\alpha_1} (s_{ij})^{\alpha_2} \left( \frac{\sum_{g \in \overline{P}} vol_g}{\widetilde{vol}_{\overline{P}}} \right)^{\alpha_4} \tag{3.16}$$

$$u(\tilde{x}) = \begin{cases} \frac{KWHLD}{\sum_{k=1}^{\tilde{r}} \left[ \sum_{i \in \overline{P}(k)} d_i \cdot \sum_{i \in \overline{P}(k)} vol_i \right]} & \text{if } \tilde{r} > K \\ 0 & \text{otherwise} \end{cases} \tag{3.17}$$

# 4 Results and Comparison to Other Solution Approaches

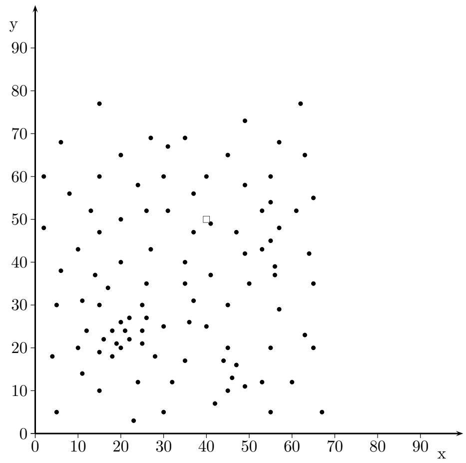
## 4.1 MP-VRP

### 4.1.1 Problem Instances

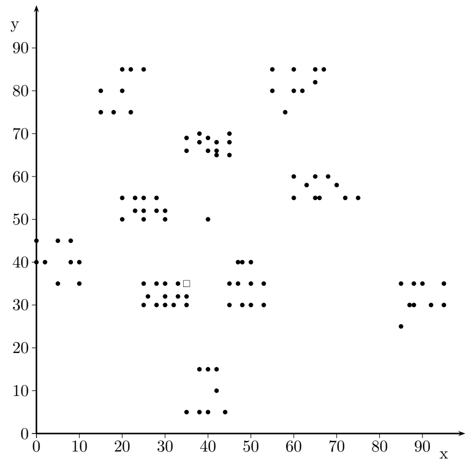
For the MP-VRP seven instances (out of 14) given in Christofides, Mingozzi, and Toth (1979) have been used for the geographic part (routing data) of the problem. The instances are available at `http://www.univie.ac.at/bwl/prod/VRPandBPP`. The second half of this 14 instances only differs in the additional constraint of a tour length restriction and has therefore not been considered in this work. These instances provide x and y coordinates in a two dimensional Euclidean space with $n$ customers (without depot) ranging from $50 \leq n \leq 199$. The first five instances are so-called random ones, meaning that the customer coordinates have been randomly generated in an predefined rectangle of the coordinate system (see Figure 4.1a). The last two instances are clustered instances (see Figure 4.1b).

For the creation of the loading specific data three different types of customers shown in Table 4.1 have been defined. According to this ordering behavior three different classes for each problem instance have been created. The different customer types are distributed among these classes as shown in Table 4.2. The first class mainly consists of customers with very small order quantities (cabinet makers who operate on a make-to-order policy) and customers with high demands (do-it-yourself stores). In class two all customer types are evenly distributed, where as in the last class 80% of the customers have a medium demand. The ordered quantity of each type of chipboards is randomly drawn from the interval in Table 4.1 according to a uniform distribution. The pallet height is set to 5, the heights of long, short, doors and heavy use chipboards are 2,1,2 and 3 respectively. The loading height of the truck is 200.

Apart from the randomly generated data, real data has been provided by a large Austrian wood retailer located in the north of Vienna (see Figure 4.2). For privacy reasons the demands have been slightly modified and redistributed randomly to the customers. The




(a) Random Instance `CMT03`, $n = 100$




(b) Clustered Instance `CMT07`, $n = 100$

Figure 4.1: Two Instances of the MP-VRP Plotted

| | customer type | | |
|---|---|---|---|
| | $demand_{min}$ | $demand_{mean}$ | $demand_{max}$ |
| long chipboards | $0 \leq d < 2$ | $4 \leq d < 6$ | $7 \leq d \leq 11$ |
| short chipboards | $0 \leq d < 9$ | $9 \leq d < 15$ | $21 \leq d \leq 33$ |
| chipboards for doors | $0 \leq d < 8$ | $8 \leq d < 12$ | $16 \leq d \leq 22$ |
| heavy use chipboards | $0 \leq d < 2$ | $4 \leq d < 6$ | $8 \leq d \leq 11$ |

Table 4.1: Demand Boundaries for Three Different Customer Types

| | customer type | | |
|---|---|---|---|
| Class | $demand_{min}$ | $demand_{mean}$ | $demand_{max}$ |
| 1 | 40% | 10% | 50% |
| 2 | 33% | 34% | 33% |
| 3 | 10% | 80% | 10% |

Table 4.2: Distribution of Customer Types over Different Classes

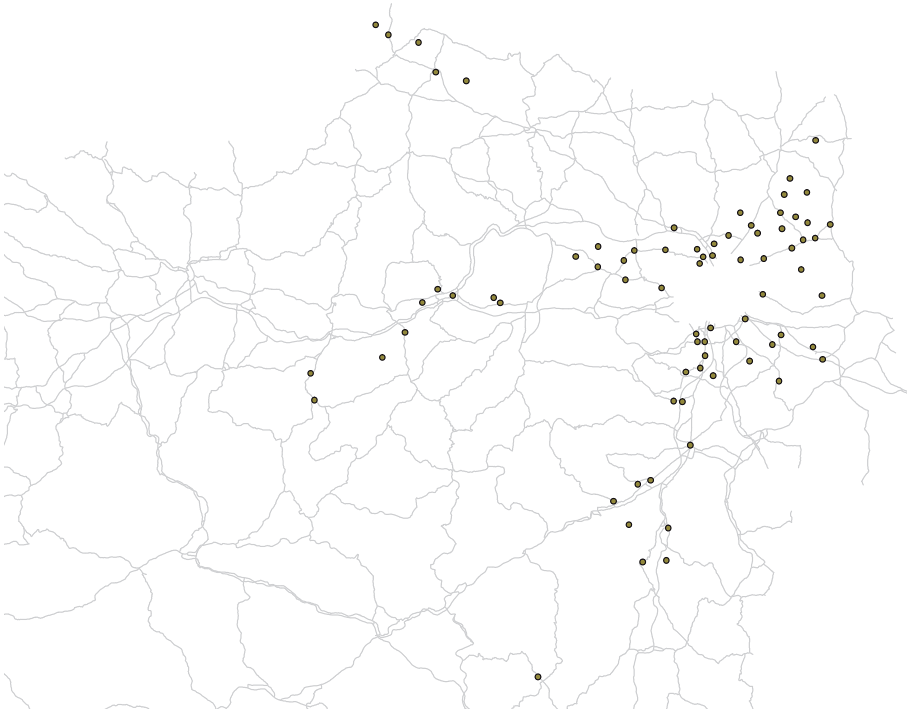


Figure 4.2: Map Showing the Real World Situation

cost matrix for both instance sets has been generated by using Euclidean distances.

### 4.1.2 Results on the Instances from the CVRP Literature

In this section four different solution approaches will be presented. On the one hand there are TS and ACO originally proposed for the MP-VRP and on the other hand there is a column generation proposed by Tricoire et al. (2007). Since the column generation approach unfortunately only succeeded on one of the smallest instances within reasonable computation time, these results are not reported here.
Table 4.3 shows a detailed comparison of the ACO and the TS approach. The first two columns give the name of the instance and the class respectively. The next columns contain the number of customers $n$ involved and the total number of items $M$ ordered. For the TS the solution quality $z$, the time $sec_h$ (in seconds) to find the best solution and the total runtime $sec_{tot}$ are reported. The ACO is a randomized (non-deterministic) algorithm. Therefore all presented data is based on ten runs per instance and class. The first three columns headed by ACO specify the best ($z_{min}$), average ($\overline{z}$) and worst solution ($z_{max}$) found, followed by the average runtime until the best solution has been found ($\overline{sec_h}$) and

the total average runtime of the algorithm ($\overline{sec}_{tot}$). The ACO algorithm has been stopped either after two hours runtime (which only happened with instance `CMT05`) or after the iteration limit has been reached (which has been set to $2n$). The last three columns give the percentage gap between the TS and the ACO in relation to the minimum, average and worst objective value of the ACO derived by $\frac{z_{TS}-z_{ACO}}{z_{TS}} \cdot 100$.
On the smallest instances `CMT01` TS outperforms ACO by 0.24% on average. Considering the average over all 21 instances ACO is on average better than TS by 0.53% (even by 0.22% if one only considers the worst solutions found by ACO). Apart from that ACO needs only half of the time to find its best solutions (1121.6 s against 2450.7 s) and additionally the total runtime is much shorter (1899.8 against 4954.5 seconds). Note that the runtime can be compared directly because both algorithms have been compiled with the same compiler and have been run on the same machine (2.4 GHz, 512MB RAM).

Table 4.4 shows the results aggregated for each instance, which again shows the slightly poor behaviour of the ACO on the small instances. Additionally the TS outperforms ACO with respect to instance `CMT07`, which is a clustered 100 customer instance. Table 4.5 shows an aggregation of results on classes. Here the average solution quality of the ACO is always better than the one obtained by the TS.
Apart from that for the smaller instances also the dynamic programming approach (see Section 3.3.1) under the assumption that all customers order long items has been tested: The computational runtime increased by more than 100%, whereas the solutions improved only by 0.18%. We do not provide detailed tables here but only report the results in condensed form. Furthermore the more advanced dynamic programming approach without the assumption that all customers order long chipboards has been tested. In this case the computational time increased such, that the average solution quality deteriorated by 3% due to the fact that most of the time the algorithm was stuck in searching for feasible loading patterns, rather than searching for good routing solutions.
After the main results of this chapter were published in Doerner et al. (2007), also a VNS approach was presented by Tricoire et al. (2007). A comparison of ACO with these results is presented in Table 4.6. The results presented for the VNS have been obtained by using the pairing heuristic $HL$ that has also been used in the ACO approach. The solution quality obtained by these two metaheuristics is nearly identical on average if both methods only operate with proven feasible solutions. In the VNS approach, however, also a variant was implemented, that also temporarily accepts solutions where the heuristic packing with heuristic HL exceeds the height of the vehicle slightly. Only if a potential new best solution is found, the feasibility is checked with an exact method. This approach gives an additional improvement. It is expected, that this idea could also work in connection with

| | | | | TS | | | ACO | | | | | gap ACO-TS | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $n$ | $M$ | $z$ | $sec_h$ | $sec_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| CMT01 | 1 | 50 | 170 | 594.06 | 12.4 | 2966.9 | 594.06 | 594.56 | 596.98 | 4.9 | 12.3 | 0.00 | 0.08 | 0.49 |
| | 2 | | 180 | 620.91 | 312.7 | 2229.2 | 622.64 | 622.82 | 623.08 | 5.9 | 11.5 | 0.28 | 0.31 | 0.35 |
| | 3 | | 193 | 636.95 | 1261.8 | 1809.0 | 637.41 | 638.97 | 640.93 | 4.9 | 11.4 | 0.07 | 0.32 | 0.62 |
| CMT02 | 1 | 75 | 278 | 990.51 | 119.1 | 2599.8 | 978.66 | 981.07 | 985.77 | 31.8 | 73.7 | -1.20 | -0.95 | -0.48 |
| | 2 | | 271 | 912.62 | 3409.6 | 3594.9 | 912.66 | 915.34 | 917.55 | 30.6 | 72.7 | 0.00 | 0.30 | 0.54 |
| | 3 | | 293 | 920.61 | 336.6 | 2694.8 | 916.48 | 917.84 | 922.86 | 29.6 | 72.4 | -0.45 | -0.30 | 0.24 |
| CMT03 | 1 | 100 | 355 | 1209.46 | 3481.8 | 4885.6 | 1194.66 | 1208.72 | 1224.53 | 144.6 | 364.3 | -1.22 | -0.06 | 1.25 |
| | 2 | | 380 | 1247.54 | 2740.1 | 3585.3 | 1234.95 | 1242.87 | 1248.16 | 139.4 | 349.2 | -1.01 | -0.37 | 0.05 |
| | 3 | | 387 | 1196.15 | 3576.6 | 3994.0 | 1185.72 | 1187.49 | 1189.26 | 130.8 | 366.8 | -0.87 | -0.72 | -0.58 |
| CMT04 | 1 | 150 | 529 | 1672.70 | 2660.6 | 7200.0 | 1648.39 | 1660.55 | 1676.81 | 1599.5 | 3978.3 | -1.45 | -0.73 | 0.25 |
| | 2 | | 548 | 1603.09 | 4925.4 | 7200.0 | 1566.90 | 1575.28 | 1580.15 | 1513.0 | 3998.5 | -2.26 | -1.73 | -1.43 |
| | 3 | | 580 | 1592.68 | 4902.5 | 7092.1 | 1578.06 | 1583.65 | 1589.18 | 1269.7 | 3940.5 | -0.92 | -0.57 | -0.22 |
| CMT05 | 1 | 199 | 712 | 2107.49 | 1717.4 | 7200.1 | 2077.57 | 2085.68 | 2092.32 | 5501.5 | 7200.0 | -1.42 | -1.03 | -0.72 |
| | 2 | | 707 | 1879.00 | 6611.1 | 7200.0 | 1853.98 | 1863.42 | 1872.34 | 5416.6 | 7200.0 | -1.33 | -0.83 | -0.35 |
| | 3 | | 773 | 2042.28 | 4282.6 | 7200.0 | 1988.83 | 1999.74 | 2014.74 | 5090.5 | 7200.0 | -2.62 | -2.08 | -1.35 |
| CMT06 | 1 | 120 | 421 | 2292.03 | 522.9 | 6406.4 | 2260.46 | 2269.56 | 2285.63 | 837.0 | 1466.8 | -1.38 | -0.98 | -0.28 |
| | 2 | | 447 | 2122.34 | 1784.2 | 7200.0 | 2087.84 | 2107.66 | 2120.44 | 719.5 | 1327.1 | -1.63 | -0.69 | -0.09 |
| | 3 | | 467 | 2237.86 | 2855.0 | 4927.7 | 2186.59 | 2195.66 | 2203.58 | 666.5 | 1368.7 | -2.29 | -1.89 | -1.53 |
| CMT07 | 1 | 100 | 346 | 1154.31 | 1944.3 | 5918.0 | 1142.78 | 1153.45 | 1161.34 | 150.2 | 302.9 | -1.00 | -0.07 | 0.61 |
| | 2 | | 375 | 1237.43 | 148.6 | 4088.1 | 1239.84 | 1248.83 | 1259.35 | 157.5 | 303.5 | 0.19 | 0.92 | 1.77 |
| | 3 | | 388 | 1183.18 | 3859.6 | 4052.3 | 1181.84 | 1182.92 | 1184.09 | 109.8 | 274.8 | -0.11 | -0.02 | 0.08 |
| AVG | | | | 1402.53 | 2450.7 | 4954.5 | 1385.25 | 1392.19 | 1399.48 | 1121.6 | 1899.8 | -0.98 | -0.53 | -0.22 |

Table 4.3: Performance of the TS and ACO on instances from the CVRP literature.

| | TS | | | ACO | | | | | gap ACO-TS | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | $z$ | $sec_h$ | $sec_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | $z_{min}$ | $\overline{z}$ |
| CMT01 | 617.31 | 529.0 | 2335.0 | 618.04 | 618.78 | 620.33 | 5.2 | 11.7 | 0.12 | 0.24 |
| CMT02 | 941.25 | 1288.4 | 2963.2 | 935.93 | 938.08 | 942.06 | 30.7 | 72.9 | -0.55 | -0.32 |
| CMT03 | 1217.72 | 3266.2 | 4155.0 | 1205.11 | 1213.03 | 1220.65 | 138.3 | 360.1 | -1.03 | -0.39 |
| CMT04 | 1622.82 | 4162.8 | 7164.0 | 1597.78 | 1606.49 | 1615.38 | 1460.7 | 3972.4 | -1.54 | -1.01 |
| CMT05 | 2009.59 | 4203.7 | 7200.0 | 1973.46 | 1982.95 | 1993.13 | 5336.2 | 7200.0 | -1.79 | -1.32 |
| CMT06 | 2217.41 | 1720.7 | 6178.0 | 2178.30 | 2190.96 | 2203.22 | 741.0 | 1387.5 | -1.76 | -1.19 |
| CMT07 | 1191.64 | 1984.2 | 4686.1 | 1188.15 | 1195.07 | 1201.59 | 139.2 | 293.7 | -0.31 | 0.27 |
| AVG | 1402.53 | 2450.7 | 4954.5 | 1385.25 | 1392.19 | 1399.48 | 1121.6 | 1899.8 | -0.98 | -0.53 |

Table 4.4: Aggregate Results per Instance (Three Classes per Line)

| | TS | | | ACO | | | | | gap ACO-TS | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | $z_{min}$ | $\overline{z}$ |
| 1 | 1431.51 | 1494.1 | 5311.0 | 1413.80 | 1421.94 | 1431.91 | 1181.4 | 1914.0 | -1.10 | -0.54 |
| 2 | 1374.70 | 2847.4 | 5013.9 | 1359.83 | 1368.03 | 1374.44 | 1140.4 | 1894.6 | -0.82 | -0.30 |
| 3 | 1401.39 | 3010.7 | 4538.6 | 1382.13 | 1386.61 | 1392.09 | 1043.1 | 1890.7 | -1.03 | -0.75 |
| AVG | 1402.53 | 2450.7 | 4954.5 | 1385.25 | 1392.19 | 1399.48 | 1121.6 | 1899.8 | -0.98 | -0.53 |

Table 4.5: Aggregate Results per Class (Seven Instances per Line)

the ACO approach.
Table 4.7 shows the performance of the two metaheuristics on the real world data set. Again TS is clearly outperformed by ACO but the average improvement is only around 0.3%. Nevertheless the ACO is much faster than TS (around 100 times). This large difference in runtime can partly be explained by the fact, that the real world instances are clustered (see Figure 4.2) and that the problem size is relatively small ($n = 76$).

### 4.1.3 Sensitivity Analysis of ACO

As described in Section 3.3.2 the standard ACO has been adapted to solve the MP-VRP. In this context the parameter $\delta$ has been introduced to guide the search with respect to routing and loading quality respectively. Table 4.9 and Table 4.8 shows the impact of $\delta$ on the average number of vehicles used and the according average solution quality. The last line in these tables represents the relative change with respect to the average number of vehicles used and average solution quality when $\delta = 1$. For the randomly generated instances, as well as for the real world data, the reduction of vehicles could not compensate the increase of total traveled distance. For the real instances a decrease of 6% concerning the number of vehicles causes an increase in routing cost by more than 17%. From this observation one may conclude that the additional pheromone for the loading information is not necessary, which has been abandoned for the 2L-CVRP and 3L-CVRP.

| | | ACO | | VNS | | | gap ACO-VNS | |
|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z_{min}$ | $\overline{z}$ | $z_{min}$ | $\overline{z}$ | $\overline{sec}_h$ | $z_{min}$ | $\overline{z}$ |
| CMT01 | 1 | 594.06 | 594.56 | 594.06 | 594.06 | 18 | 0.00 | 0.08 |
| | 2 | 622.64 | 622.82 | 620.91 | 620.91 | 175 | 0.28 | 0.31 |
| | 3 | 637.41 | 638.97 | 636.95 | 637.75 | 151 | 0.07 | 0.19 |
| CMT02 | 1 | 978.66 | 981.07 | 986.18 | 991.04 | 342 | -0.76 | -1.01 |
| | 2 | 912.66 | 915.34 | 912.06 | 913.06 | 18 | 0.07 | 0.25 |
| | 3 | 916.48 | 917.84 | 916.48 | 918.76 | 82 | 0.00 | -0.10 |
| CMT03 | 1 | 1194.66 | 1208.72 | 1204.03 | 1210.26 | 56 | -0.78 | -0.13 |
| | 2 | 1234.95 | 1242.87 | 1233.45 | 1236.72 | 122 | 0.12 | 0.50 |
| | 3 | 1185.72 | 1187.49 | 1188.31 | 1190.32 | 705 | -0.22 | -0.24 |
| CMT04 | 1 | 1648.39 | 1660.55 | 1658.28 | 1664.00 | 244 | -0.60 | -0.21 |
| | 2 | 1566.90 | 1575.28 | 1568.58 | 1577.59 | 367 | -0.11 | -0.15 |
| | 3 | 1578.06 | 1583.65 | 1578.71 | 1585.47 | 365 | -0.04 | -0.11 |
| CMT05 | 1 | 2077.57 | 2085.68 | 2072.84 | 2080.80 | 604 | 0.23 | 0.23 |
| | 2 | 1853.98 | 1863.42 | 1856.71 | 1866.02 | 482 | -0.15 | -0.14 |
| | 3 | 1988.83 | 1999.74 | 1995.71 | 1999.16 | 653 | -0.34 | 0.03 |
| CMT06 | 1 | 2260.46 | 2269.56 | 2258.99 | 2279.41 | 475 | 0.07 | -0.43 |
| | 2 | 2087.84 | 2107.66 | 2083.15 | 2100.21 | 1119 | 0.23 | 0.35 |
| | 3 | 2186.59 | 2195.66 | 2186.32 | 2208.57 | 586 | 0.01 | -0.58 |
| CMT07 | 1 | 1142.78 | 1153.45 | 1139.32 | 1141.29 | 584 | 0.30 | 1.07 |
| | 2 | 1239.84 | 1248.83 | 1228.81 | 1234.82 | 128 | 0.90 | 1.13 |
| | 3 | 1181.84 | 1182.92 | 1180.61 | 1182.80 | 743 | 0.10 | 0.01 |
| AVG | | 1385.25 | 1392.19 | 1385.74 | 1392.05 | 382 | -0.03 | 0.05 |

Table 4.6: Result Comparison of VNS by Tricoire et al. (2007) with ACO

| | | | TS | | | ACO | | | | gap ACO-TS | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $n$ | $M$ | $z$ | $sec_h$ | $sec_{tot}$ | $z_{min}$ | $\overline{z}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ |
| WOOD01 | 76 | 142 | 1616.68 | 2694.0 | 7200.1 | 1594.88 | 1599.71 | 36.9 | 68.0 | -1.35 | -1.05 |
| WOOD02 | 76 | 141 | 1483.94 | 7098.4 | 7200.0 | 1481.28 | 1491.98 | 49.8 | 76.8 | -0.18 | 0.54 |
| WOOD03 | 76 | 142 | 1389.88 | 1234.8 | 7200.0 | 1384.73 | 1388.41 | 37.7 | 68.0 | -0.37 | -0.11 |
| WOOD04 | 76 | 144 | 1485.71 | 6380.8 | 7141.6 | 1469.97 | 1477.19 | 41.0 | 72.1 | -1.06 | -0.57 |
| WOOD05 | 76 | 184 | 1494.18 | 4190.4 | 6323.4 | 1484.82 | 1490.08 | 40.0 | 70.5 | -0.63 | -0.27 |
| AVG | | | 1494.07 | 4319.8 | 7013.0 | 1483.14 | 1489.47 | 41.1 | 71.1 | -0.72 | -0.29 |

Table 4.7: Performance of TS and ACO on Real World Data

| | 0.8 | | 0.85 | | 0.9 | | 0.95 | | 1.00 | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | $\overline{z}$ | $\overline{NV}$ | $\overline{z}$ | $\overline{NV}$ | $\overline{z}$ | $\overline{NV}$ | $\overline{z}$ | $\overline{NV}$ | $\overline{z}$ | $\overline{NV}$ |
| WOOD01 | 1843.00 | 10.0 | 1786.53 | 10.0 | 1734.28 | 10.0 | 1668.05 | 10.0 | 1599.71 | 10.0 |
| WOOD02 | 1773.28 | 9.0 | 1698.44 | 9.0 | 1649.77 | 9.0 | 1581.83 | 9.0 | 1491.98 | 10.0 |
| WOOD03 | 1698.64 | 9.0 | 1647.41 | 9.0 | 1522.57 | 9.8 | 1453.55 | 10.0 | 1388.41 | 10.0 |
| WOOD04 | 1790.20 | 9.2 | 1676.78 | 9.6 | 1608.61 | 9.7 | 1554.60 | 10.0 | 1477.19 | 10.0 |
| WOOD05 | 1626.51 | 10.0 | 1598.39 | 10.0 | 1562.36 | 10.0 | 1541.98 | 10.0 | 1490.08 | 10.0 |
| AVG | 1746.33 | 9.4 | 1681.51 | 9.5 | 1615.52 | 9.7 | 1560.00 | 9.8 | 1489.47 | 10.0 |
| gap to $\delta = 1$ | 17.25 | -6.00 | 12.89 | -5.00 | 8.46 | -3.00 | 4.74 | -2.00 | 0.00 | 0.00 |

Table 4.8: Sensitivity Analysis of ACO for Different Values of $\delta$ for Real World Instances

| | | 0.8 | | 0.85 | | 0.9 | | 0.95 | | 1.00 | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $\overline{z}$ | $\overline{NV}$ | $\overline{z}$ | $\overline{NV}$ | $\overline{z}$ | $\overline{NV}$ | $\overline{z}$ | $\overline{NV}$ | $\overline{z}$ | $\overline{NV}$ |
| CMT01 | 1 | 603.86 | 7.0 | 599.91 | 7.0 | 598.90 | 7.0 | 595.93 | 7.0 | 594.56 | 7 |
| | 2 | 749.27 | 7.0 | 716.74 | 7.0 | 684.29 | 7.0 | 652.88 | 7.0 | 622.82 | 7 |
| | 3 | 645.62 | 8.0 | 645.00 | 8.0 | 642.71 | 8.0 | 641.04 | 8.0 | 638.97 | 8 |
| CMT02 | 1 | 1038.94 | 13.0 | 1015.62 | 13.0 | 1002.00 | 13.0 | 988.67 | 13.0 | 981.07 | 13 |
| | 2 | 1153.95 | 11.0 | 1121.80 | 11.0 | 1069.67 | 11.0 | 968.32 | 11.6 | 915.34 | 12 |
| | 3 | 965.51 | 11.9 | 941.32 | 12.0 | 945.44 | 11.9 | 933.29 | 12.0 | 917.84 | 12 |
| CMT03 | 1 | 1398.19 | 16.0 | 1375.17 | 16.0 | 1325.27 | 16.0 | 1251.93 | 16.8 | 1208.72 | 16.8 |
| | 2 | 1401.42 | 17.0 | 1366.85 | 17.0 | 1331.62 | 17.0 | 1290.92 | 17.0 | 1242.87 | 17 |
| | 3 | 1242.78 | 15.9 | 1240.03 | 15.9 | 1214.28 | 15.9 | 1201.69 | 15.9 | 1187.49 | 16 |
| CMT04 | 1 | 1961.36 | 23.9 | 1840.52 | 24.0 | 1791.73 | 24.0 | 1717.85 | 24.0 | 1660.55 | 24.2 |
| | 2 | 2049.09 | 22.4 | 1876.16 | 22.8 | 1784.74 | 23.0 | 1670.35 | 23.0 | 1575.28 | 23 |
| | 3 | 1649.65 | 23.0 | 1638.20 | 23.0 | 1623.85 | 23.0 | 1597.02 | 23.0 | 1583.65 | 23 |
| CMT05 | 1 | 2462.47 | 32.0 | 2342.48 | 32.0 | 2303.33 | 32.0 | 2171.84 | 32.2 | 2085.68 | 32.7 |
| | 2 | 2418.68 | 27.0 | 2282.98 | 27.0 | 2037.21 | 27.8 | 1949.83 | 28.0 | 1863.42 | 28 |
| | 3 | 2045.16 | 31.0 | 2055.26 | 31.0 | 2040.05 | 31.0 | 2018.12 | 31.0 | 1999.74 | 31.1 |
| CMT06 | 1 | 2645.22 | 18.0 | 2606.11 | 18.0 | 2513.42 | 18.0 | 2433.85 | 18.4 | 2269.56 | 19 |
| | 2 | 2429.27 | 18.0 | 2389.46 | 18.0 | 2333.19 | 18.0 | 2262.99 | 18.0 | 2107.66 | 18.1 |
| | 3 | 2373.26 | 18.2 | 2292.42 | 18.6 | 2227.36 | 18.9 | 2202.02 | 18.9 | 2195.66 | 19 |
| CMT07 | 1 | 1389.25 | 15.0 | 1350.00 | 15.0 | 1284.26 | 15.0 | 1213.65 | 15.0 | 1153.45 | 15.5 |
| | 2 | 1355.52 | 16.0 | 1331.63 | 16.0 | 1295.07 | 16.0 | 1266.51 | 16.0 | 1248.83 | 16.5 |
| | 3 | 1248.88 | 16.0 | 1233.66 | 16.0 | 1213.72 | 16.0 | 1198.37 | 16.0 | 1182.92 | 16 |
| AVG | | 1582.25 | 17.5 | 1536.25 | 17.5 | 1488.67 | 17.6 | 1439.38 | 17.7 | 1392.19 | 17.9 |
| gap to $\delta = 1$ | | 13.65 | -2.23 | 10.35 | -2.23 | 6.93 | -1.68 | 3.39 | -1.12 | 0.00 | 0.00 |

Table 4.9: Sensitivity Analysis of ACO for Different Values of $\delta$ for the Randomly Generated Instances

## 4.2 2L-CVRP

### 4.2.1 Problem Instances

The ACO algorithm was tested on the existing instances from the literature introduced by Iori, Salazar-González, and Vigo (2006) and Gendreau et al. (2008). In these instances, the geographical data and the weights demanded by customers were taken from instances of the CVRP literature (see Golden et al. (1998) and Christofides, Mingozzi, and Toth (1979)). The number of items and their dimensions of each customer were created according to five classes: Class 1 represents pure CVRP instances meaning that the loading aspect never becomes tight. In classes 2 to 5 the items have been uniformly generated in the intervals given in Table 4.10. Remember that $m_i$ stands for the number of items that are ordered by one customer. In total the instance set consists of $36 \times 5$ different instances containing between 15 and 255 customers and between 15 and 786 items. The instances can be downloaded from `http://www.or.deis.unibo.it/research.html`. For the sake of clarity only average results for each instance will be presented in this section, however the detailed results can be found in appendix A.

All experiments concerning the ACO algorithm have been implemented in C++, compiled with g++ compiler and run on a Pentium 4 with 3.2 GHz.

| | | $Vertical$ | | $Homogeneous$ | | $Horizontal$ | |
|---|---|---|---|---|---|---|---|
| $C$ | $m_i$ | $l_{iq}$ | $w_{iq}$ | $l_{iq}$ | $w_{iq}$ | $l_{iq}$ | $w_{iq}$ |
| 2 | $[1,2]$ | $\left[\frac{4L}{10}, \frac{9L}{10}\right]$ | $\left[\frac{W}{10}, \frac{2W}{10}\right]$ | $\left[\frac{2L}{10}, \frac{5L}{10}\right]$ | $\left[\frac{2W}{10}, \frac{5W}{10}\right]$ | $\left[\frac{L}{10}, \frac{2L}{10}\right]$ | $\left[\frac{4W}{10}, \frac{9W}{10}\right]$ |
| 3 | $[1,3]$ | $\left[\frac{3L}{10}, \frac{8L}{10}\right]$ | $\left[\frac{W}{10}, \frac{2W}{10}\right]$ | $\left[\frac{2L}{10}, \frac{4L}{10}\right]$ | $\left[\frac{2W}{10}, \frac{4W}{10}\right]$ | $\left[\frac{L}{10}, \frac{2L}{10}\right]$ | $\left[\frac{3W}{10}, \frac{8W}{10}\right]$ |
| 4 | $[1,4]$ | $\left[\frac{2L}{10}, \frac{7L}{10}\right]$ | $\left[\frac{W}{10}, \frac{2W}{10}\right]$ | $\left[\frac{L}{10}, \frac{4L}{10}\right]$ | $\left[\frac{W}{10}, \frac{4W}{10}\right]$ | $\left[\frac{L}{10}, \frac{2L}{10}\right]$ | $\left[\frac{2W}{10}, \frac{7W}{10}\right]$ |
| 5 | $[1,5]$ | $\left[\frac{L}{10}, \frac{6L}{10}\right]$ | $\left[\frac{W}{10}, \frac{2W}{10}\right]$ | $\left[\frac{L}{10}, \frac{3L}{10}\right]$ | $\left[\frac{W}{10}, \frac{3W}{10}\right]$ | $\left[\frac{L}{10}, \frac{2L}{10}\right]$ | $\left[\frac{W}{10}, \frac{6W}{10}\right]$ |

Table 4.10: Item Properties for Different Problem Classes with $W = 20, L = 40$

### 4.2.2 Results on the Small Size Instances

Iori, Salazar-González, and Vigo (2006) presented an exact solution approach for instances with up to 40 customers for the 2|RO|L (rear oriented loading) case. Note that the distances between customers have been calculated using Euclidean distances rounded down to the next integer. Furthermore no single customer routes are allowed and all $K$ vehicles available at the depot have to be used. This additional requirements could be implemented in the ACO without great effort. The results are summarized in Table 4.11, which contains three different blocks for the branch-and-cut by Iori, Salazar-González, and Vigo (2006), the TS by Gendreau et al. (2008) and the ACO, respectively. For the branch-and-cut the average on five classes for the objective value (total routing cost,$z$), the average runtime until the best solution was found ($sec_h$), the average total runtime of the algorithm ($sec_{tot}$) and the total number of proven optima ($opt$) found are reported. For the TS two additional columns are inserted: $\%gap$ gives the average gap to $z$ from the branch-and-cut calculated by $\left(\frac{\overline{z}-z}{z}\right) \cdot 100$ and $imp$ the total number of improvements found by TS. Note that for the ACO for each instance and class ten independent runs have been performed due to the random element within the ACO. Therefore $\overline{z}$ is the average on five classes with ten runs each. $z_{min}$ ($z_{max}$) is the average on the five classes for the best (worst) solution of the ten independent runs. Analogously three gaps between the branch-and-cut and the ACO with respect to $z_{min}$, $\overline{z}$ and $z_{max}$ are reported. The branch-and-cut could find 58 optima out of $17 \times 5 = 85$ instances, the TS obtained 33 optima and could improve 19 of the suboptimal solutions of the branch-and-cut. The ACO on the one hand could find more proven optima (45) and on the other hand improved more suboptimal instances (25) than the TS. Concerning the runtime the TS and the ACO seem to be quite equivalent. The average runtime is less than two minutes for the TS and a bit more than 30 seconds for the ACO (on a computer that is twice as fast as the one for the TS). However the ACO is even better than the TS concerning only the worst solutions found by the ACO, yielding an average solution value of 749.3 against 767.4.

The first $9 \times 5 = 45$ instances are solved to optimality by the branch-and-cut. The TS hits only 26 of these optima, whereas the ACO hits 36 optima. ACO can even find 42 out of these 45 optima by setting the runtime of the loading routine to 100 seconds and 100000 number of backtracking steps. However this increased the average total runtime from 8.7 to 189.2 seconds.

| | | B&C[a] | | | | TS[b] | | | | | | ACO | | | | | | | %gap ACO - B&C | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | $n$ | $z$ | $sec_h$ | $sec_{tot}$ | *opt* | $z$ | $sec_h$ | $sec_{tot}$ | %*gap* | *opt* | *imp* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | *opt* | *imp* | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 15 | 281.0 | 25.4 | 29.9 | 5 | 281.4 | 0.8 | 7.6 | 0.14 | 4 | 0 | 285.0 | 286.4 | 287.0 | 3.2 | 9.5 | 3 | 0 | 1.42 | 1.91 | 2.13 |
| 2 | 15 | 336.6 | 13.9 | 14.7 | 5 | 337.2 | 0.4 | 3.9 | 0.17 | 4 | 0 | 337.2 | 337.7 | 339.0 | 0.3 | 0.8 | 4 | 0 | 0.17 | 0.31 | 0.70 |
| 3 | 20 | 375.4 | 18.6 | 25.4 | 5 | 377.6 | 5.1 | 18.0 | 0.93 | 4 | 0 | 375.4 | 376.3 | 377.2 | 3.0 | 6.2 | 5 | 0 | 0.00 | 0.24 | 0.47 |
| 4 | 20 | 430.0 | 11.6 | 17.0 | 5 | 433.8 | 1.5 | 14.0 | 0.87 | 3 | 0 | 431.0 | 431.0 | 431.2 | 2.2 | 3.9 | 4 | 0 | 0.23 | 0.24 | 0.28 |
| 5 | 21 | 377.2 | 597.7 | 597.8 | 5 | 387.0 | 3.8 | 28.8 | 2.59 | 2 | 0 | 377.2 | 378.1 | 379.2 | 9.5 | 19.0 | 5 | 0 | 0.00 | 0.23 | 0.53 |
| 6 | 21 | 490.4 | 62.4 | 67.4 | 5 | 494.6 | 2.8 | 23.1 | 0.86 | 3 | 0 | 490.6 | 492.3 | 494.8 | 2.8 | 5.5 | 4 | 0 | 0.04 | 0.39 | 0.90 |
| 7 | 22 | 687.2 | 674.2 | 676.3 | 5 | 699.8 | 12.7 | 39.2 | 1.76 | 1 | 0 | 689.0 | 690.6 | 693.6 | 3.4 | 9.9 | 3 | 0 | 0.25 | 0.48 | 0.90 |
| 8 | 22 | 705.8 | 256.4 | 261.6 | 5 | 717.4 | 10.8 | 45.0 | 1.62 | 2 | 0 | 708.6 | 710.8 | 712.8 | 8.2 | 15.2 | 4 | 0 | 0.40 | 0.70 | 0.99 |
| 9 | 25 | 614.2 | 304.8 | 469.9 | 5 | 616.6 | 5.4 | 38.2 | 0.39 | 3 | 0 | 614.4 | 617.0 | 625.6 | 6.5 | 8.2 | 4 | 0 | 0.03 | 0.45 | 1.86 |
| 10 | 29 | 675.8 | 22559.1 | 51161.0 | 3 | 684.0 | 40.7 | 150.8 | 1.58 | 0 | 1 | 664.0 | 669.1 | 672.8 | 28.3 | 43.9 | 1 | 1 | -1.48 | -0.71 | -0.17 |
| 11 | 29 | 725.6 | 42684.4 | 69120.4 | 1 | 718.4 | 110.7 | 175.7 | -0.88 | 1 | 2 | 688.0 | 692.2 | 696.8 | 98.8 | 110.2 | 1 | 4 | -4.59 | -4.08 | -3.52 |
| 12 | 30 | 609.4 | 39002.8 | 86400.4 | 0 | 612.0 | 48.3 | 87.3 | 0.41 | 0 | 2 | 600.6 | 602.8 | 607.4 | 8.2 | 10.5 | 0 | 4 | -1.42 | -1.06 | -0.31 |
| 13 | 32 | 2713.6 | 26502.5 | 58268.0 | 2 | 2588.4 | 95.9 | 296.2 | -3.27 | 1 | 2 | 2532.0 | 2542.2 | 2549.6 | 37.4 | 48.6 | 2 | 3 | -5.30 | -4.93 | -4.65 |
| 14 | 32 | 1233.4 | 50872.6 | 74228.8 | 1 | 1157.8 | 84.7 | 343.3 | -5.09 | 0 | 4 | 1143.8 | 1146.1 | 1148.8 | 73.6 | 102.2 | 0 | 4 | -6.12 | -5.93 | -5.72 |
| 15 | 32 | 1252.6 | 26353.5 | 69124.2 | 1 | 1191.6 | 220.7 | 308.0 | -4.50 | 1 | 3 | 1178.4 | 1182.8 | 1185.2 | 133.2 | 191.9 | 1 | 4 | -5.53 | -5.21 | -5.04 |
| 16 | 35 | 683.8 | 8582.7 | 10098.1 | 5 | 686.4 | 35.3 | 161.8 | 0.38 | 4 | 0 | 684.2 | 685.3 | 687.2 | 6.8 | 9.7 | 4 | 0 | 0.06 | 0.22 | 0.50 |
| 17 | 40 | 854.6 | 359.9 | 86400.3 | 0 | 844.4 | 46.4 | 234.7 | -1.19 | 0 | 5 | 843.8 | 845.7 | 849.6 | 3.3 | 9.3 | 0 | 5 | -1.26 | -1.03 | -0.58 |
| AVG | | 767.4 | 12875.4 | 29821.2 | 58 | 754.6 | 42.7 | 116.2 | -0.21 | 33 | 19 | 743.7 | 746.2 | 749.3 | 25.2 | 35.6 | 45 | 25 | -1.36 | -1.05 | -0.63 |

[a] see Iori, Salazar-González, and Vigo (2006), Pentium IV, 1.7 GHz, B&C = branch-and-cut
[b] see Gendreau et al. (2008), Pentium IV, 1.7 GHz

Table 4.11: Integer costs, No One-Customer Routes and Fixed Number of Vehicles for 2|RO|L

### 4.2.3 Results on the Complete Set of Instances

For the complete set of 180 instances again three different approaches have to be compared: The presented ACO, a TS and a Guided Tabu Search (GTS). For calculating the routing costs the non integer Euclidean distances have been used. Moreover single customer routes are allowed and not all vehicles $K$ have to be used.

Table 4.12 represents a comparison aggregated on the 36 different classes per line for 2|RO|L. For all three metaheuristics the average objective values and average run times are reported. The gaps are not only computed with respect to the average solution quality obtained by the ACO, but also with respect to the best and worst solutions found. ACO outperforms the other two approaches, only the GTS can succeed in class 1 with respect to the average gap. Apart from that it is interesting that the largest average gap between ACO and TS arises in class 2 ($-5.56\%$), whereas between ACO and GTS in class 3 ($-3.58\%$).

Table 4.13 shows the comparison as average on five classes per line for 2|RO|L. Again the ACO outperforms both other metaheuristics (compared to the average solution values found ACO is 3.81% better than TS and 2.28% better than GTS), even with respect to the worst solutions found ($-2.92\%$ against TS and $-1.37\%$ against GTS). Concerning the instance average with respect to $z_{min}$ ACO is never worse than TS and GTS, for $\overline{z}$ ACO is worse for one instance compared to TS and always better compared to GTS and even for $z_{max}$ ACO is only beaten four times by TS and six times by GTS.

Table 4.14 shows the same information for the 2|UO|L (unrestricted oriented) case. Here the gaps between ACO and TS with respect to $z_{min}$, $\overline{z}$ and $z_{max}$ are $-4.07\%$, $-3.61\%$ and $-3.02\%$, the GTS is again better than TS but cannot compete with the ACO yielding the following gaps: $-2.34\%$ ($z_{min}$), $-1.87\%(\overline{z})$ and $-1.26\%(z_{max})$.

| | TS[a] | | | GTS[b] | ACO | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | $\%\overline{gap}$ [c] | $\%\overline{gap}$ [d] |
| 1 | 792.31 | 772.9 | 1757.4 | 777.75 | 775.806861 | 782.7928 | 792.340056 | 238.7 | 270.7 | -1.02 | 0.55 |
| 2 | 1290.19 | 867.5 | 1670.4 | 1246.77 | 1187.60303 | 1198.97373 | 1213.09236 | 3025.5 | 3098.7 | -5.56 | -3.40 |
| 3 | 1273.77 | 902.8 | 1789.4 | 1266.06 | 1199.60417 | 1210.04534 | 1222.66519 | 2881.6 | 2949.1 | -3.73 | -3.58 |
| 4 | 1339.18 | 1021.5 | 1836.3 | 1294.51 | 1230.19119 | 1246.05426 | 1262.04894 | 3071.3 | 3167.6 | -5.04 | -3.12 |
| 5 | 1163.28 | 1117.6 | 2031.5 | 1126.05 | 1091.17819 | 1101.74776 | 1113.33311 | 2995.8 | 3078.7 | -3.69 | -1.85 |
| AVG | 1171.75 | 936.5 | 1817.0 | 1142.23 | 1096.87669 | 1107.92278 | 1120.69593 | 2442.6 | 2513.0 | -3.81 | -2.28 |

[a] see Gendreau et al. (2008), Pentium IV, 1.7 GHz
[b] see Zachariadis, Tarantilis, and Kiranoudis (2007), Pentium IV, 2.4 GHz

Table 4.12: Comparison of Three Different Metaheuristics for 2|RO|L (Averages on 36 Instances per Line)

| | | TS[a] | | | GTS[b] | | | ACO | | | | | %*gap* ACO - TS | | | %*gap* ACO - GTS | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | $n$ | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $sec_h$ | $sec_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 15 | 295.01 | 2.6 | 9.2 | 299.12 | 2.9 | 4.6 | 291.33 | 291.73 | 293.46 | 4.9 | 6.3 | -1.19 | -1.06 | -0.49 | -2.45 | -2.32 | -1.77 |
| 2 | 15 | 343.18 | 0.4 | 3.5 | 344.36 | 1.6 | 2.6 | 343.18 | 343.51 | 344.97 | 0.2 | 0.6 | 0.00 | 0.09 | 0.51 | -0.35 | -0.25 | 0.16 |
| 3 | 20 | 380.19 | 3.8 | 18.9 | 386.36 | 2.6 | 4.6 | 376.80 | 378.20 | 379.50 | 1.8 | 2.7 | -0.87 | -0.51 | -0.17 | -2.39 | -2.04 | -1.71 |
| 4 | 20 | 440.91 | 1.4 | 17.0 | 441.87 | 2.4 | 3.9 | 439.07 | 439.35 | 441.94 | 2.1 | 2.8 | -0.41 | -0.34 | 0.26 | -0.62 | -0.55 | 0.05 |
| 5 | 21 | 382.30 | 4.1 | 27.6 | 392.15 | 5.6 | 9.5 | 380.97 | 381.76 | 382.97 | 10.5 | 13.9 | -0.34 | -0.14 | 0.17 | -2.78 | -2.58 | -2.28 |
| 6 | 21 | 501.40 | 5.1 | 19.5 | 503.21 | 2.9 | 4.7 | 498.75 | 499.44 | 500.89 | 3.2 | 4.2 | -0.52 | -0.38 | -0.09 | -0.87 | -0.73 | -0.45 |
| 7 | 22 | 691.23 | 15.5 | 53.0 | 692.78 | 14.6 | 25.8 | 675.53 | 678.66 | 683.28 | 8.4 | 10.2 | -2.17 | -1.73 | -1.07 | -2.38 | -1.93 | -1.28 |
| 8 | 22 | 691.89 | 32.8 | 83.7 | 686.29 | 17.4 | 26.5 | 681.03 | 683.23 | 685.55 | 9.4 | 17.1 | -1.49 | -1.19 | -0.86 | -0.74 | -0.44 | -0.10 |
| 9 | 25 | 620.77 | 10.6 | 40.0 | 619.29 | 8.6 | 14.2 | 613.16 | 615.78 | 620.31 | 3.6 | 4.6 | -1.20 | -0.79 | -0.07 | -0.95 | -0.53 | 0.18 |
| 10 | 29 | 679.68 | 43.5 | 179.6 | 689.45 | 23.5 | 41.3 | 665.80 | 671.24 | 676.51 | 40.9 | 43.8 | -2.03 | -1.25 | -0.50 | -3.32 | -2.55 | -1.81 |
| 11 | 29 | 719.76 | 99.0 | 199.4 | 715.52 | 44.0 | 77.1 | 689.62 | 693.87 | 698.92 | 41.9 | 44.4 | -3.57 | -3.05 | -2.44 | -3.28 | -2.75 | -2.13 |
| 12 | 30 | 627.59 | 58.8 | 99.5 | 624.89 | 66.4 | 106.8 | 614.62 | 615.76 | 618.32 | 4.9 | 6.1 | -1.98 | -1.79 | -1.37 | -1.58 | -1.40 | -0.98 |
| 13 | 32 | 2550.89 | 49.0 | 312.8 | 2544.74 | 40.5 | 72.2 | 2475.03 | 2483.61 | 2498.50 | 47.1 | 50.9 | -2.78 | -2.46 | -1.90 | -2.58 | -2.25 | -1.70 |
| 14 | 32 | 1048.72 | 146.0 | 439.5 | 1041.76 | 86.7 | 143.8 | 996.18 | 1005.40 | 1011.85 | 125.1 | 130.1 | -4.73 | -3.92 | -3.32 | -4.16 | -3.35 | -2.74 |
| 15 | 32 | 1160.25 | 165.4 | 313.4 | 1154.90 | 34.6 | 61.9 | 1137.00 | 1145.68 | 1151.87 | 115.1 | 119.4 | -1.87 | -1.19 | -0.71 | -1.38 | -0.69 | -0.21 |
| 16 | 35 | 703.60 | 28.0 | 157.2 | 705.77 | 33.4 | 50.9 | 700.74 | 701.43 | 703.43 | 6.3 | 8.1 | -0.40 | -0.30 | -0.01 | -0.71 | -0.61 | -0.33 |
| 17 | 40 | 865.72 | 88.9 | 226.2 | 864.82 | 49.9 | 84.9 | 863.77 | 864.30 | 865.97 | 3.9 | 6.9 | -0.23 | -0.16 | 0.03 | -0.12 | -0.06 | 0.13 |
| 18 | 44 | 1037.65 | 566.5 | 1167.8 | 1027.97 | 214.0 | 328.0 | 1000.12 | 1004.78 | 1008.54 | 236.9 | 243.6 | -3.39 | -2.94 | -2.58 | -2.62 | -2.17 | -1.80 |
| 19 | 50 | 746.91 | 365.2 | 1521.5 | 745.67 | 292.8 | 466.8 | 724.47 | 729.03 | 736.21 | 101.1 | 105.4 | -2.73 | -2.13 | -1.16 | -2.59 | -1.99 | -1.01 |
| 20 | 71 | 513.84 | 808.9 | 3370.3 | 510.17 | 440.8 | 747.0 | 480.42 | 484.31 | 487.87 | 834.6 | 844.9 | -5.60 | -4.93 | -4.30 | -5.26 | -4.58 | -3.96 |
| 21 | 75 | 1025.79 | 1702.2 | 3561.2 | 1022.58 | 759.3 | 1303.7 | 977.72 | 989.01 | 1003.05 | 870.9 | 899.5 | -4.32 | -3.29 | -2.01 | -3.87 | -2.83 | -1.53 |
| 22 | 75 | 1052.39 | 1573.8 | 3461.8 | 1051.02 | 1184.3 | 1998.8 | 1009.81 | 1017.90 | 1029.82 | 595.5 | 605.7 | -3.71 | -2.96 | -1.83 | -3.64 | -2.89 | -1.76 |
| 23 | 75 | 1121.18 | 675.8 | 3600.0 | 1088.81 | 919.8 | 1461.3 | 1035.51 | 1052.62 | 1066.08 | 946.7 | 958.5 | -7.06 | -5.59 | -4.43 | -4.49 | -2.98 | -1.78 |
| 24 | 75 | 1208.52 | 2642.5 | 3324.6 | 1172.36 | 1303.7 | 2096.4 | 1129.37 | 1137.84 | 1148.85 | 252.6 | 260.6 | -6.10 | -5.39 | -4.48 | -3.47 | -2.73 | -1.81 |
| 25 | 100 | 1350.56 | 2336.5 | 3600.1 | 1349.11 | 1988.7 | 3389.2 | 1299.85 | 1310.59 | 1320.55 | 2060.9 | 2088.3 | -3.46 | -2.70 | -1.99 | -3.27 | -2.51 | -1.79 |
| 26 | 100 | 1341.30 | 1554.6 | 3600.3 | 1344.68 | 1115.5 | 1936.5 | 1283.23 | 1295.84 | 1306.91 | 2932.0 | 3325.9 | -4.00 | -3.18 | -2.44 | -4.02 | -3.21 | -2.48 |
| 27 | 100 | 1439.37 | 1308.2 | 3600.0 | 1390.20 | 818.6 | 1292.3 | 1332.14 | 1343.27 | 1352.86 | 1331.1 | 1354.9 | -6.94 | -6.18 | -5.50 | -3.93 | -3.14 | -2.45 |
| 28 | 120 | 2502.48 | 2576.9 | 3600.1 | 2476.66 | 1541.4 | 2622.8 | 2354.81 | 2397.27 | 2440.45 | 8060.9 | 8570.5 | -5.52 | -3.35 | -1.07 | -4.20 | -1.99 | 0.34 |
| 29 | 134 | 2296.03 | 1162.5 | 3600.2 | 2206.23 | 1257.7 | 2179.6 | 2068.00 | 2109.07 | 2151.45 | 8507.7 | 8830.4 | -8.82 | -6.85 | -4.69 | -5.70 | -3.67 | -1.47 |
| 30 | 150 | 1873.27 | 2021.4 | 3600.2 | 1832.96 | 1229.5 | 1980.6 | 1710.14 | 1729.28 | 1761.11 | 8325.7 | 8689.7 | -8.13 | -7.13 | -5.47 | -5.67 | -4.66 | -2.97 |
| 31 | 199 | 2366.54 | 2102.2 | 3600.5 | 2327.74 | 1681.6 | 2748.7 | 2181.49 | 2206.89 | 2241.84 | 8594.7 | 8752.3 | -7.90 | -6.76 | -5.16 | -6.18 | -5.02 | -3.38 |
| 32 | 199 | 2354.60 | 2305.2 | 3600.6 | 2235.69 | 2528.1 | 4313.5 | 2136.35 | 2174.12 | 2216.18 | 8547.3 | 8745.6 | -8.50 | -6.87 | -4.96 | -3.97 | -2.25 | -0.25 |
| 33 | 199 | 2360.74 | 2221.2 | 3600.6 | 2317.97 | 2367.0 | 4104.9 | 2202.20 | 2230.82 | 2266.92 | 8604.4 | 8743.9 | -6.25 | -4.99 | -3.31 | -4.36 | -3.08 | -1.36 |
| 34 | 240 | 1408.64 | 2184.4 | 3601.0 | 1186.77 | 3674.1 | 4596.2 | 1132.31 | 1145.83 | 1164.32 | 8690.2 | 8769.8 | -19.37 | -18.47 | -17.16 | -4.22 | -3.15 | -1.61 |
| 35 | 252 | 1786.93 | 2223.1 | 3600.2 | 1515.69 | 3291.8 | 4313.2 | 1415.36 | 1442.91 | 1473.72 | 8927.1 | 8947.2 | -18.34 | -16.98 | -15.51 | -5.66 | -4.02 | -2.23 |
| 36 | 255 | 1693.10 | 2626.3 | 3600.9 | 1610.61 | 2825.2 | 4732.2 | 1571.70 | 1590.92 | 1610.11 | 9083.5 | 9258.3 | -7.37 | -6.23 | -5.08 | -1.47 | -0.26 | 0.97 |
| AVG | | 1171.75 | 936.5 | 1817.0 | 1142.23 | 829.8 | 1315.2 | 1096.88 | 1107.92 | 1120.70 | 2442.6 | 2513.0 | -4.54 | -3.81 | -2.92 | -3.03 | -2.28 | -1.37 |

[a] see Gendreau et al. (2008), Pentium IV, 1.7 GHz
[b] see Zachariadis, Tarantilis, and Kiranoudis (2007), Pentium IV, 2.4 GHz

Table 4.13: Comparison of Three Different Metaheuristics for 2|RO|L (Average on Five Classes per Instance)

| | | TS[a] | | | GTS[b] | | | ACO | | | | | %gap ACO - TS | | | %gap ACO - GTS | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | $n$ | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $sec_h$ | $sec_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 15 | 289.03 | 4.2 | 9.7 | 292.34 | 2.3 | 3.8 | 284.36 | 284.68 | 285.50 | 2.6 | 3.3 | -1.55 | -1.44 | -1.16 | -2.65 | -2.54 | -2.27 |
| 2 | 15 | 339.81 | 0.1 | 3.7 | 340.51 | 1.3 | 2.1 | 339.81 | 339.81 | 339.81 | 0.2 | 0.5 | 0.00 | 0.00 | 0.00 | -0.20 | -0.20 | -0.20 |
| 3 | 20 | 373.83 | 1.6 | 19.6 | 379.27 | 1.3 | 2.3 | 372.74 | 372.74 | 372.74 | 0.8 | 1.6 | -0.30 | -0.30 | -0.30 | -1.63 | -1.63 | -1.63 |
| 4 | 20 | 436.14 | 0.6 | 14.7 | 439.34 | 2.0 | 3.4 | 437.10 | 437.10 | 437.10 | 1.3 | 1.9 | 0.22 | 0.22 | 0.22 | -0.51 | -0.51 | -0.51 |
| 5 | 21 | 379.21 | 5.0 | 25.2 | 380.83 | 4.0 | 6.4 | 378.28 | 378.35 | 379.01 | 5.6 | 8.2 | -0.24 | -0.23 | -0.05 | -0.66 | -0.64 | -0.47 |
| 6 | 21 | 499.98 | 7.2 | 18.8 | 498.74 | 3.9 | 6.3 | 496.80 | 496.96 | 497.05 | 1.9 | 2.9 | -0.63 | -0.60 | -0.58 | -0.39 | -0.35 | -0.34 |
| 7 | 22 | 673.99 | 6.3 | 48.4 | 676.51 | 3.8 | 5.7 | 671.24 | 671.60 | 672.14 | 3.4 | 5.8 | -0.41 | -0.35 | -0.27 | -0.73 | -0.67 | -0.60 |
| 8 | 22 | 669.70 | 11.2 | 61.3 | 678.19 | 5.2 | 8.3 | 666.62 | 668.62 | 671.16 | 7.3 | 11.0 | -0.45 | -0.14 | 0.24 | -1.67 | -1.37 | -0.98 |
| 9 | 25 | 617.28 | 3.6 | 41.2 | 614.05 | 4.3 | 6.5 | 611.14 | 611.82 | 615.32 | 2.0 | 3.1 | -0.96 | -0.85 | -0.28 | -0.47 | -0.36 | 0.21 |
| 10 | 29 | 668.07 | 36.0 | 192.0 | 677.56 | 8.8 | 15.2 | 653.51 | 659.43 | 664.25 | 20.1 | 23.4 | -2.06 | -1.24 | -0.55 | -3.27 | -2.47 | -1.79 |
| 11 | 29 | 692.24 | 55.7 | 207.6 | 693.33 | 15.2 | 26.4 | 678.19 | 678.52 | 678.62 | 14.1 | 16.7 | -1.81 | -1.77 | -1.76 | -2.01 | -1.97 | -1.96 |
| 12 | 30 | 618.61 | 49.0 | 82.7 | 615.46 | 50.8 | 76.9 | 611.81 | 613.14 | 617.39 | 2.8 | 4.2 | -1.08 | -0.87 | -0.18 | -0.58 | -0.36 | 0.32 |
| 13 | 32 | 2479.83 | 57.5 | 332.1 | 2480.79 | 36.7 | 56.8 | 2421.22 | 2425.06 | 2427.44 | 25.4 | 30.7 | -2.25 | -2.10 | -2.00 | -2.25 | -2.09 | -1.99 |
| 14 | 32 | 1005.71 | 375.9 | 565.6 | 996.95 | 137.3 | 244.6 | 963.27 | 966.87 | 970.78 | 59.5 | 63.3 | -4.04 | -3.70 | -3.35 | -3.13 | -2.79 | -2.43 |
| 15 | 32 | 1128.64 | 156.7 | 375.9 | 1125.80 | 70.5 | 113.2 | 1101.08 | 1107.88 | 1114.62 | 46.5 | 49.4 | -2.22 | -1.61 | -1.02 | -2.16 | -1.57 | -0.99 |
| 16 | 35 | 701.35 | 20.5 | 160.2 | 701.56 | 65.2 | 110.4 | 699.55 | 700.47 | 701.57 | 5.0 | 6.6 | -0.25 | -0.12 | 0.03 | -0.28 | -0.15 | 0.00 |
| 17 | 40 | 865.62 | 64.9 | 218.2 | 864.07 | 27.7 | 48.0 | 863.77 | 864.63 | 866.23 | 3.9 | 6.9 | -0.21 | -0.11 | 0.07 | -0.03 | 0.07 | 0.25 |
| 18 | 44 | 1013.38 | 589.3 | 1318.0 | 1007.62 | 210.0 | 333.6 | 979.08 | 982.07 | 985.94 | 111.2 | 115.9 | -3.14 | -2.85 | -2.49 | -2.68 | -2.39 | -2.02 |
| 19 | 50 | 722.72 | 633.7 | 1524.5 | 726.45 | 321.2 | 490.4 | 706.38 | 709.33 | 713.43 | 46.2 | 49.9 | -2.10 | -1.69 | -1.08 | -2.57 | -2.16 | -1.55 |
| 20 | 71 | 500.13 | 954.5 | 3237.3 | 490.28 | 454.6 | 739.6 | 470.53 | 472.66 | 475.08 | 263.0 | 274.7 | -5.16 | -4.77 | -4.33 | -3.69 | -3.29 | -2.84 |
| 21 | 75 | 990.60 | 460.1 | 3600.0 | 977.87 | 147.5 | 234.9 | 948.79 | 953.67 | 960.56 | 312.5 | 326.1 | -3.91 | -3.45 | -2.76 | -2.66 | -2.20 | -1.50 |
| 22 | 75 | 1018.10 | 1191.2 | 3544.5 | 1009.02 | 885.5 | 1499.9 | 983.80 | 989.29 | 995.65 | 232.1 | 253.9 | -3.16 | -2.62 | -1.98 | -2.32 | -1.77 | -1.13 |
| 23 | 75 | 1055.87 | 2032.4 | 3538.6 | 1040.75 | 1106.9 | 1909.1 | 1010.42 | 1017.90 | 1025.74 | 329.9 | 346.3 | -4.16 | -3.45 | -2.70 | -2.68 | -1.97 | -1.21 |
| 24 | 75 | 1159.88 | 1454.2 | 3256.1 | 1126.36 | 645.3 | 1010.1 | 1103.52 | 1111.42 | 1119.26 | 112.6 | 123.7 | -4.70 | -4.01 | -3.32 | -1.96 | -1.25 | -0.54 |
| 25 | 100 | 1315.46 | 1205.8 | 3600.0 | 1300.99 | 699.5 | 1115.5 | 1259.44 | 1267.04 | 1274.91 | 652.4 | 685.7 | -3.90 | -3.36 | -2.78 | -2.84 | -2.29 | -1.71 |
| 26 | 100 | 1287.50 | 1173.9 | 3600.1 | 1280.42 | 906.6 | 1471.7 | 1240.91 | 1245.51 | 1250.88 | 655.9 | 699.6 | -3.36 | -3.03 | -2.65 | -2.80 | -2.47 | -2.09 |
| 27 | 100 | 1380.13 | 521.3 | 3600.0 | 1336.81 | 568.2 | 1011.9 | 1295.56 | 1305.57 | 1314.26 | 552.8 | 581.4 | -5.84 | -5.14 | -4.51 | -2.92 | -2.19 | -1.54 |
| 28 | 120 | 2406.30 | 2051.2 | 3600.2 | 2398.03 | 1019.6 | 1745.8 | 2285.38 | 2321.62 | 2357.59 | 6273.7 | 6919.3 | -4.78 | -2.77 | -0.69 | -3.98 | -1.93 | 0.18 |
| 29 | 134 | 2215.27 | 1406.5 | 3600.1 | 2089.79 | 1175.6 | 1880.8 | 2017.45 | 2036.03 | 2067.77 | 7754.1 | 8461.5 | -8.03 | -6.90 | -5.11 | -3.22 | -2.08 | -0.24 |
| 30 | 150 | 1771.29 | 1185.4 | 3600.3 | 1719.28 | 1543.6 | 2560.5 | 1639.68 | 1652.02 | 1666.55 | 6498.0 | 7070.5 | -6.91 | -6.19 | -5.32 | -4.13 | -3.39 | -2.50 |
| 31 | 199 | 2298.40 | 2375.8 | 3600.2 | 2215.00 | 2126.8 | 3498.5 | 2104.98 | 2120.11 | 2142.34 | 7830.9 | 8275.9 | -8.40 | -7.64 | -6.48 | -4.98 | -4.19 | -3.00 |
| 32 | 199 | 2259.67 | 1664.8 | 3600.4 | 2147.09 | 1973.2 | 3183.9 | 2053.43 | 2074.71 | 2101.79 | 8413.4 | 8652.1 | -8.48 | -7.44 | -6.06 | -3.90 | -2.82 | -1.38 |
| 33 | 199 | 2275.50 | 1843.2 | 3600.2 | 2230.27 | 2001.0 | 3270.6 | 2113.92 | 2129.80 | 2152.09 | 7782.8 | 8304.0 | -6.72 | -5.92 | -4.75 | -4.65 | -3.83 | -2.63 |
| 34 | 240 | 1344.63 | 1359.1 | 3601.3 | 1130.75 | 2206.6 | 3373.7 | 1089.30 | 1095.96 | 1104.38 | 8256.4 | 8562.6 | -18.88 | -18.40 | -17.71 | -3.40 | -2.83 | -2.01 |
| 35 | 252 | 1610.81 | 2061.7 | 3600.4 | 1374.43 | 2553.0 | 3656.9 | 1318.95 | 1329.93 | 1343.72 | 8711.6 | 8941.8 | -17.07 | -16.34 | -15.51 | -3.74 | -2.90 | -1.92 |
| 36 | 255 | 1663.33 | 2265.8 | 3600.1 | 1559.95 | 2442.8 | 3663.3 | 1501.97 | 1515.52 | 1534.15 | 9126.9 | 9252.1 | -9.55 | -8.67 | -7.52 | -2.62 | -1.66 | -0.42 |
| AVG | | 1131.33 | 757.9 | 1822.2 | 1100.46 | 650.8 | 1038.5 | 1065.94 | 1072.44 | 1080.47 | 2058.8 | 2170.4 | -4.07 | -3.61 | -3.02 | -2.34 | -1.87 | -1.26 |

[a] see Gendreau et al. (2008), Pentium IV, 1.7 GHz
[b] see Zachariadis, Tarantilis, and Kiranoudis (2007), Pentium IV, 2.4 GHz

Table 4.14: Comparison of Three Different Metaheuristics for 2|UO|L (Average on Five Classes per Instance)

Another interesting comparison is shown in Table 4.15, where the effects of the different loading configurations are evaluated. Note that here the averages per line are based on classes 2 to 5. Class 1 is not taken into account because it is a classical CVRP instance, where the loading concerning the area of the items has no effect on the solution. From this table the following managerial conclusions can be drawn: As expected the largest improvement is possible between 2|RO|L (the tightest loading configuration) and 2|UN|L (the loosest loading configuration), which results in saving nearly 5% in routing cost. 2|UO|L and 2|RN|L are indeed equivalent. Disregarding class one and comparing $\overline{z}$ 2|RN|L is better in 65 cases, whereas 2|UO|L is better in 64 cases and in 15 both loading configurations yield equal average objective values. Allowing items to be rotated and keeping the rear loading constraint yields a saving of more than 3% of total routing costs. In this context it is also interesting to investigate the number of vehicles in use. Adding up the number of vehicles used for all instances, classes and the ten runs performed for each instance and class 27 051, 26 393, 26 430, 25 740 vehicles were needed for 2|RO|L, 2|UO|L, 2|RN|L and 2|UN|L respectively. Table 4.15 does not give any information about run times but the aggregate values can be seen in Table 4.17. The biggest difference is again between 2|RO|L and 2|UN|L where the average total runtime $sec_{tot}$ (average total runtime to find the best solution $sec_h$) reduces from 2513.0 (2442.6) to 1631.4 (1534.0) seconds. The difference between 2|UO|L and 2|RN|L is only little, yielding 2170.4 (2058.8) and 2264.5 (2156.6) seconds respectively.

Nevertheless the comparison between the three approaches so far suffers from different premises and is therefore favoring the ACO approach. ACO has been granted the longest runtime limit on the fastest machine. However Table 4.16 gives clear evidence that ACO outperforms the other two approaches also for shorter run times. Even when the ACO is only given half an hour runtime it outperforms TS and GTS, achieving an $\%\overline{gap}$ of $-2.84$ and $-1.26$ for 2|RO|L and $-3.22$ and $-1.45$ for 2|UO|L. Clearly the gap increases when the runtime limit increases.

We have also performed a sensitivity analysis w. r. t. the parameter $\alpha_3$ measuring the weight of the virtual rectangle in (3.11); see Section 3.4.2. In Fuellerer et al. (2007) this parameter was set to $\alpha_3 = 1$. Since this term is essentially a part of the visibility, here we chose $\alpha_3 = \alpha_2 = 5$. Table 4.17 shows a comparison between the results for $\alpha_3 = 1$ and $\alpha_3 = 5$. Both versions have been run on identical machines and compiled with identical compilers, therefore also the run time can be compared directly. The solution quality obtained is almost equivalent for 2|UO|L, 2|RN|L and 2|UN|L, only for the 2|RO|L the presented version is a little bit better. Furthermore the run time with the new version has decreased slightly for all four different loading configurations.

| | 2\|RO\|L | | | 2\|UO\|L | | | | 2\|RN\|L | | | | 2\|UN\|L | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\%\overline{gap}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\%\overline{gap}$ |
| 1 | 294.48 | 294.98 | 297.15 | 285.77 | 286.17 | 287.19 | -2.98 | 281.70 | 282.45 | 283.03 | -4.20 | 281.16 | 281.53 | 283.15 | -4.52 |
| 2 | 345.23 | 345.65 | 347.47 | 341.02 | 341.02 | 341.02 | -1.32 | 341.02 | 341.02 | 341.02 | -1.32 | 341.02 | 341.02 | 341.02 | -1.32 |
| 3 | 381.40 | 383.15 | 384.78 | 376.32 | 376.32 | 376.32 | -1.73 | 376.65 | 376.68 | 376.74 | -1.62 | 372.93 | 374.57 | 375.12 | -2.18 |
| 4 | 441.11 | 441.47 | 444.70 | 438.66 | 438.66 | 438.66 | -0.64 | 435.01 | 437.08 | 438.66 | -0.99 | 435.01 | 435.01 | 435.01 | -1.46 |
| 5 | 382.39 | 383.38 | 384.90 | 379.03 | 379.12 | 379.95 | -1.10 | 378.59 | 379.85 | 381.78 | -0.91 | 378.59 | 378.87 | 379.51 | -1.16 |
| 6 | 499.48 | 500.33 | 502.15 | 497.04 | 497.24 | 497.36 | -0.61 | 497.62 | 497.78 | 499.22 | -0.51 | 497.04 | 497.05 | 497.13 | -0.65 |
| 7 | 702.27 | 706.19 | 711.96 | 696.91 | 697.36 | 698.03 | -1.25 | 696.23 | 697.03 | 698.46 | -1.28 | 687.37 | 688.38 | 688.50 | -2.49 |
| 8 | 709.15 | 711.89 | 714.79 | 691.14 | 693.63 | 696.81 | -2.57 | 690.84 | 693.79 | 695.38 | -2.49 | 678.75 | 679.48 | 682.06 | -4.61 |
| 9 | 614.54 | 617.81 | 623.47 | 612.01 | 612.86 | 617.24 | -0.80 | 612.02 | 613.28 | 615.17 | -0.73 | 612.02 | 612.62 | 614.09 | -0.84 |
| 10 | 698.30 | 705.10 | 711.69 | 682.93 | 690.34 | 696.37 | -2.05 | 685.26 | 689.21 | 694.76 | -2.25 | 672.46 | 674.09 | 676.38 | -4.29 |
| 11 | 735.77 | 741.09 | 747.40 | 721.49 | 721.90 | 722.02 | -2.63 | 712.59 | 718.51 | 726.61 | -3.03 | 700.06 | 700.19 | 700.38 | -5.49 |
| 12 | 615.77 | 616.89 | 619.34 | 612.26 | 613.63 | 618.18 | -0.52 | 613.27 | 617.46 | 619.93 | 0.09 | 613.18 | 614.82 | 618.43 | -0.33 |
| 13 | 2592.20 | 2602.93 | 2621.54 | 2524.94 | 2529.74 | 2532.72 | -2.80 | 2514.52 | 2523.62 | 2542.25 | -3.03 | 2468.31 | 2468.54 | 2470.36 | -5.15 |
| 14 | 1035.80 | 1047.33 | 1055.39 | 994.67 | 999.17 | 1004.06 | -4.44 | 995.00 | 1000.95 | 1009.04 | -4.24 | 980.14 | 982.90 | 984.57 | -5.94 |
| 15 | 1211.83 | 1222.69 | 1230.42 | 1166.93 | 1175.43 | 1183.86 | -3.92 | 1164.26 | 1171.26 | 1180.02 | -4.33 | 1134.97 | 1143.37 | 1150.44 | -6.53 |
| 16 | 701.27 | 702.13 | 704.64 | 699.79 | 700.94 | 702.31 | -0.17 | 699.79 | 700.19 | 703.75 | -0.28 | 699.79 | 700.38 | 702.30 | -0.25 |
| 17 | 864.26 | 864.78 | 866.81 | 864.26 | 865.20 | 867.13 | 0.05 | 862.36 | 862.85 | 863.28 | -0.22 | 862.57 | 862.57 | 862.57 | -0.25 |
| 18 | 1069.26 | 1074.43 | 1078.74 | 1042.97 | 1046.04 | 1050.48 | -2.64 | 1019.14 | 1024.05 | 1030.71 | -4.65 | 1012.19 | 1013.52 | 1017.11 | -5.59 |
| 19 | 774.43 | 779.47 | 787.01 | 751.82 | 754.84 | 758.54 | -3.17 | 740.88 | 748.37 | 754.01 | -3.90 | 728.11 | 732.15 | 734.97 | -6.00 |
| 20 | 540.03 | 544.84 | 549.23 | 527.66 | 530.29 | 533.24 | -2.67 | 519.33 | 522.16 | 526.53 | -4.09 | 508.60 | 509.89 | 510.72 | -6.36 |
| 21 | 1049.60 | 1063.28 | 1080.08 | 1013.44 | 1019.10 | 1026.97 | -4.16 | 1013.18 | 1021.93 | 1029.50 | -3.81 | 989.45 | 995.75 | 1001.24 | -6.28 |
| 22 | 1076.53 | 1085.76 | 1099.15 | 1044.03 | 1050.00 | 1056.43 | -3.27 | 1030.27 | 1041.16 | 1049.43 | -4.06 | 1009.12 | 1014.11 | 1018.06 | -6.56 |
| 23 | 1083.05 | 1102.51 | 1117.22 | 1051.68 | 1059.10 | 1066.80 | -3.90 | 1046.44 | 1054.36 | 1064.46 | -4.30 | 1024.69 | 1033.55 | 1040.90 | -6.18 |
| 24 | 1154.15 | 1161.74 | 1172.37 | 1121.84 | 1128.71 | 1135.38 | -2.80 | 1122.22 | 1129.03 | 1137.98 | -2.75 | 1103.92 | 1109.86 | 1117.95 | -4.40 |
| 25 | 1417.11 | 1429.76 | 1441.32 | 1366.60 | 1375.33 | 1384.28 | -3.84 | 1355.16 | 1367.28 | 1380.11 | -4.30 | 1332.83 | 1339.09 | 1347.16 | -6.31 |
| 26 | 1399.15 | 1414.91 | 1428.74 | 1346.24 | 1352.00 | 1358.71 | -4.30 | 1339.37 | 1348.63 | 1359.37 | -4.56 | 1313.49 | 1318.35 | 1323.44 | -6.68 |
| 27 | 1390.12 | 1402.99 | 1413.07 | 1344.39 | 1355.86 | 1364.82 | -3.37 | 1341.72 | 1354.14 | 1364.61 | -3.47 | 1310.03 | 1318.04 | 1330.89 | -6.05 |
| 28 | 2680.89 | 2719.14 | 2756.36 | 2594.10 | 2624.58 | 2652.78 | -3.51 | 2595.05 | 2616.06 | 2644.48 | -3.78 | 2518.80 | 2544.62 | 2582.83 | -6.43 |
| 29 | 2291.64 | 2332.63 | 2371.48 | 2228.45 | 2241.34 | 2266.88 | -3.91 | 2210.79 | 2227.91 | 2252.18 | -4.48 | 2174.24 | 2185.15 | 2201.27 | -6.32 |
| 30 | 1878.24 | 1899.64 | 1935.79 | 1790.17 | 1803.07 | 1817.60 | -5.05 | 1796.86 | 1810.48 | 1825.86 | -4.59 | 1750.29 | 1757.63 | 1767.34 | -7.40 |
| 31 | 2391.39 | 2416.55 | 2449.35 | 2295.75 | 2308.07 | 2324.98 | -4.48 | 2288.63 | 2304.93 | 2325.09 | -4.57 | 2230.18 | 2242.57 | 2262.75 | -7.17 |
| 32 | 2336.87 | 2377.33 | 2419.14 | 2233.22 | 2253.07 | 2276.15 | -5.19 | 2242.05 | 2260.92 | 2287.10 | -4.82 | 2180.30 | 2194.44 | 2208.60 | -7.64 |
| 33 | 2419.83 | 2449.49 | 2484.43 | 2309.48 | 2323.22 | 2340.89 | -5.15 | 2306.91 | 2322.76 | 2342.77 | -5.11 | 2238.96 | 2252.09 | 2263.31 | -8.04 |
| 34 | 1237.31 | 1253.35 | 1273.55 | 1183.55 | 1191.01 | 1198.63 | -4.98 | 1183.49 | 1191.29 | 1202.35 | -4.90 | 1146.70 | 1155.32 | 1163.31 | -7.80 |
| 35 | 1552.17 | 1583.75 | 1620.64 | 1431.66 | 1442.52 | 1458.14 | -8.36 | 1438.17 | 1451.44 | 1468.40 | -7.78 | 1390.79 | 1395.88 | 1404.34 | -11.27 |
| 36 | 1810.15 | 1832.04 | 1854.00 | 1722.98 | 1737.80 | 1759.05 | -5.20 | 1730.03 | 1743.72 | 1761.87 | -4.79 | 1665.91 | 1675.69 | 1688.92 | -8.56 |
| AVG | 1177.14 | 1189.21 | 1202.78 | 1138.48 | 1144.85 | 1152.50 | -3.04 | 1135.46 | 1142.88 | 1152.11 | -3.22 | 1112.33 | 1117.31 | 1123.50 | -4.96 |

Table 4.15: Aggregate Results for the Complete Set of Instances (Averages over 36 × 5 Instances per Line) with the Four Different Loading Configurations

| | $sec_{max}$ | TS[a] $z$ | TS[a] $sec_h$ | TS[a] $sec_{tot}$ | GTS[b] $z$ | GTS[b] $sec_h$ | GTS[b] $sec_{tot}$ | ACO $z_{min}$ | ACO $\overline{z}$ | ACO $z_{max}$ | ACO $\overline{sec}_h$ | ACO $\overline{sec}_{tot}$ | %$\overline{gap}$ [c] | %$\overline{gap}$ [d] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2\|RO\|L | 1800 | | | | | | | 1114.95 | 1129.48 | 1146.25 | 614.6 | 621.3 | -2.84 | -1.26 |
| 2\|UO\|L | 1800 | | | | | | | 1072.48 | 1080.81 | 1091.03 | 493.0 | 501.9 | -3.22 | -1.45 |
| 2\|RO\|L | 3600 | 1171.75 | 936.5 | 1817.0 | 1142.23 | 829.8 | 1315.2 | 1106.59 | 1119.73 | 1135.04 | 1024.9 | 1034.5 | -3.30 | -1.74 |
| 2\|UO\|L | 3600 | 1131.33 | 757.9 | 1822.2 | 1100.46 | 650.8 | 1038.5 | 1069.58 | 1077.15 | 1086.34 | 856.1 | 866.7 | -3.39 | -1.63 |
| 2\|RO\|L | 7200 | | | | | | | 1100.25 | 1111.89 | 1125.14 | 1764.2 | 1777.3 | -3.63 | -2.10 |
| 2\|UO\|L | 7200 | | | | | | | 1066.88 | 1073.89 | 1082.39 | 1540.4 | 1568.1 | -3.54 | -1.79 |
| 2\|RO\|L | 10800 | | | | | | | 1096.88 | 1107.92 | 1120.70 | 2442.6 | 2513.0 | -3.81 | -2.28 |
| 2\|UO\|L | 10800 | | | | | | | 1065.94 | 1072.44 | 1080.47 | 2058.8 | 2170.4 | -3.61 | -1.87 |

[a] see Gendreau et al. (2008), Pentium IV, 1.7 GHz
[b] see Zachariadis, Tarantilis, and Kiranoudis (2007), Pentium IV, 2.4 GHz

Table 4.16: Aggregate Results on the Complete Set of Instances (Averages over 180 Instances per Line) with Different Runtime Limits

| | ACO $\alpha_3 = 1$ | | | | | ACO $\alpha_3 = 5$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
| 2\|RO\|L | 1100.21 | 1111.77 | 1125.48 | 2462.4 | 2560.3 | 1096.88 | 1107.92 | 1120.70 | 2442.6 | 2513.0 |
| 2\|UO\|L | 1066.25 | 1073.20 | 1081.82 | 2165.3 | 2285.1 | 1065.94 | 1072.44 | 1080.47 | 2058.8 | 2170.4 |
| 2\|RN\|L | 1063.80 | 1071.84 | 1080.98 | 2247.9 | 2352.4 | 1063.53 | 1070.86 | 1080.15 | 2156.6 | 2264.5 |
| 2\|UN\|L | 1044.51 | 1050.55 | 1057.28 | 1818.1 | 1950.2 | 1045.03 | 1050.41 | 1057.27 | 1534.0 | 1631.4 |

Table 4.17: Aggregate Results on the Complete Set of Instances (Averages over 180 Instances Per Line) with Four Different Loading Configurations

## 4.3 3L-CVRP

### 4.3.1 Problem Instances

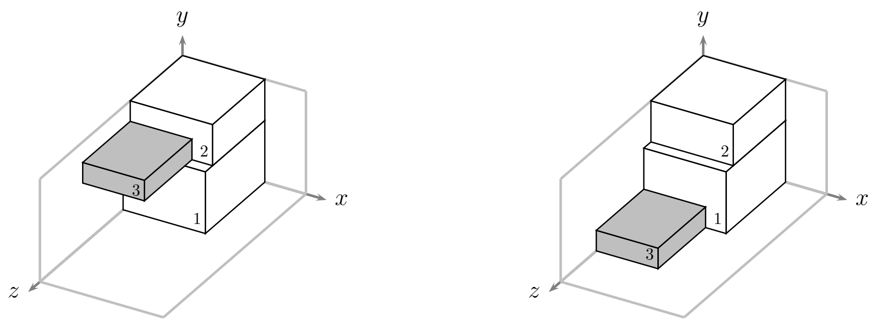
The ACO algorithm was tested on the set of instances proposed in Gendreau et al. (2006), that can be downloaded from `http://www.or.deis.unibo.it/research.html`. They provide an interesting test bed since heuristic solutions are available for comparison. In the instances, the graphs, the weights demanded by the customers and the vehicle weight capacities are taken from 27 Euclidean CVRP instances (see Toth and Vigo (2002*b*) for a detailed description of CVRP test bed instances). The arc costs are determined as the Euclidean (not rounded) distances between customers coordinates. The loading volume has dimensions W = 25, H = 30 and L = 60. For each customer the number of requested items is randomly generated according to a uniform distribution between 1 and 3. For each item, its dimension is randomly generated according to a uniform distribution in the interval between 20% and 60% of the corresponding vehicle dimension. The benchmark algorithm in Gendreau et al. (2006) was run on a Pentium IV 3 GHz with 512 MB of RAM, under Windows XP operation system. Being deterministic, it was run a single time on each instance. It was allowed a CPU time limit of 1800 seconds for instances 1 - 9, 3600 for instances 10 - 18 and 7200 seconds for instances 19 - 27. The ACO is halted instead when $2\tilde{n}$ iterations have been performed, or 3 CPU hours have been elapsed. The parameter setting is identical to the one used for the 2L-CVRP (see Table 3.3). All experiments concerning the ACO algorithm have been implemented in C++, compiled with g++ compiler and run on a Pentium 4 with 3.2 GHz.

### 4.3.2 Results on the Complete Set of Instances

Table 4.18 gives an aggregate comparison between the TS approach by Gendreau et al. (2006) and the ACO algorithm. For the ACO four different settings of the touching area heuristic have been tested. If an item is placed on the vehicle floor and $bot = 1$, the item bottom is added to its touching area, if $bot = 0$ the item bottom is ignored for calculating its touching area. This measure favors the stacking of items. Furthermore the multiplier $stack$ can be used to emphasize the stacking of items by multiplying the touching area generated on the x-z plane of two items by $stack$. Table 4.18 clearly shows that favoring the stacking either by ignoring the vehicle bottom or increasing $stack$ is reasonable because it improves the average improvement towards the TS by about 0.4% percentage points. Apparently the best choice is to set $bot = 0$ and $stack = 1$, which is therefore chosen for all further investigations in this chapter.
Remember that for $3|R|N|1|75|L$ the items of the customers of one route are sorted in reverse order of visit. The items of one customer are sorted by their fragility status (non fragile ones first) breaking ties by non-increasing volume. Replacing the last tie breaker by non-increasing base area yields nearly identical results.
Table 4.19 shows the aggregate results for all investigated loading configurations comparing the solutions of ACO and TS. It is quite obvious that ACO outperforms TS. The largest average improvement by 6.43% can be reached for the most constrained loading

| TS | | ACO | | | | | | | %gap ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $z$ | $sec_h$ | $bot$ | $stack$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1042.26 | 2058.9 | 1 | 1 | 965.15 | 972.48 | 980.17 | 1798.6 | 1838.8 | -6.59 | -5.97 | -5.29 |
| 1042.26 | 2058.9 | 1 | 2 | 962.34 | 968.77 | 976.52 | 1730.9 | 1804.5 | -6.90 | -6.32 | -5.62 |
| 1042.26 | 2058.9 | 0 | 1 | 960.87 | 966.66 | 972.54 | 1746.6 | 1793.1 | -6.98 | -6.43 | -5.89 |
| 1042.26 | 2058.9 | 0 | 2 | 961.94 | 967.56 | 973.41 | 1729.7 | 1767.5 | -6.91 | -6.41 | -5.87 |

Table 4.18: Aggregate Results for the Complete Set of Instances (Averages over 27 Instances) for $3|R|N|1|75|L$ with Four Different Parameters for the Touching Area Heuristic




(a) $bot = 0$ (b) $bot = 1$

Figure 4.3: Different Placements of Gray Item Depending on *bot*

$3|R|N|1|75|L$. The smallest average improvement by 2.03% is obtained for the simplest loading $3|U|N|0|0|L$. Detailed results for selected problems can be found in appendix B. Apart from that, both versions of the touching area heuristic (setting *bot* to 0 or 1) are investigated. For the $3|R|N|1|75|L$ and $3|R|N|0|75|L$ case results are clearly better when $bot = 0$ yielding an average improvement of about 0.4% compared to the TS. For $3|U|N|1|75|L$, $3|R|N|1|0|L$ and $3|U|N|0|0|L$ the opposite is true. For the latter two configurations there is a simple reason why adding the touching area of items with the vehicle bottom is reasonable: $\underline{A} = 0$. No supporting area is required and therefore by setting $bot = 1$ circumvents "nearly flying items" as depicted in Figure 4.3. When the touching area of and item with the bin floor is disregarded ($bot = 0$) item 3 would be placed as shown in Figure 4.3a with a touching area of $w_3h_3 + l_3h_3 + (l_1 - l_2)w_3$, which is clearly larger than locating it on the vehicle floor (touching area $= w_3h_3 + l_3h_3$) as shown in Figure 4.3b. If $bot = 1$ item 3 would be placed on the vehicle floor with a touching area of $w_3h_3 + l_3h_3 + l_3w_3$. However if $suppA \geq 50\%$ this additional constraint would force item 3 on the vehicle floor and $bot = 0$ would indeed favor the stacking of items when all constraints are met as intended. Nevertheless it is not obvious why for $3|R|N|1|0|L$ $bot = 1$ gives slightly better results than $bot = 0$. However one may conclude from the presented empirical evidence that $bot = 0$ is only reasonable when LIFO loading is required and $\underline{A} > 0$. For all other cases it is better to set $bot = 1$.

Another interesting question to investigate is the role or impact of the virtual cuboid ($\alpha_4$) on the solution quality. Table 4.20 shows the aggregate results of the ACO algorithm with different weights for the virtual cuboid. Concerning solution quality there is almost no difference between the different weights, however the average runtime until the best solution is found ($\overline{sec}_h$) and the total runtime ($\overline{sec}_{tot}$) significantly decrease when larger values of $\alpha_4$ are used. For all experiments except for this table presented $\alpha_4$ is set to 1.

| | TS | | ACO | | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $z$ | $sec_h$ | $bot$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| $3\|R\|N\|1\|75\|L$ | 1042.26 | 2058.9 | 0 | 960.87 | 966.66 | 972.54 | 1746.6 | 1793.1 | -6.98 | -6.43 | -5.89 |
| $3\|R\|N\|1\|75\|L$ | 1042.26 | 2058.9 | 1 | 965.15 | 972.48 | 980.17 | 1798.6 | 1838.8 | -6.59 | -5.97 | -5.29 |
| $3\|R\|N\|0\|75\|L$ | 1014.49 | 2410.8 | 0 | 940.30 | 945.04 | 950.76 | 1337.0 | 1375.1 | -6.49 | -6.15 | -5.68 |
| $3\|R\|N\|0\|75\|L$ | 1014.49 | 2410.8 | 1 | 943.29 | 948.09 | 953.53 | 1337.8 | 1375.3 | -6.16 | -5.78 | -5.31 |
| $3\|U\|N\|1\|75\|L$ | 951.19 | 1709.8 | 0 | 910.34 | 916.25 | 921.23 | 676.6 | 716.8 | -3.86 | -3.41 | -2.99 |
| $3\|U\|N\|1\|75\|L$ | 951.19 | 1709.8 | 1 | 909.68 | 913.65 | 918.94 | 687.8 | 722.8 | -3.89 | -3.58 | -3.18 |
| $3\|R\|N\|1\|0\|L$ | 939.53 | 1882.5 | 0 | 914.72 | 919.69 | 926.02 | 1340.0 | 1376.0 | -2.14 | -1.75 | -1.16 |
| $3\|R\|N\|1\|0\|L$ | 939.53 | 1882.5 | 1 | 901.09 | 905.07 | 910.09 | 1547.6 | 1582.6 | -3.70 | -3.29 | -2.77 |
| $3\|U\|N\|0\|0\|L$ | 876.31 | 1567.4 | 0 | 854.39 | 856.67 | 858.86 | 689.3 | 741.5 | -1.97 | -1.75 | -1.55 |
| $3\|U\|N\|0\|0\|L$ | 876.31 | 1567.4 | 1 | 849.62 | 852.10 | 854.65 | 825.5 | 880.2 | -2.29 | -2.03 | -1.79 |

Table 4.19: Aggregate Results for the Complete Set of Instances (Averages over 27 Instances) for Five Different Loading Configurations with a Runtime Limit of 10800 Seconds

| | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
|---|---|---|---|---|---|
| $\alpha_4 = 0$ | 959.94 | 967.87 | 973.91 | 1794.0 | 1833.15 |
| $\alpha_4 = 1$ | 960.87 | 966.66 | 972.54 | 1746.6 | 1793.11 |
| $\alpha_4 = 5$ | 961.62 | 967.16 | 973.45 | 1557.0 | 1605.65 |

Table 4.20: Aggregate Results for the Complete Set of Instances (Averages over 27 Instances) for $3|R|N|1|75|L$ and Different Values of $\alpha_4$

As in the 2L-CVRP case the ACO and the TS algorithms have been run on different but quite similar machines and have been granted different runtime limits. Tables 4.21 and 4.22 present the same information as Table 4.19 but with a runtime limit of 1800 and 3600 seconds respectively. For 1800 seconds runtime limit the largest improvement by 5.35% is again obtained for $3|R|N|1|75|L$ and the smallest improvement by 1.80% for $3|U|N|0|0|L$. For 3600 runtime limit the largest (smallest) improvement by 5.78% (1.99%) is obtained for $3|R|N|0|75|L$ ($3|U|N|0|0|L$).

Last but not least the influence of the different loading constraints on the objective value is investigated. Table 4.23 gives an overview of 13 different loading configurations tested with the relative deviation to $3|R|N|1|75|L$. An interesting detail concerning the tuning of the touching area heuristic with the parameter *bot* refers to the comparison of $3|R|N|1|0|L$ and $3|R|O|1|0|L$. If rotation is allowed using bot= 1 instead of bot= 0 is slightly better and for the case that rotation is not allowed it is the other way round. The influence of allowing rotation on the x-z plane is reported in the last column: On the one hand the objective value increases at about 1% on average if rotation is not allowed but on the other hand also the the runtime decreases significantly because only the oriented positions have

| | TS | | ACO | | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $z$ | $sec_h$ | $bot$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| $3\|R\|N\|1\|75\|L$ | 1042.26 | 2058.9 | 0 | 973.65 | 982.32 | 989.48 | 555.2 | 714.3 | -6.08 | -5.35 | -4.71 |
| $3\|R\|N\|1\|75\|L$ | 1042.26 | 2058.9 | 1 | 983.08 | 993.3 | 1001.56 | 545.1 | 722.1 | -5.28 | -4.41 | -3.68 |
| $3\|R\|N\|0\|75\|L$ | 1014.5 | 2410.8 | 0 | 950.07 | 956.51 | 962.18 | 532.3 | 669.5 | -5.79 | -5.30 | -4.81 |
| $3\|R\|N\|0\|75\|L$ | 1014.5 | 2410.8 | 1 | 955.77 | 961.63 | 967.46 | 535.7 | 674.6 | -5.25 | -4.77 | -4.27 |
| $3\|U\|N\|1\|75\|L$ | 951.19 | 1709.8 | 0 | 914.50 | 919.79 | 924.28 | 461.3 | 546.7 | -3.58 | -3.17 | -2.78 |
| $3\|U\|N\|1\|75\|L$ | 951.19 | 1709.8 | 1 | 912.56 | 917.25 | 922.43 | 456.1 | 548.5 | -3.70 | -3.34 | -2.94 |
| $3\|R\|N\|1\|0\|L$ | 939.53 | 1882.5 | 0 | 923.24 | 928.69 | 933.47 | 491.3 | 678.7 | -1.44 | -1.02 | -0.58 |
| $3\|R\|N\|1\|0\|L$ | 939.53 | 1882.5 | 1 | 910.45 | 914.69 | 918.76 | 512.4 | 708.4 | -2.93 | -2.51 | -2.08 |
| $3\|U\|N\|0\|0\|L$ | 876.31 | 1567.4 | 0 | 856.26 | 858.94 | 861.14 | 472.7 | 574.8 | -1.83 | -1.58 | -1.38 |
| $3\|U\|N\|0\|0\|L$ | 876.31 | 1567.4 | 1 | 851.95 | 854.94 | 857.56 | 458.0 | 613.2 | -2.12 | -1.80 | -1.55 |

Table 4.21: Aggregate Results for the Complete Set of Instances (Averages over 27 Instances) for Five Different Loading Configurations with a Runtime Limit of 1800 Seconds

| | TS | | ACO | | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $z$ | $sec_h$ | $bot$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| $3\|R\|N\|1\|75\|L$ | 1042.26 | 2058.9 | 0 | 970.97 | 977.13 | 983.02 | 962.7 | 1148.5 | -6.32 | -5.77 | -5.24 |
| $3\|R\|N\|1\|75\|L$ | 1042.26 | 2058.9 | 1 | 977.79 | 986.10 | 993.95 | 957.2 | 1190.0 | -5.72 | -4.98 | -4.27 |
| $3\|R\|N\|0\|75\|L$ | 1014.5 | 2410.8 | 0 | 945.99 | 951.21 | 956.92 | 865.8 | 1024.1 | -6.14 | -5.78 | -5.31 |
| $3\|R\|N\|0\|75\|L$ | 1014.5 | 2410.8 | 1 | 948.70 | 955.14 | 960.95 | 894.0 | 1028.7 | -5.84 | -5.36 | -4.85 |
| $3\|U\|N\|1\|75\|L$ | 951.19 | 1709.8 | 0 | 910.97 | 917.12 | 921.85 | 640.3 | 684.0 | -3.81 | -3.34 | -2.95 |
| $3\|U\|N\|1\|75\|L$ | 951.19 | 1709.8 | 1 | 910.94 | 914.58 | 919.69 | 636.8 | 688.4 | -3.80 | -3.52 | -3.12 |
| $3\|R\|N\|1\|0\|L$ | 939.53 | 1882.5 | 0 | 919.27 | 924.65 | 930.19 | 878.8 | 1043.6 | -1.82 | -1.40 | -0.87 |
| $3\|R\|N\|1\|0\|L$ | 939.53 | 1882.5 | 1 | 905.19 | 909.99 | 914.85 | 957.6 | 1149.4 | -3.39 | -2.92 | -2.43 |
| $3\|U\|N\|0\|0\|L$ | 876.31 | 1567.4 | 0 | 854.56 | 856.92 | 859.28 | 671.9 | 721.5 | -1.96 | -1.73 | -1.52 |
| $3\|U\|N\|0\|0\|L$ | 876.31 | 1567.4 | 1 | 850.13 | 852.63 | 855.38 | 752.6 | 824.3 | -2.25 | -1.99 | -1.73 |

Table 4.22: Aggregate Results for the Complete Set of Instances (Averages over 27 Instances) for Five Different Loading Configurations with a Runtime Limit of 3600 Seconds

| | ACO | | | | | | %gap to 3\|R\|N\|1\|75\|L | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $bot$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{z}_{norot} - \overline{z}_{rot}$ [a] |
| 3\|R\|N\|1\|75\|L | 1 | 960.87 | 966.66 | 972.54 | 1746.6 | 1793.1 | 0.00 | 0.00 | 0.00 | |
| 3\|R\|N\|1\|**50**\|L | 1 | 927.44 | 933.16 | 939.52 | 1349.0 | 1385.3 | -3.07 | -3.14 | -3.14 | |
| 3\|R\|N\|1\|**25**\|L | 1 | 916.30 | 923.49 | 929.47 | 1338.2 | 1377.9 | -4.00 | -3.92 | -3.97 | |
| 3\|R\|N\|1\|**0**\|L | 0 | 901.09 | 905.07 | 910.09 | 1547.6 | 1582.6 | -5.46 | -5.61 | -5.65 | |
| 3\|R\|N\|**0**\|75\|L | 1 | 940.30 | 945.04 | 950.76 | 1337.0 | 1375.1 | -1.87 | -2.09 | -2.17 | |
| 3\|**U**\|N\|1\|75\|L | 0 | 909.68 | 913.65 | 918.94 | 687.8 | 722.8 | -4.71 | -4.97 | -5.11 | |
| 3\|**U**\|N\|**0**\|**0**\|L | 0 | 849.62 | 852.10 | 854.65 | 825.5 | 880.2 | -9.64 | -9.93 | -10.23 | |
| 3\|R\|**O**\|1\|75\|L | 0 | 968.86 | 976.38 | 982.95 | 1464.2 | 1500.6 | 0.91 | 1.02 | 1.09 | 1.02 |
| 3\|R\|**O**\|1\|0\|L | 1 | 908.63 | 913.20 | 917.69 | 1169.3 | 1206.8 | -4.69 | -4.88 | -5.01 | 0.73 |
| 3\|R\|**O**\|**0**\|75\|L | 0 | 946.92 | 951.47 | 955.80 | 1077.9 | 1112.1 | -1.26 | -1.49 | -1.66 | 0.61 |
| 3\|**U**\|O\|1\|75\|L | 0 | 926.21 | 929.10 | 932.32 | 545.4 | 582.1 | -3.24 | -3.57 | -3.85 | 1.40 |
| 3\|**U**\|O\|**0**\|**0**\|L | 1 | 862.02 | 865.63 | 868.34 | 647.7 | 689.6 | -8.72 | -8.95 | -9.25 | 0.98 |

[a] $\overline{z}_{norot} - \overline{z}_{rot}$: absolute difference of average gap between two loading configurations w. r. t. rotation.

Table 4.23: Aggregate Results for the Complete Set of Instances (Averages over 27 Instances) for 13 Different Loading Configurations with a Runtime Limit of 10800 Seconds

to be checked.

# 5 Conclusion

This thesis presents three different problems concerning the combination of routing and loading (packing) problems that have been addressed in the literature quite recently. The presented ACO approaches are highly effective compared to other solution techniques addressing the same problems not only with respect to solution quality but also computational runtime (see Sections 4.1, 4.2 and 4.3). Additionally the impacts of different loading constraints has been intensively studied and quite clear managerial advices can be drawn from the presented results. The next paragraphs are going to illuminate the hot spots for each problem.

In the MP-VRP, which has been motivated by real world application, the ACO used two pheromone matrices: one for the routing and one for the packing, which allowed to guide the search towards pure routing solutions and solutions with better loading patterns, less vehicles but larger total routing costs. Moreover besides the savings values $s_{ij}$, $g_{ij}$ measured the quality of combing customer $i$ and $j$ with respect to the amount of bulk material needed. Both matrices could be calculated in a preprocessing step. However as indicated in Section 4.1.3 the second pheromone matrix is dispensable because the most interesting solutions have been found when the ants only exploited the routing information (consisting of savings values and pheromone information). The loading subproblem has not been treated in the literature so far. Therefore the simple $HL$ loading heuristic has been created. It is based on the notion of forming pairs of consecutive customers. Based on this root idea two dynamic programming approaches have been implemented to achieve better loading solutions. Nevertheless the increase in runtime has not been compensated by the unconvincing gain in solution quality. The ACO could outperform an appropriate TS concerning the solution quality and runtime.

For the 2L-CVRP and 3L-CVRP the usage of a second pheromone matrix for good loading combinations has been abandoned. The main reason for this decision was the experience in the MP-VRP. Nevertheless it has been shown that without any visibility for the loading some instances could not be solved. Therefore a new fast measure, the virtual rectangle (virtual cuboid) has been created, to bias ants' decisions not only towards low routing cost but also towards vehicles utilizing their loading capacity. Additionally the objective function for the local search has been modified to be able to distinguish between solutions with $K + \varepsilon$ vehicles. Another crucial finding is the pool of already checked solutions that tremendously speeds up the algorithm. For the touching area heuristic the parameter *bot* has been found to be useful when supporting area is required.

Although the ACO yields really excellent performance compared to other approaches there are still several open questions:
For practitioners normally a feasible loading is additionally assessed by constraints like hubloads, balancing of weight across the loading space and loading safety considerations. This has not been regarded for any of the three problems. However it should be possible to incorporate these constraints in the loading routines easily to a certain extent.
A more complex problem is the creation of an exact method for the 3D-1-VLP. As mentioned before rotation is a two edged constraint: On the one hand rotation weakens the best available lower bounds considerably and increases the complexity of the problem, but on the other hand it allows more compact loadings. Moreover this thesis shows that the generation of new placing points for subsequent items in the 3D-1-VLP is not trivial when $\underline{A} \in ]0, 100[$. Thus it is necessary for future research to investigate which placing points have to be added to guarantee that the optimal solution can be found.

# A Detailed Results for the 2L-CVRP

| | | TS | | | GTS | ACO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
| 1 | 1 | 278.73 | 2.0 | 2.2 | 278.73 | 278.73 | 278.73 | 278.73 | 0.1 | 0.4 |
| 1 | 2 | 301.45 | 4.9 | 6.2 | 319.86 | 290.84 | 291.50 | 297.42 | 7.3 | 8.9 |
| 1 | 3 | 313.91 | 2.9 | 12.2 | 314.33 | 304.41 | 305.20 | 307.44 | 16.0 | 19.1 |
| 1 | 4 | 296.75 | 1.1 | 8.4 | 296.75 | 296.75 | 297.27 | 297.80 | 0.7 | 1.3 |
| 1 | 5 | 284.23 | 2.0 | 17.2 | 285.93 | 285.93 | 285.93 | 285.93 | 0.6 | 1.9 |
| 2 | 1 | 334.96 | 0.0 | 1.4 | 334.96 | 334.96 | 334.96 | 334.96 | 0.1 | 0.3 |
| 2 | 2 | 347.73 | 1.6 | 1.9 | 347.73 | 347.73 | 347.73 | 347.73 | 0.1 | 0.4 |
| 2 | 3 | 356.24 | 0.0 | 3.3 | 356.24 | 356.24 | 357.56 | 361.83 | 0.5 | 1.2 |
| 2 | 4 | 342.00 | 0.3 | 4.8 | 342.00 | 342.00 | 342.33 | 345.36 | 0.2 | 0.5 |
| 2 | 5 | 334.96 | 0.2 | 6.1 | 340.88 | 334.96 | 334.96 | 334.96 | 0.3 | 0.5 |
| 3 | 1 | 359.77 | 3.5 | 8.4 | 358.40 | 358.40 | 358.40 | 358.40 | 0.2 | 0.8 |
| 3 | 2 | 411.24 | 7.5 | 9.9 | 414.39 | 403.93 | 403.93 | 403.93 | 1.2 | 2.3 |
| 3 | 3 | 394.72 | 0.5 | 19.7 | 413.63 | 394.72 | 398.15 | 400.44 | 3.6 | 4.5 |
| 3 | 4 | 376.83 | 5.3 | 25.7 | 383.11 | 368.56 | 372.12 | 376.35 | 3.5 | 4.1 |
| 3 | 5 | 358.40 | 2.4 | 30.9 | 362.27 | 358.40 | 358.40 | 358.40 | 0.4 | 1.6 |
| 4 | 1 | 430.88 | 0.1 | 5.7 | 430.88 | 430.89 | 430.89 | 430.89 | 0.3 | 0.7 |
| 4 | 2 | 440.94 | 0.7 | 7.7 | 451.98 | 440.94 | 440.94 | 440.94 | 1.5 | 2.1 |
| 4 | 3 | 446.61 | 4.7 | 14.2 | 448.24 | 445.25 | 445.25 | 445.25 | 3.2 | 3.2 |
| 4 | 4 | 455.25 | 1.0 | 19.3 | 447.37 | 447.37 | 447.37 | 447.37 | 4.2 | 5.4 |
| 4 | 5 | 430.88 | 0.3 | 37.8 | 430.88 | 430.89 | 432.32 | 445.25 | 1.6 | 2.4 |
| 5 | 1 | 375.28 | 1.4 | 12.8 | 375.28 | 375.28 | 375.28 | 375.28 | 0.3 | 1.0 |
| 5 | 2 | 390.62 | 7.4 | 17.8 | 407.45 | 388.72 | 388.72 | 388.72 | 12.1 | 18.5 |
| 5 | 3 | 383.87 | 3.0 | 21.6 | 401.09 | 381.69 | 382.53 | 383.88 | 19.9 | 24.3 |
| 5 | 4 | 386.47 | 4.8 | 28.9 | 399.65 | 383.88 | 387.01 | 391.71 | 16.7 | 20.6 |
| 5 | 5 | 375.28 | 4.0 | 56.9 | 377.26 | 375.28 | 375.28 | 375.28 | 3.3 | 5.2 |
| 6 | 1 | 495.85 | 0.3 | 8.7 | 495.85 | 495.85 | 495.85 | 495.85 | 0.3 | 0.9 |
| 6 | 2 | 499.08 | 7.1 | 12.6 | 499.08 | 499.08 | 500.34 | 503.29 | 1.9 | 3.5 |
| 6 | 3 | 504.68 | 13.0 | 15.3 | 509.65 | 504.68 | 504.68 | 504.68 | 4.8 | 5.8 |
| 6 | 4 | 511.52 | 5.1 | 30.8 | 515.60 | 498.32 | 500.46 | 504.78 | 6.9 | 7.8 |
| 6 | 5 | 495.85 | 0.1 | 30.0 | 495.85 | 495.85 | 495.85 | 495.85 | 2.0 | 3.0 |
| 7 | 1 | 568.56 | 0.5 | 22.6 | 568.56 | 568.56 | 568.56 | 568.56 | 0.2 | 1.3 |
| 7 | 2 | 751.15 | 2.4 | 20.0 | 764.45 | 734.65 | 734.65 | 734.65 | 6.8 | 8.7 |
| 7 | 3 | 702.59 | 9.3 | 35.2 | 712.57 | 709.72 | 717.34 | 732.52 | 9.3 | 11.3 |
| 7 | 4 | 732.54 | 29.3 | 53.6 | 723.78 | 703.49 | 703.67 | 703.85 | 13.1 | 14.4 |
| 7 | 5 | 701.31 | 36.0 | 133.8 | 694.54 | 661.22 | 669.08 | 676.81 | 12.7 | 15.5 |

| | | TS | | | GTS | ACO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
| 8 | 1 | 568.56 | 0.5 | 36.2 | 568.56 | 568.56 | 568.56 | 568.56 | 0.2 | 1.3 |
| 8 | 2 | 730.87 | 1.5 | 19.3 | 730.87 | 725.91 | 726.70 | 730.87 | 6.2 | 9.2 |
| 8 | 3 | 771.29 | 14.1 | 37.8 | 749.70 | 741.12 | 741.12 | 741.12 | 12.0 | 12.0 |
| 8 | 4 | 723.55 | 45.8 | 61.4 | 727.58 | 723.11 | 732.20 | 735.28 | 7.4 | 12.8 |
| 8 | 5 | 665.20 | 102.0 | 263.9 | 654.74 | 646.46 | 647.55 | 651.91 | 20.9 | 50.4 |
| 9 | 1 | 607.65 | 0.4 | 13.5 | 607.65 | 607.65 | 607.65 | 607.65 | 0.6 | 1.3 |
| 9 | 2 | 625.13 | 1.8 | 17.6 | 611.49 | 611.49 | 617.79 | 624.09 | 1.8 | 2.7 |
| 9 | 3 | 638.31 | 23.5 | 36.9 | 644.54 | 613.90 | 617.52 | 631.37 | 8.6 | 8.9 |
| 9 | 4 | 625.13 | 5.0 | 46.8 | 625.13 | 625.13 | 628.28 | 630.76 | 4.2 | 5.9 |
| 9 | 5 | 607.65 | 22.6 | 85.1 | 607.65 | 607.65 | 607.65 | 607.65 | 3.0 | 4.2 |
| 10 | 1 | 538.79 | 6.1 | 81.7 | 535.80 | 535.80 | 535.80 | 535.80 | 2.3 | 4.2 |
| 10 | 2 | 715.51 | 49.3 | 72.5 | 740.43 | 700.20 | 700.20 | 700.20 | 37.7 | 40.5 |
| 10 | 3 | 660.27 | 18.7 | 140.4 | 671.24 | 628.94 | 642.05 | 655.16 | 43.1 | 44.9 |
| 10 | 4 | 769.73 | 59.0 | 195.1 | 770.82 | 764.85 | 769.07 | 777.59 | 49.7 | 51.1 |
| 10 | 5 | 714.08 | 84.4 | 408.4 | 728.95 | 699.22 | 709.09 | 713.81 | 71.8 | 78.0 |
| 11 | 1 | 505.01 | 2.5 | 98.9 | 505.01 | 505.01 | 505.01 | 505.01 | 0.8 | 3.1 |
| 11 | 2 | 754.62 | 23.6 | 76.6 | 748.96 | 721.54 | 722.44 | 723.34 | 26.3 | 28.8 |
| 11 | 3 | 789.95 | 10.5 | 122.8 | 783.92 | 738.54 | 747.72 | 756.67 | 28.6 | 30.8 |
| 11 | 4 | 899.46 | 98.4 | 232.5 | 868.01 | 824.30 | 833.34 | 846.18 | 84.6 | 85.1 |
| 11 | 5 | 649.75 | 360.1 | 466.2 | 671.71 | 658.71 | 660.84 | 663.40 | 69.0 | 74.2 |
| 12 | 1 | 610.57 | 28.5 | 32.5 | 610.00 | 610.00 | 611.22 | 614.24 | 1.5 | 2.4 |
| 12 | 2 | 641.84 | 24.4 | 45.2 | 638.06 | 619.63 | 619.63 | 619.63 | 4.1 | 6.0 |
| 12 | 3 | 610.57 | 24.8 | 52.9 | 610.57 | 610.00 | 611.12 | 614.59 | 2.0 | 3.6 |
| 12 | 4 | 664.76 | 39.8 | 124.7 | 655.60 | 623.20 | 623.20 | 623.20 | 11.7 | 12.1 |
| 12 | 5 | 610.23 | 176.3 | 242.3 | 610.23 | 610.23 | 613.60 | 619.92 | 5.3 | 6.5 |
| 13 | 1 | 2006.34 | 29.9 | 161.6 | 2006.34 | 2006.34 | 2006.34 | 2006.34 | 1.3 | 4.0 |
| 13 | 2 | 2836.79 | 50.8 | 73.8 | 2778.28 | 2669.39 | 2672.98 | 2687.33 | 29.3 | 32.8 |
| 13 | 3 | 2625.82 | 30.3 | 186.4 | 2660.74 | 2497.42 | 2504.38 | 2528.74 | 64.5 | 64.9 |
| 13 | 4 | 2743.17 | 8.8 | 299.6 | 2748.57 | 2689.59 | 2699.05 | 2725.66 | 60.5 | 60.7 |
| 13 | 5 | 2542.34 | 125.1 | 842.7 | 2529.77 | 2512.41 | 2535.31 | 2544.43 | 79.8 | 91.8 |
| 14 | 1 | 837.67 | 22.2 | 152.1 | 837.67 | 837.67 | 837.67 | 837.67 | 4.2 | 6.1 |
| 14 | 2 | 1195.39 | 14.1 | 138.5 | 1185.52 | 1105.00 | 1133.38 | 1139.11 | 99.9 | 103.6 |
| 14 | 3 | 1156.33 | 154.9 | 235.3 | 1139.74 | 1098.42 | 1103.48 | 1109.41 | 126.3 | 129.9 |
| 14 | 4 | 1058.98 | 130.9 | 465.8 | 1036.37 | 993.47 | 995.23 | 1002.79 | 170.7 | 180.6 |
| 14 | 5 | 995.25 | 408.1 | 1205.6 | 1009.49 | 946.31 | 957.23 | 970.27 | 224.3 | 230.2 |
| 15 | 1 | 837.67 | 1.8 | 182.5 | 837.67 | 837.67 | 837.67 | 837.67 | 2.8 | 4.3 |
| 15 | 2 | 1143.73 | 4.7 | 119.8 | 1120.00 | 1110.84 | 1117.28 | 1118.02 | 85.1 | 97.3 |
| 15 | 3 | 1209.60 | 132.6 | 175.3 | 1188.63 | 1192.19 | 1195.65 | 1201.07 | 149.3 | 150.4 |
| 15 | 4 | 1311.09 | 14.7 | 364.3 | 1329.39 | 1272.66 | 1292.45 | 1308.20 | 119.0 | 123.0 |
| 15 | 5 | 1299.14 | 673.3 | 725.4 | 1298.80 | 1271.64 | 1285.36 | 1294.37 | 219.4 | 222.2 |
| 16 | 1 | 698.61 | 2.8 | 99.0 | 698.61 | 698.61 | 698.61 | 698.61 | 2.0 | 3.9 |
| 16 | 2 | 698.61 | 15.3 | 92.8 | 704.58 | 698.61 | 700.45 | 704.08 | 4.6 | 6.6 |
| 16 | 3 | 698.61 | 18.2 | 135.2 | 708.83 | 698.61 | 700.21 | 706.61 | 9.2 | 10.8 |
| 16 | 4 | 723.58 | 100.3 | 200.8 | 718.21 | 709.27 | 709.27 | 709.27 | 10.9 | 12.2 |

| | | TS | | | GTS | ACO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
| 16 | 5 | 698.61 | 3.6 | 258.1 | 698.61 | 698.61 | 698.61 | 698.61 | 4.6 | 6.9 |
| 17 | 1 | 862.62 | 59.0 | 162.8 | 863.27 | 861.79 | 862.37 | 862.62 | 3.3 | 6.2 |
| 17 | 2 | 874.34 | 25.7 | 135.4 | 870.86 | 870.86 | 871.31 | 875.44 | 4.7 | 7.2 |
| 17 | 3 | 862.62 | 73.0 | 198.8 | 861.79 | 861.79 | 862.76 | 865.92 | 3.4 | 6.5 |
| 17 | 4 | 866.42 | 151.0 | 283.6 | 864.58 | 862.62 | 862.62 | 862.62 | 4.8 | 8.1 |
| 17 | 5 | 862.62 | 136.0 | 350.5 | 863.58 | 861.79 | 862.43 | 863.27 | 3.4 | 6.7 |
| 18 | 1 | 723.54 | 81.9 | 587.3 | 730.85 | 723.54 | 726.18 | 727.74 | 9.4 | 14.8 |
| 18 | 2 | 1129.51 | 172.6 | 307.8 | 1092.60 | 1059.44 | 1059.45 | 1059.50 | 169.6 | 169.6 |
| 18 | 3 | 1130.19 | 246.1 | 831.0 | 1133.13 | 1115.67 | 1120.94 | 1124.11 | 147.5 | 153.6 |
| 18 | 4 | 1209.25 | 197.1 | 948.7 | 1199.03 | 1159.10 | 1165.15 | 1171.23 | 260.2 | 260.8 |
| 18 | 5 | 995.74 | 2134.9 | 3164.1 | 984.23 | 942.82 | 952.20 | 960.13 | 598.0 | 619.3 |
| 19 | 1 | 524.61 | 253.6 | 1005.2 | 524.61 | 524.61 | 527.29 | 533.00 | 8.0 | 11.3 |
| 19 | 2 | 826.00 | 64.5 | 547.2 | 819.80 | 792.79 | 796.69 | 802.93 | 59.8 | 62.6 |
| 19 | 3 | 825.01 | 321.7 | 942.9 | 839.19 | 801.13 | 804.35 | 809.69 | 101.7 | 103.7 |
| 19 | 4 | 866.91 | 757.8 | 1511.5 | 854.50 | 825.35 | 834.09 | 844.06 | 120.8 | 122.1 |
| 19 | 5 | 692.00 | 428.5 | 3600.5 | 690.26 | 678.46 | 682.75 | 691.37 | 215.4 | 227.5 |
| 20 | 1 | 241.97 | 325.0 | 2978.7 | 244.54 | 241.97 | 242.16 | 242.44 | 51.4 | 79.0 |
| 20 | 2 | 619.26 | 155.6 | 3072.6 | 576.24 | 554.17 | 560.09 | 565.27 | 1365.5 | 1378.9 |
| 20 | 3 | 581.94 | 674.5 | 3600.0 | 583.17 | 548.59 | 553.04 | 556.17 | 1042.3 | 1042.9 |
| 20 | 4 | 614.05 | 627.1 | 3600.0 | 614.73 | 562.28 | 568.26 | 573.54 | 519.4 | 519.9 |
| 20 | 5 | 512.00 | 2262.1 | 3600.1 | 532.17 | 495.07 | 497.99 | 501.92 | 1194.6 | 1204.0 |
| 21 | 1 | 688.18 | 2070.7 | 3600.1 | 687.60 | 690.20 | 691.94 | 694.91 | 26.8 | 48.1 |
| 21 | 2 | 1183.37 | 2292.1 | 3405.7 | 1137.92 | 1084.41 | 1099.62 | 1134.73 | 2356.0 | 2465.7 |
| 21 | 3 | 1197.72 | 1512.3 | 3600.0 | 1267.29 | 1166.59 | 1179.45 | 1188.99 | 357.3 | 358.0 |
| 21 | 4 | 1078.30 | 2486.7 | 3600.0 | 1070.05 | 1022.59 | 1039.76 | 1050.40 | 572.3 | 573.1 |
| 21 | 5 | 981.38 | 149.1 | 3600.1 | 950.06 | 924.82 | 934.29 | 946.21 | 1042.1 | 1052.4 |
| 22 | 1 | 740.66 | 2210.1 | 3600.0 | 740.66 | 742.91 | 746.45 | 752.52 | 57.4 | 74.0 |
| 22 | 2 | 1141.28 | 440.5 | 2909.1 | 1143.24 | 1085.44 | 1091.15 | 1108.89 | 865.8 | 889.9 |
| 22 | 3 | 1162.24 | 323.0 | 3600.0 | 1168.11 | 1122.11 | 1130.74 | 1137.56 | 414.0 | 415.0 |
| 22 | 4 | 1195.72 | 1480.8 | 3600.1 | 1169.14 | 1113.97 | 1128.14 | 1137.39 | 511.6 | 512.1 |
| 22 | 5 | 1022.06 | 3414.9 | 3600.0 | 1033.96 | 984.62 | 993.03 | 1012.74 | 1128.5 | 1137.7 |
| 23 | 1 | 860.47 | 866.9 | 3600.0 | 839.07 | 845.34 | 853.09 | 861.51 | 56.0 | 72.0 |
| 23 | 2 | 1335.10 | 619.2 | 3600.0 | 1212.71 | 1113.05 | 1139.46 | 1165.44 | 2806.1 | 2836.6 |
| 23 | 3 | 1189.67 | 629.7 | 3600.1 | 1193.57 | 1136.88 | 1152.51 | 1159.26 | 404.5 | 404.8 |
| 23 | 4 | 1170.37 | 332.4 | 3600.0 | 1185.41 | 1099.29 | 1124.71 | 1144.55 | 741.1 | 741.2 |
| 23 | 5 | 1050.31 | 930.7 | 3600.0 | 1013.28 | 982.99 | 993.36 | 999.65 | 725.6 | 738.2 |
| 24 | 1 | 1048.91 | 2371.0 | 3598.8 | 1035.33 | 1030.25 | 1042.26 | 1054.78 | 49.6 | 58.7 |
| 24 | 2 | 1382.85 | 1487.2 | 2224.0 | 1325.92 | 1240.46 | 1248.42 | 1269.76 | 365.6 | 384.8 |
| 24 | 3 | 1202.90 | 3370.6 | 3600.0 | 1168.25 | 1126.82 | 1132.54 | 1139.13 | 199.9 | 202.5 |
| 24 | 4 | 1322.48 | 3449.2 | 3600.0 | 1238.44 | 1168.98 | 1179.85 | 1187.15 | 392.5 | 393.2 |
| 24 | 5 | 1085.48 | 2534.4 | 3600.1 | 1093.88 | 1080.34 | 1086.15 | 1093.45 | 255.7 | 263.9 |
| 25 | 1 | 830.26 | 3597.6 | 3600.3 | 829.45 | 830.82 | 833.88 | 837.46 | 172.6 | 240.4 |
| 25 | 2 | 1567.22 | 2299.0 | 3600.0 | 1542.71 | 1494.36 | 1500.77 | 1508.94 | 1871.8 | 1909.0 |
| 25 | 3 | 1495.01 | 2411.6 | 3600.0 | 1554.00 | 1455.32 | 1470.17 | 1480.81 | 3751.9 | 3752.5 |

| | | TS | | | GTS | ACO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
| 25 | 4 | 1542.21 | 1139.7 | 3600.0 | 1544.76 | 1481.09 | 1498.03 | 1515.56 | 1262.5 | 1263.5 |
| 25 | 5 | 1318.08 | 2234.5 | 3600.1 | 1274.63 | 1237.68 | 1250.09 | 1259.98 | 3245.6 | 3276.1 |
| 26 | 1 | 819.56 | 355.9 | 3600.1 | 819.56 | 819.56 | 819.56 | 819.56 | 174.2 | 275.1 |
| 26 | 2 | 1427.04 | 1296.6 | 3600.0 | 1406.07 | 1337.65 | 1349.67 | 1361.24 | 1988.0 | 2058.7 |
| 26 | 3 | 1447.94 | 2781.5 | 3600.0 | 1499.53 | 1408.14 | 1415.38 | 1423.64 | 1258.5 | 1258.9 |
| 26 | 4 | 1594.35 | 2099.3 | 3600.1 | 1674.14 | 1548.79 | 1585.84 | 1610.44 | 9043.3 | 10802.3 |
| 26 | 5 | 1417.61 | 1239.9 | 3601.5 | 1324.11 | 1302.01 | 1308.73 | 1319.65 | 2195.8 | 2234.7 |
| 27 | 1 | 1099.95 | 985.2 | 3600.0 | 1097.63 | 1100.22 | 1104.42 | 1112.03 | 191.8 | 257.4 |
| 27 | 2 | 1610.60 | 3017.9 | 3600.0 | 1512.41 | 1386.89 | 1405.72 | 1417.68 | 2560.2 | 2612.1 |
| 27 | 3 | 1550.71 | 1883.2 | 3600.0 | 1518.69 | 1455.60 | 1462.74 | 1467.67 | 798.5 | 798.9 |
| 27 | 4 | 1540.83 | 66.9 | 3600.1 | 1448.80 | 1407.04 | 1412.83 | 1419.92 | 1034.2 | 1034.7 |
| 27 | 5 | 1394.74 | 588.1 | 3600.1 | 1373.48 | 1310.95 | 1330.65 | 1347.00 | 2070.5 | 2071.3 |
| 28 | 1 | 1078.27 | 3080.0 | 3600.0 | 1042.12 | 1050.49 | 1109.78 | 1176.84 | 265.7 | 351.2 |
| 28 | 2 | 2872.77 | 3375.3 | 3600.0 | 2822.69 | 2676.29 | 2723.66 | 2783.93 | 9518.2 | 10158.3 |
| 28 | 3 | 2929.11 | 2276.0 | 3600.0 | 2954.63 | 2787.15 | 2836.01 | 2882.74 | 10293.6 | 10741.2 |
| 28 | 4 | 2931.98 | 1809.0 | 3600.1 | 2928.88 | 2773.42 | 2809.70 | 2831.51 | 10298.5 | 10800.5 |
| 28 | 5 | 2700.27 | 2344.4 | 3600.4 | 2635.00 | 2486.69 | 2507.20 | 2527.24 | 9928.7 | 10801.2 |
| 29 | 1 | 1179.01 | 1834.5 | 3600.8 | 1188.15 | 1173.46 | 1214.80 | 1271.35 | 830.4 | 949.0 |
| 29 | 2 | 2658.72 | 30.3 | 3600.0 | 2518.99 | 2305.05 | 2369.35 | 2433.29 | 10642.2 | 10800.4 |
| 29 | 3 | 2390.00 | 227.1 | 3600.0 | 2318.45 | 2217.50 | 2239.83 | 2269.93 | 10238.8 | 10800.6 |
| 29 | 4 | 2732.84 | 208.6 | 3600.1 | 2590.22 | 2392.67 | 2418.39 | 2452.27 | 10372.4 | 10800.6 |
| 29 | 5 | 2519.56 | 3512.2 | 3600.0 | 2415.33 | 2251.33 | 2302.96 | 2330.41 | 10455.0 | 10801.5 |
| 30 | 1 | 1061.55 | 288.8 | 3601.0 | 1037.05 | 1037.71 | 1047.84 | 1062.37 | 320.2 | 402.2 |
| 30 | 2 | 2087.30 | 3475.1 | 3600.0 | 2002.71 | 1901.93 | 1908.54 | 1917.43 | 9899.0 | 10645.2 |
| 30 | 3 | 2194.36 | 1648.2 | 3600.0 | 2304.98 | 1969.50 | 2000.76 | 2066.34 | 10575.9 | 10800.4 |
| 30 | 4 | 2164.01 | 2955.7 | 3600.0 | 2139.16 | 1999.67 | 2032.69 | 2064.98 | 10726.0 | 10800.6 |
| 30 | 5 | 1859.14 | 1739.1 | 3600.0 | 1680.91 | 1641.87 | 1656.56 | 1694.41 | 10107.7 | 10800.4 |
| 31 | 1 | 1464.04 | 1780.8 | 3600.1 | 1421.20 | 1341.89 | 1368.25 | 1411.77 | 559.1 | 559.1 |
| 31 | 2 | 2574.21 | 2247.1 | 3600.1 | 2542.41 | 2399.56 | 2415.28 | 2445.75 | 10636.8 | 10800.3 |
| 31 | 3 | 2592.07 | 3190.0 | 3600.1 | 2644.98 | 2418.74 | 2450.71 | 2484.87 | 10394.8 | 10800.4 |
| 31 | 4 | 2808.08 | 2914.5 | 3600.0 | 2759.44 | 2593.34 | 2620.92 | 2656.75 | 10650.2 | 10800.4 |
| 31 | 5 | 2394.31 | 378.4 | 3602.3 | 2270.65 | 2153.90 | 2179.27 | 2210.04 | 10732.5 | 10801.1 |
| 32 | 1 | 1352.61 | 2531.7 | 3601.1 | 1328.68 | 1334.26 | 1361.25 | 1404.35 | 520.8 | 525.9 |
| 32 | 2 | 2652.73 | 1716.6 | 3600.1 | 2537.87 | 2391.85 | 2427.99 | 2462.58 | 10685.7 | 10800.4 |
| 32 | 3 | 2576.98 | 1929.3 | 3600.0 | 2521.68 | 2390.40 | 2410.49 | 2436.61 | 10508.1 | 10800.5 |
| 32 | 4 | 2816.35 | 2596.4 | 3600.0 | 2603.47 | 2472.82 | 2535.80 | 2606.34 | 10619.3 | 10800.4 |
| 32 | 5 | 2374.33 | 2752.2 | 3601.5 | 2186.76 | 2092.42 | 2135.05 | 2171.02 | 10402.6 | 10800.8 |
| 33 | 1 | 1361.51 | 788.6 | 3600.6 | 1328.19 | 1331.69 | 1356.13 | 1396.88 | 497.0 | 517.3 |
| 33 | 2 | 2605.16 | 1948.9 | 3600.0 | 2572.98 | 2419.32 | 2440.31 | 2485.24 | 10685.3 | 10800.2 |
| 33 | 3 | 2687.43 | 2429.5 | 3600.0 | 2677.29 | 2514.74 | 2539.99 | 2561.15 | 10601.7 | 10800.4 |
| 33 | 4 | 2851.39 | 3253.7 | 3600.3 | 2811.12 | 2600.70 | 2648.80 | 2685.83 | 10599.1 | 10800.9 |
| 33 | 5 | 2298.21 | 2685.3 | 3602.1 | 2200.25 | 2144.57 | 2168.85 | 2205.48 | 10639.1 | 10800.6 |
| 34 | 1 | 858.94 | 1941.9 | 3604.4 | 719.91 | 712.32 | 715.72 | 727.38 | 620.1 | 647.6 |
| 34 | 2 | 1573.77 | 2103.2 | 3600.0 | 1302.54 | 1253.69 | 1267.35 | 1287.94 | 10695.2 | 10800.4 |

| | | TS | | | GTS | ACO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
| 34 | 3 | 1641.66 | 861.0 | 3600.0 | 1379.90 | 1284.88 | 1303.83 | 1321.36 | 10747.0 | 10800.2 |
| 34 | 4 | 1586.18 | 3586.8 | 3600.5 | 1374.86 | 1298.07 | 1320.11 | 1351.96 | 10722.5 | 10800.3 |
| 34 | 5 | 1382.64 | 2429.2 | 3600.2 | 1156.65 | 1112.59 | 1122.13 | 1132.95 | 10665.9 | 10800.8 |
| 35 | 1 | 992.86 | 766.7 | 3600.3 | 877.04 | 868.12 | 879.56 | 886.01 | 1507.2 | 1533.5 |
| 35 | 2 | 1846.79 | 2028.0 | 3600.1 | 1564.85 | 1484.93 | 1503.65 | 1525.69 | 10726.3 | 10800.4 |
| 35 | 3 | 1852.52 | 2263.6 | 3600.0 | 1671.31 | 1564.16 | 1577.98 | 1599.27 | 10800.4 | 10800.4 |
| 35 | 4 | 2743.16 | 3271.3 | 3600.1 | 2072.36 | 1809.08 | 1888.15 | 1972.53 | 10801.4 | 10801.4 |
| 35 | 5 | 1499.30 | 2785.9 | 3600.5 | 1392.88 | 1350.52 | 1365.23 | 1385.08 | 10800.5 | 10800.5 |
| 36 | 1 | 678.87 | 1530.9 | 3603.9 | 594.10 | 617.91 | 626.43 | 634.53 | 2654.6 | 3088.1 |
| 36 | 2 | 1994.16 | 2217.9 | 3600.0 | 1914.94 | 1833.09 | 1865.93 | 1896.31 | 10691.6 | 10800.5 |
| 36 | 3 | 2082.29 | 2987.0 | 3600.0 | 2004.45 | 1924.22 | 1943.49 | 1959.96 | 10598.4 | 10800.7 |
| 36 | 4 | 1954.92 | 2840.1 | 3600.1 | 1871.24 | 1830.06 | 1851.79 | 1868.57 | 10751.7 | 10800.3 |
| 36 | 5 | 1755.27 | 3555.4 | 3600.3 | 1668.30 | 1653.21 | 1666.94 | 1691.16 | 10721.1 | 10801.8 |
| AVG | | 1171.75 | 936.5 | 1817.0 | 1142.23 | 1096.88 | 1107.92 | 1120.70 | 2442.6 | 2513.0 |

Table A.1: Detailed Comparison of ACO, TS and GTS (No Detailed Runtimes are Available in Zachariadis, Tarantilis, and Kiranoudis (2007)) for 2|RO|L

| | | B&C | | | | TS | | | | | | ACO | | | | | | | Gap ACO-B&C | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | *opt* | $sec_h$ | $sec_{tot}$ | $z$ | $sec_h$ | $sec_{tot}$ | *%gap* | *opt* | *imp* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | *opt* | *imp* | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 1 | 273 | 1 | 3.8 | 3.9 | 273 | 0.0 | 2.4 | 0.00 | 1 | 0 | 273 | 273.0 | 273 | 0.0 | 0.5 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 1 | 2 | 285 | 1 | 51.4 | 68.8 | 285 | 0.2 | 4.6 | 0.00 | 1 | 0 | 285 | 289.5 | 291 | 5.4 | 14.0 | 1 | 0 | 0.00 | 1.58 | 2.11 |
| 1 | 3 | 280 | 1 | 17.1 | 21.4 | 280 | 1.3 | 7.8 | 0.00 | 1 | 0 | 298 | 299.8 | 300 | 9.8 | 29.7 | 0 | 0 | 6.43 | 7.07 | 7.14 |
| 1 | 4 | 288 | 1 | 1.8 | 2.4 | 290 | 0.3 | 7.5 | 0.69 | 0 | 0 | 290 | 290.6 | 292 | 0.4 | 1.3 | 0 | 0 | 0.69 | 0.90 | 1.39 |
| 1 | 5 | 279 | 1 | 53.1 | 53.1 | 279 | 2.3 | 15.7 | 0.00 | 1 | 0 | 279 | 279.0 | 279 | 0.4 | 2.0 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 2 | 1 | 329 | 1 | 0.2 | 0.5 | 329 | 0.1 | 1.4 | 0.00 | 1 | 0 | 329 | 329.0 | 329 | 0.0 | 0.4 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 2 | 2 | 342 | 1 | 11.1 | 11.9 | 342 | 1.1 | 2.0 | 0.00 | 1 | 0 | 342 | 342.0 | 342 | 0.1 | 0.5 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 2 | 3 | 347 | 1 | 6.4 | 8.1 | 350 | 0.2 | 3.2 | 0.86 | 0 | 0 | 350 | 351.8 | 356 | 0.8 | 1.9 | 0 | 0 | 0.86 | 1.38 | 2.59 |
| 2 | 4 | 336 | 1 | 22.4 | 23.4 | 336 | 0.1 | 6.7 | 0.00 | 1 | 0 | 336 | 336.6 | 339 | 0.2 | 0.6 | 1 | 0 | 0.00 | 0.18 | 0.89 |
| 2 | 5 | 329 | 1 | 29.2 | 29.4 | 329 | 0.3 | 6.1 | 0.00 | 1 | 0 | 329 | 329.0 | 329 | 0.2 | 0.5 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 3 | 1 | 351 | 1 | 14.9 | 15.6 | 351 | 0.2 | 8.4 | 0.00 | 1 | 0 | 351 | 351.0 | 351 | 0.1 | 1.1 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 3 | 2 | 396 | 1 | 32.6 | 65.8 | 407 | 8.9 | 9.8 | 2.78 | 0 | 0 | 396 | 396.0 | 396 | 0.9 | 6.9 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 3 | 3 | 387 | 1 | 6.0 | 6.1 | 387 | 1.0 | 20.0 | 0.00 | 1 | 0 | 387 | 391.2 | 393 | 11.3 | 17.5 | 1 | 0 | 0.00 | 1.09 | 1.55 |
| 3 | 4 | 374 | 1 | 39.3 | 39.4 | 374 | 14.3 | 21.9 | 0.00 | 1 | 0 | 374 | 374.5 | 377 | 2.4 | 3.8 | 1 | 0 | 0.00 | 0.13 | 0.80 |
| 3 | 5 | 369 | 1 | 0.2 | 0.2 | 369 | 1.0 | 29.9 | 0.00 | 1 | 0 | 369 | 369.0 | 369 | 0.1 | 1.8 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 4 | 1 | 423 | 1 | 0.4 | 0.4 | 423 | 0.2 | 5.7 | 0.00 | 1 | 0 | 423 | 423.0 | 423 | 0.1 | 0.9 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 4 | 2 | 434 | 1 | 5.3 | 5.5 | 434 | 0.3 | 7.4 | 0.00 | 1 | 0 | 434 | 434.0 | 434 | 0.5 | 3.0 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 4 | 3 | 432 | 1 | 5.2 | 8.2 | 438 | 5.0 | 12.8 | 1.39 | 0 | 0 | 437 | 437.0 | 437 | 3.9 | 3.9 | 0 | 0 | 1.16 | 1.16 | 1.16 |
| 4 | 4 | 438 | 1 | 23.9 | 26.4 | 451 | 0.8 | 16.9 | 2.97 | 0 | 0 | 438 | 438.1 | 439 | 4.9 | 8.7 | 1 | 0 | 0.00 | 0.02 | 0.23 |
| 4 | 5 | 423 | 1 | 23.0 | 44.7 | 423 | 1.2 | 27.1 | 0.00 | 1 | 0 | 423 | 423.0 | 423 | 1.4 | 3.2 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 5 | 1 | 367 | 1 | 0.1 | 0.1 | 367 | 2.9 | 12.8 | 0.00 | 1 | 0 | 367 | 367.0 | 367 | 0.0 | 1.4 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 5 | 2 | 380 | 1 | 8.0 | 8.4 | 396 | 4.0 | 19.7 | 4.21 | 0 | 0 | 380 | 380.0 | 380 | 2.1 | 18.3 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 5 | 3 | 373 | 1 | 1.8 | 1.9 | 377 | 0.6 | 20.5 | 1.07 | 0 | 0 | 373 | 375.0 | 377 | 11.4 | 23.7 | 1 | 0 | 0.00 | 0.54 | 1.07 |
| 5 | 4 | 377 | 1 | 50.2 | 50.3 | 406 | 5.9 | 33.0 | 7.69 | 0 | 0 | 377 | 379.4 | 383 | 33.4 | 45.8 | 1 | 0 | 0.00 | 0.64 | 1.59 |
| 5 | 5 | 389 | 1 | 2928.2 | 2928.3 | 389 | 5.7 | 57.8 | 0.00 | 1 | 0 | 389 | 389.0 | 389 | 0.7 | 5.7 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 6 | 1 | 488 | 1 | 5.9 | 10.7 | 488 | 0.1 | 10.3 | 0.00 | 1 | 0 | 488 | 488.0 | 488 | 0.1 | 1.2 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 6 | 2 | 491 | 1 | 145.8 | 145.9 | 498 | 1.8 | 14.0 | 1.43 | 0 | 0 | 491 | 494.2 | 495 | 0.5 | 3.0 | 1 | 0 | 0.00 | 0.65 | 0.81 |
| 6 | 3 | 496 | 1 | 135.4 | 150.4 | 496 | 11.2 | 16.9 | 0.00 | 1 | 0 | 496 | 496.6 | 500 | 3.7 | 6.6 | 1 | 0 | 0.00 | 0.12 | 0.81 |
| 6 | 4 | 489 | 1 | 14.1 | 16.6 | 503 | 0.1 | 40.2 | 2.86 | 0 | 0 | 490 | 494.7 | 503 | 9.2 | 13.3 | 0 | 0 | 0.20 | 1.17 | 2.86 |
| 6 | 5 | 488 | 1 | 10.8 | 13.3 | 488 | 0.7 | 34.2 | 0.00 | 1 | 0 | 488 | 488.0 | 488 | 0.4 | 3.3 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 7 | 1 | 558 | 1 | 0.0 | 0.0 | 558 | 0.3 | 22.0 | 0.00 | 1 | 0 | 558 | 558.0 | 558 | 0.0 | 1.6 | 1 | 0 | 0.00 | 0.00 | 0.00 |

| | | B&C | | | | TS | | | | | | ACO | | | | | | | Gap ACO-B&C | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | *opt* | $sec_h$ | $sec_{tot}$ | $z$ | $sec_h$ | $sec_{tot}$ | *%gap* | *opt* | *imp* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | *opt* | *imp* | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 7 | 2 | 724 | 1 | 32.0 | 32.5 | 752 | 0.7 | 18.9 | 3.87 | 0 | 0 | 724 | 724.0 | 724 | 2.4 | 10.4 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 7 | 3 | 698 | 1 | 3.2 | 4.4 | 704 | 19.9 | 29.8 | 0.86 | 0 | 0 | 698 | 704.7 | 718 | 4.3 | 10.3 | 1 | 0 | 0.00 | 0.96 | 2.87 |
| 7 | 4 | 714 | 1 | 2596.8 | 2597.3 | 742 | 26.5 | 50.4 | 3.92 | 0 | 0 | 722 | 722.0 | 722 | 3.7 | 12.8 | 0 | 0 | 1.12 | 1.12 | 1.12 |
| 7 | 5 | 742 | 1 | 738.9 | 747.2 | 743 | 16.1 | 75.1 | 0.13 | 0 | 0 | 743 | 744.5 | 746 | 6.6 | 14.3 | 0 | 0 | 0.13 | 0.34 | 0.54 |
| 8 | 1 | 657 | 1 | 0.0 | 0.0 | 657 | 7.0 | 31.3 | 0.00 | 1 | 0 | 657 | 657.0 | 657 | 0.1 | 1.6 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 8 | 2 | 720 | 1 | 75.9 | 91.8 | 720 | 3.1 | 20.1 | 0.00 | 1 | 0 | 720 | 720.0 | 720 | 1.4 | 8.7 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 8 | 3 | 730 | 1 | 70.0 | 73.0 | 752 | 20.5 | 33.4 | 3.01 | 0 | 0 | 730 | 730.3 | 733 | 13.7 | 13.7 | 1 | 0 | 0.00 | 0.04 | 0.41 |
| 8 | 4 | 701 | 1 | 7.4 | 14.1 | 722 | 11.2 | 50.0 | 3.00 | 0 | 0 | 715 | 721.7 | 724 | 5.0 | 11.6 | 0 | 0 | 2.00 | 2.95 | 3.28 |
| 8 | 5 | 721 | 1 | 1128.9 | 1128.9 | 736 | 12.0 | 90.2 | 2.08 | 0 | 0 | 721 | 724.8 | 730 | 21.0 | 40.2 | 1 | 0 | 0.00 | 0.53 | 1.25 |
| 9 | 1 | 609 | 1 | 6.2 | 31.9 | 609 | 1.9 | 11.9 | 0.00 | 1 | 0 | 609 | 609.0 | 609 | 0.4 | 1.7 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 9 | 2 | 612 | 1 | 453.5 | 460.5 | 612 | 2.9 | 15.4 | 0.00 | 1 | 0 | 612 | 616.5 | 627 | 2.8 | 5.3 | 1 | 0 | 0.00 | 0.74 | 2.45 |
| 9 | 3 | 615 | 1 | 164.8 | 194.3 | 626 | 7.9 | 38.4 | 1.79 | 0 | 0 | 615 | 619.0 | 633 | 15.9 | 16.1 | 1 | 0 | 0.00 | 0.65 | 2.93 |
| 9 | 4 | 626 | 1 | 852.1 | 1593.3 | 627 | 5.9 | 43.5 | 0.16 | 0 | 0 | 627 | 628.4 | 635 | 9.9 | 13.0 | 0 | 0 | 0.16 | 0.38 | 1.44 |
| 9 | 5 | 609 | 1 | 47.4 | 69.2 | 609 | 8.5 | 81.9 | 0.00 | 1 | 0 | 609 | 612.0 | 624 | 3.6 | 5.0 | 1 | 0 | 0.00 | 0.49 | 2.46 |
| 10 | 1 | 524 | 1 | 1226.2 | 83249.5 | 544 | 19.9 | 97.3 | 3.82 | 0 | 0 | 524 | 524.0 | 524 | 0.9 | 9.9 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 10 | 2 | 774 | 0 | 48373.3 | 86401.1 | 703 | 3.8 | 72.2 | -9.17 | 0 | 1 | 687 | 687.0 | 687 | 20.8 | 41.7 | 0 | 1 | -11.24 | -11.24 | -11.24 |
| 10 | 3 | 637 | 1 | 2304.5 | 3150.0 | 676 | 47.9 | 118.7 | 6.12 | 0 | 0 | 638 | 655.5 | 663 | 30.5 | 45.6 | 0 | 0 | 0.16 | 2.90 | 4.08 |
| 10 | 4 | 738 | 1 | 12671.3 | 12696.3 | 773 | 47.3 | 156.9 | 4.74 | 0 | 0 | 761 | 762.9 | 768 | 44.8 | 45.4 | 0 | 0 | 3.12 | 3.37 | 4.07 |
| 10 | 5 | 706 | 0 | 48220.2 | 70308.0 | 724 | 84.5 | 308.9 | 2.55 | 0 | 0 | 710 | 715.9 | 722 | 44.3 | 76.8 | 0 | 0 | 0.57 | 1.40 | 2.27 |
| 11 | 1 | 500 | 1 | 0.1 | 0.1 | 500 | 0.8 | 107.8 | 0.00 | 1 | 0 | 500 | 500.0 | 500 | 0.1 | 5.9 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 11 | 2 | 789 | 0 | 99.8 | 86400.5 | 734 | 4.7 | 72.2 | -6.97 | 0 | 1 | 708 | 709.2 | 710 | 101.8 | 140.0 | 0 | 1 | -10.27 | -10.11 | -10.01 |
| 11 | 3 | 763 | 0 | 63747.8 | 86400.3 | 785 | 72.2 | 101.7 | 2.88 | 0 | 0 | 743 | 743.0 | 743 | 27.1 | 32.1 | 0 | 1 | -2.62 | -2.62 | -2.62 |
| 11 | 4 | 881 | 0 | 85181.5 | 86400.8 | 877 | 196.4 | 209.5 | -0.45 | 0 | 1 | 800 | 813.7 | 832 | 293.7 | 294.2 | 0 | 1 | -9.19 | -7.64 | -5.56 |
| 11 | 5 | 695 | 0 | 64392.9 | 86400.1 | 696 | 279.2 | 387.0 | 0.14 | 0 | 0 | 689 | 694.9 | 699 | 71.6 | 78.8 | 0 | 1 | -0.86 | -0.01 | 0.58 |
| 12 | 1 | 599 | 0 | 50093.5 | 86400.3 | 598 | 19.4 | 33.8 | -0.17 | 0 | 1 | 596 | 598.2 | 600 | 1.2 | 3.3 | 0 | 1 | -0.50 | -0.13 | 0.17 |
| 12 | 2 | 625 | 0 | 75.6 | 86400.7 | 628 | 9.6 | 42.9 | 0.48 | 0 | 0 | 605 | 607.7 | 614 | 7.5 | 10.8 | 0 | 1 | -3.20 | -2.77 | -1.76 |
| 12 | 3 | 597 | 0 | 36171.0 | 86400.6 | 597 | 12.9 | 50.7 | 0.00 | 0 | 0 | 597 | 598.1 | 602 | 2.0 | 4.6 | 0 | 0 | 0.00 | 0.18 | 0.84 |
| 12 | 4 | 624 | 0 | 86321.6 | 86400.2 | 640 | 96.0 | 120.6 | 2.56 | 0 | 0 | 608 | 608.7 | 615 | 23.7 | 24.2 | 0 | 1 | -2.56 | -2.45 | -1.44 |
| 12 | 5 | 602 | 0 | 22352.4 | 86400.4 | 597 | 103.5 | 188.2 | -0.83 | 0 | 1 | 597 | 601.2 | 606 | 6.8 | 9.6 | 0 | 1 | -0.83 | -0.13 | 0.66 |
| 13 | 1 | 1991 | 1 | 13.2 | 14.5 | 1991 | 47.4 | 218.7 | 0.00 | 1 | 0 | 1991 | 1991.0 | 1991 | 0.1 | 5.3 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 13 | 2 | 3523 | 0 | 10784.1 | 86400.3 | 2775 | 43.7 | 123.1 | -21.23 | 0 | 1 | 2714 | 2714.0 | 2714 | 13.8 | 32.6 | 0 | 1 | -22.96 | -22.96 | -22.96 |

| | | B&C | | | | TS | | | | | | ACO | | | | | | | Gap ACO-B&C | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | *opt* | $sec_h$ | $sec_{tot}$ | $z$ | $sec_h$ | $sec_{tot}$ | %*gap* | *opt* | *imp* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ | *opt* | *imp* | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 13 | 3 | 2574 | 1 | 3087.9 | 32124.5 | 2696 | 121.5 | 170.3 | 4.73 | 0 | 0 | 2574 | 2576.7 | 2587 | 54.5 | 57.0 | 1 | 0 | 0.00 | 0.10 | 0.51 |
| 13 | 4 | 2673 | 0 | 34999.2 | 86400.4 | 2743 | 29.3 | 277.0 | 2.62 | 0 | 0 | 2671 | 2694.4 | 2708 | 56.6 | 56.6 | 0 | 1 | -0.07 | 0.80 | 1.31 |
| 13 | 5 | 2807 | 0 | 83628.4 | 86400.1 | 2737 | 237.5 | 691.9 | -2.49 | 0 | 1 | 2710 | 2734.7 | 2748 | 62.0 | 91.6 | 0 | 1 | -3.46 | -2.58 | -2.10 |
| 14 | 1 | 827 | 0 | 25734.8 | 86400.4 | 823 | 21.8 | 145.0 | -0.48 | 0 | 1 | 823 | 823.0 | 823 | 2.3 | 17.7 | 0 | 1 | -0.48 | -0.48 | -0.48 |
| 14 | 2 | 1459 | 0 | 80521.1 | 86400.8 | 1266 | 52.3 | 144.4 | -13.23 | 0 | 1 | 1241 | 1242.1 | 1244 | 57.0 | 91.7 | 0 | 1 | -14.94 | -14.87 | -14.74 |
| 14 | 3 | 1211 | 0 | 42204.7 | 86400.1 | 1204 | 137.9 | 207.1 | -0.58 | 0 | 1 | 1190 | 1197.6 | 1200 | 73.7 | 98.0 | 0 | 1 | -1.73 | -1.11 | -0.91 |
| 14 | 4 | 1166 | 1 | 25391.4 | 25542.6 | 1187 | 108.5 | 324.9 | 1.80 | 0 | 0 | 1177 | 1179.1 | 1185 | 126.9 | 131.7 | 0 | 0 | 0.94 | 1.12 | 1.63 |
| 14 | 5 | 1504 | 0 | 80511.1 | 86400.1 | 1309 | 103.2 | 895.0 | -12.97 | 0 | 1 | 1288 | 1288.5 | 1292 | 107.9 | 172.1 | 0 | 1 | -14.36 | -14.33 | -14.10 |
| 15 | 1 | 907 | 1 | 17.8 | 17.8 | 907 | 51.9 | 196.4 | 0.00 | 1 | 0 | 907 | 907.0 | 907 | 0.2 | 9.5 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 15 | 2 | 1203 | 0 | 386.9 | 86401.4 | 1135 | 94.7 | 133.9 | -5.65 | 0 | 1 | 1101 | 1101.0 | 1101 | 54.5 | 100.3 | 0 | 1 | -8.48 | -8.48 | -8.48 |
| 15 | 3 | 1405 | 0 | 15880.7 | 86401.2 | 1183 | 37.1 | 205.9 | -15.80 | 0 | 1 | 1170 | 1179.8 | 1185 | 418.1 | 553.7 | 0 | 1 | -16.73 | -16.03 | -15.66 |
| 15 | 4 | 1358 | 0 | 55614.5 | 86400.3 | 1372 | 268.2 | 332.2 | 1.03 | 0 | 0 | 1353 | 1364.8 | 1371 | 84.5 | 102.0 | 0 | 1 | -0.37 | 0.50 | 0.96 |
| 15 | 5 | 1390 | 0 | 59867.8 | 86400.1 | 1361 | 651.4 | 671.7 | -2.09 | 0 | 1 | 1361 | 1361.4 | 1362 | 108.9 | 193.8 | 0 | 1 | -2.09 | -2.06 | -2.01 |
| 16 | 1 | 682 | 1 | 7086.5 | 7086.7 | 682 | 56.1 | 96.6 | 0.00 | 1 | 0 | 682 | 682.0 | 682 | 1.9 | 5.2 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 16 | 2 | 682 | 1 | 1767.0 | 3374.8 | 682 | 20.5 | 91.7 | 0.00 | 1 | 0 | 682 | 685.6 | 688 | 5.7 | 8.5 | 1 | 0 | 0.00 | 0.53 | 0.88 |
| 16 | 3 | 682 | 1 | 345.1 | 3853.3 | 682 | 15.3 | 138.1 | 0.00 | 1 | 0 | 682 | 683.8 | 691 | 7.9 | 11.5 | 1 | 0 | 0.00 | 0.26 | 1.32 |
| 16 | 4 | 691 | 1 | 33375.9 | 33391.8 | 704 | 21.7 | 206.1 | 1.88 | 0 | 0 | 693 | 693.0 | 693 | 13.9 | 15.7 | 0 | 0 | 0.29 | 0.29 | 0.29 |
| 16 | 5 | 682 | 1 | 339.0 | 2784.0 | 682 | 62.8 | 276.6 | 0.00 | 1 | 0 | 682 | 682.0 | 682 | 4.6 | 7.8 | 1 | 0 | 0.00 | 0.00 | 0.00 |
| 17 | 1 | 859 | 0 | 493.1 | 86400.3 | 842 | 84.4 | 158.6 | -1.98 | 0 | 1 | 842 | 843.8 | 846 | 2.3 | 8.3 | 0 | 1 | -1.98 | -1.77 | -1.51 |
| 17 | 2 | 866 | 0 | 283.8 | 86400.4 | 851 | 56.6 | 132.5 | -1.73 | 0 | 1 | 851 | 851.6 | 857 | 4.6 | 9.4 | 0 | 1 | -1.73 | -1.66 | -1.04 |
| 17 | 3 | 850 | 0 | 656.0 | 86400.0 | 842 | 38.8 | 200.6 | -0.94 | 0 | 1 | 842 | 844.9 | 850 | 4.2 | 9.1 | 0 | 1 | -0.94 | -0.60 | 0.00 |
| 17 | 4 | 853 | 0 | 102.9 | 86400.4 | 845 | 7.4 | 279.7 | -0.94 | 0 | 1 | 842 | 842.9 | 845 | 3.3 | 10.5 | 0 | 1 | -1.29 | -1.18 | -0.94 |
| 17 | 5 | 845 | 0 | 263.6 | 86400.5 | 842 | 44.7 | 402.2 | -0.36 | 0 | 1 | 842 | 845.4 | 850 | 2.2 | 9.2 | 0 | 1 | -0.36 | 0.05 | 0.59 |
| AVG | | 767.4 | 58 | 12875.4 | 29821.2 | 754.6 | 42.7 | 116.2 | -0.21 | 33 | 19 | 743.7 | 746.2 | 749.3 | 25.2 | 35.6 | 45 | 25 | -1.36 | -1.05 | -0.63 |

Table A.2: Detailed Results for 2|RO|L with Euclidian Distances Rounded Down to the Next Integer

| | | TS | | | GTS | ACO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
| 1 | 1 | 278.73 | 2.1 | 2.5 | 278.73 | 278.73 | 278.73 | 278.73 | 0.1 | 0.4 |
| 1 | 2 | 284.42 | 1.3 | 4.9 | 305.92 | 284.52 | 284.72 | 285.50 | 1.3 | 2.7 |
| 1 | 3 | 306.31 | 10.0 | 12.7 | 299.70 | 296.87 | 297.15 | 299.70 | 10.2 | 11.5 |
| 1 | 4 | 296.94 | 2.0 | 9.9 | 296.75 | 282.95 | 282.95 | 282.95 | 0.9 | 0.9 |
| 1 | 5 | 278.73 | 5.6 | 18.5 | 280.60 | 278.73 | 279.85 | 280.60 | 0.3 | 0.8 |
| 2 | 1 | 334.96 | 0.0 | 1.5 | 334.96 | 334.96 | 334.96 | 334.96 | 0.1 | 0.3 |
| 2 | 2 | 334.96 | 0.1 | 2.0 | 334.96 | 334.96 | 334.96 | 334.96 | 0.2 | 0.4 |
| 2 | 3 | 352.16 | 0.1 | 3.0 | 355.65 | 352.16 | 352.16 | 352.16 | 0.6 | 0.8 |
| 2 | 4 | 342.00 | 0.0 | 5.7 | 342.00 | 342.00 | 342.00 | 342.00 | 0.1 | 0.4 |
| 2 | 5 | 334.96 | 0.1 | 6.2 | 334.96 | 334.96 | 334.96 | 334.96 | 0.1 | 0.4 |
| 3 | 1 | 359.77 | 3.7 | 9.6 | 358.40 | 358.40 | 358.40 | 358.40 | 0.2 | 0.8 |
| 3 | 2 | 387.70 | 0.6 | 11.8 | 401.81 | 387.70 | 387.70 | 387.70 | 1.1 | 2.1 |
| 3 | 3 | 394.72 | 0.8 | 23.6 | 409.17 | 394.72 | 394.72 | 394.72 | 1.6 | 2.7 |
| 3 | 4 | 368.56 | 0.8 | 23.1 | 368.56 | 364.45 | 364.45 | 364.45 | 0.6 | 1.4 |
| 3 | 5 | 358.40 | 2.0 | 29.8 | 358.40 | 358.40 | 358.40 | 358.40 | 0.3 | 1.0 |
| 4 | 1 | 430.88 | 0.1 | 6.0 | 430.88 | 430.89 | 430.89 | 430.89 | 0.3 | 0.7 |
| 4 | 2 | 430.88 | 0.8 | 9.4 | 440.94 | 430.89 | 430.89 | 430.89 | 1.0 | 1.9 |
| 4 | 3 | 440.68 | 0.4 | 14.9 | 446.61 | 445.49 | 445.49 | 445.49 | 1.5 | 2.2 |
| 4 | 4 | 447.37 | 0.3 | 15.7 | 447.37 | 447.37 | 447.37 | 447.37 | 2.8 | 3.3 |
| 4 | 5 | 430.88 | 1.1 | 27.7 | 430.88 | 430.89 | 430.89 | 430.89 | 0.7 | 1.3 |
| 5 | 1 | 375.28 | 1.4 | 13.1 | 375.28 | 375.28 | 375.28 | 375.28 | 0.3 | 1.0 |
| 5 | 2 | 379.94 | 9.6 | 19.6 | 381.85 | 375.28 | 375.28 | 375.28 | 6.3 | 10.2 |
| 5 | 3 | 381.69 | 10.4 | 19.1 | 387.89 | 381.69 | 381.69 | 381.69 | 11.8 | 14.7 |
| 5 | 4 | 383.87 | 0.3 | 24.6 | 383.87 | 383.88 | 384.24 | 387.54 | 7.8 | 11.9 |
| 5 | 5 | 375.28 | 3.6 | 49.6 | 375.28 | 375.28 | 375.28 | 375.28 | 1.7 | 2.9 |
| 6 | 1 | 495.85 | 0.3 | 9.6 | 495.85 | 495.85 | 495.85 | 495.85 | 0.3 | 0.9 |
| 6 | 2 | 498.16 | 5.8 | 12.1 | 498.16 | 495.85 | 495.85 | 495.85 | 1.7 | 2.6 |
| 6 | 3 | 504.68 | 4.1 | 15.9 | 499.08 | 498.16 | 498.89 | 499.08 | 4.3 | 5.1 |
| 6 | 4 | 505.38 | 25.4 | 30.3 | 504.78 | 498.32 | 498.38 | 498.65 | 2.4 | 4.3 |
| 6 | 5 | 495.85 | 0.3 | 26.1 | 495.85 | 495.85 | 495.85 | 495.85 | 0.9 | 1.4 |
| 7 | 1 | 568.56 | 0.5 | 26.5 | 568.56 | 568.56 | 568.56 | 568.56 | 0.2 | 1.3 |
| 7 | 2 | 725.46 | 0.9 | 20.0 | 741.91 | 725.46 | 725.46 | 725.46 | 3.9 | 6.4 |
| 7 | 3 | 702.59 | 14.7 | 38.0 | 706.99 | 701.08 | 702.03 | 702.99 | 2.8 | 6.3 |
| 7 | 4 | 703.85 | 7.0 | 49.2 | 703.85 | 702.45 | 702.45 | 702.45 | 5.2 | 6.7 |
| 7 | 5 | 669.47 | 8.3 | 108.5 | 661.22 | 658.64 | 659.52 | 661.22 | 5.0 | 8.2 |
| 8 | 1 | 568.56 | 0.6 | 39.9 | 568.56 | 568.56 | 568.56 | 568.56 | 0.2 | 1.3 |
| 8 | 2 | 674.55 | 3.0 | 23.8 | 718.18 | 709.39 | 709.39 | 709.39 | 8.1 | 8.1 |
| 8 | 3 | 741.12 | 5.4 | 27.4 | 749.70 | 740.85 | 740.85 | 740.85 | 7.3 | 7.3 |
| 8 | 4 | 714.77 | 17.2 | 60.5 | 711.07 | 692.47 | 692.71 | 694.79 | 5.0 | 5.8 |
| 8 | 5 | 649.52 | 29.8 | 155.1 | 643.43 | 621.85 | 631.59 | 642.22 | 15.9 | 32.9 |
| 9 | 1 | 607.65 | 0.4 | 14.0 | 607.65 | 607.65 | 607.65 | 607.65 | 0.6 | 1.4 |
| 9 | 2 | 607.65 | 0.4 | 16.3 | 607.65 | 607.65 | 608.91 | 620.25 | 2.6 | 3.5 |
| 9 | 3 | 638.31 | 7.8 | 37.1 | 622.16 | 607.65 | 607.65 | 607.65 | 3.0 | 4.3 |
| 9 | 4 | 625.13 | 4.8 | 48.6 | 625.13 | 625.10 | 627.22 | 633.42 | 3.1 | 4.2 |

| | | TS | | | GTS | ACO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
| 9 | 5 | 607.65 | 4.6 | 90.2 | 607.65 | 607.65 | 607.65 | 607.65 | 0.7 | 2.0 |
| 10 | 1 | 538.79 | 6.6 | 87.3 | 535.80 | 535.80 | 535.80 | 535.80 | 2.3 | 4.2 |
| 10 | 2 | 701.25 | 43.2 | 92.4 | 708.63 | 689.68 | 690.97 | 691.57 | 25.3 | 26.6 |
| 10 | 3 | 640.19 | 22.9 | 166.7 | 655.70 | 624.62 | 632.98 | 642.27 | 28.4 | 31.3 |
| 10 | 4 | 761.66 | 49.2 | 201.6 | 792.30 | 724.77 | 743.66 | 754.24 | 24.0 | 27.6 |
| 10 | 5 | 698.47 | 58.3 | 411.8 | 695.37 | 692.66 | 693.76 | 697.40 | 20.4 | 27.5 |
| 11 | 1 | 505.01 | 3.0 | 115.3 | 505.01 | 505.01 | 505.01 | 505.01 | 0.8 | 3.1 |
| 11 | 2 | 722.58 | 20.2 | 95.8 | 719.56 | 711.08 | 711.08 | 711.08 | 14.9 | 20.8 |
| 11 | 3 | 715.81 | 5.1 | 117.8 | 746.12 | 721.66 | 722.73 | 723.00 | 19.2 | 19.2 |
| 11 | 4 | 859.18 | 175.9 | 216.5 | 843.52 | 816.45 | 817.00 | 817.24 | 18.9 | 19.2 |
| 11 | 5 | 658.60 | 74.2 | 492.5 | 652.42 | 636.77 | 636.77 | 636.77 | 16.8 | 21.3 |
| 12 | 1 | 610.57 | 32.6 | 37.3 | 610.00 | 610.00 | 611.22 | 614.24 | 1.5 | 2.4 |
| 12 | 2 | 627.43 | 12.0 | 46.1 | 628.86 | 610.57 | 612.34 | 620.47 | 3.3 | 4.9 |
| 12 | 3 | 610.23 | 20.5 | 52.8 | 610.00 | 610.00 | 612.45 | 615.61 | 2.0 | 3.0 |
| 12 | 4 | 630.59 | 80.0 | 103.9 | 618.23 | 614.24 | 614.73 | 619.21 | 3.8 | 6.3 |
| 12 | 5 | 614.23 | 100.1 | 173.2 | 610.23 | 614.24 | 614.98 | 617.45 | 3.2 | 4.2 |
| 13 | 1 | 2006.34 | 29.8 | 162.5 | 2006.34 | 2006.34 | 2006.34 | 2006.34 | 1.3 | 4.0 |
| 13 | 2 | 2616.19 | 9.1 | 155.1 | 2705.05 | 2588.81 | 2588.81 | 2588.81 | 39.0 | 39.0 |
| 13 | 3 | 2527.44 | 35.5 | 219.6 | 2542.86 | 2477.97 | 2477.97 | 2477.97 | 26.3 | 35.4 |
| 13 | 4 | 2725.50 | 64.5 | 281.5 | 2714.69 | 2616.95 | 2617.03 | 2617.79 | 30.7 | 30.7 |
| 13 | 5 | 2523.68 | 148.4 | 841.7 | 2434.99 | 2416.04 | 2435.14 | 2446.29 | 30.0 | 44.3 |
| 14 | 1 | 837.67 | 23.0 | 156.8 | 837.67 | 837.67 | 837.67 | 837.67 | 4.1 | 6.0 |
| 14 | 2 | 1079.51 | 34.9 | 205.9 | 1117.24 | 1038.68 | 1041.38 | 1047.36 | 70.0 | 70.5 |
| 14 | 3 | 1110.59 | 240.3 | 319.8 | 1092.10 | 1032.40 | 1043.42 | 1055.71 | 87.7 | 87.7 |
| 14 | 4 | 1014.75 | 331.5 | 462.5 | 994.66 | 985.01 | 989.29 | 990.60 | 63.0 | 67.2 |
| 14 | 5 | 986.02 | 1249.6 | 1683.1 | 943.08 | 922.58 | 922.58 | 922.58 | 72.6 | 85.2 |
| 15 | 1 | 837.67 | 2.2 | 204.6 | 837.67 | 837.67 | 837.67 | 837.67 | 2.8 | 4.3 |
| 15 | 2 | 1043.84 | 24.8 | 172.9 | 1099.75 | 1021.41 | 1044.65 | 1063.37 | 46.8 | 55.1 |
| 15 | 3 | 1173.83 | 7.8 | 167.0 | 1186.61 | 1171.35 | 1171.37 | 1171.42 | 66.2 | 66.2 |
| 15 | 4 | 1305.89 | 65.7 | 351.0 | 1258.49 | 1244.74 | 1254.46 | 1266.13 | 60.1 | 60.4 |
| 15 | 5 | 1281.97 | 683.3 | 983.9 | 1246.46 | 1230.22 | 1231.22 | 1234.51 | 56.4 | 60.7 |
| 16 | 1 | 698.61 | 3.0 | 102.8 | 698.61 | 698.61 | 698.61 | 698.61 | 2.0 | 3.9 |
| 16 | 2 | 698.61 | 39.9 | 91.4 | 702.70 | 698.61 | 701.37 | 704.08 | 5.9 | 7.6 |
| 16 | 3 | 698.61 | 35.4 | 129.1 | 698.61 | 698.61 | 698.61 | 698.61 | 6.0 | 7.3 |
| 16 | 4 | 712.30 | 20.1 | 180.7 | 709.27 | 703.35 | 705.19 | 707.93 | 8.8 | 8.8 |
| 16 | 5 | 698.61 | 4.1 | 296.7 | 698.61 | 698.61 | 698.61 | 698.61 | 2.5 | 5.1 |
| 17 | 1 | 862.62 | 62.7 | 173.4 | 863.27 | 861.79 | 862.37 | 862.62 | 3.3 | 6.2 |
| 17 | 2 | 873.85 | 33.2 | 139.5 | 870.86 | 870.86 | 872.80 | 875.44 | 5.4 | 7.7 |
| 17 | 3 | 862.62 | 168.6 | 195.5 | 861.79 | 862.62 | 863.50 | 867.85 | 4.0 | 6.2 |
| 17 | 4 | 866.40 | 1.9 | 263.1 | 861.79 | 861.79 | 862.37 | 862.62 | 3.2 | 7.4 |
| 17 | 5 | 862.62 | 58.3 | 319.5 | 862.62 | 861.79 | 862.12 | 862.62 | 3.5 | 7.1 |
| 18 | 1 | 723.54 | 85.5 | 620.9 | 730.85 | 723.54 | 726.18 | 727.74 | 9.1 | 14.5 |
| 18 | 2 | 1087.09 | 2.7 | 651.5 | 1065.30 | 1029.26 | 1031.42 | 1043.51 | 168.3 | 168.3 |
| 18 | 3 | 1111.11 | 16.6 | 858.5 | 1124.54 | 1091.89 | 1094.03 | 1094.86 | 113.8 | 117.6 |

| | | TS | | | GTS | ACO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
| 18 | 4 | 1184.74 | 47.3 | 1342.1 | 1171.51 | 1124.38 | 1132.34 | 1137.17 | 136.2 | 139.5 |
| 18 | 5 | 960.43 | 2794.4 | 3116.9 | 945.88 | 926.34 | 926.36 | 926.40 | 128.8 | 139.7 |
| 19 | 1 | 524.61 | 267.4 | 1057.8 | 524.61 | 524.61 | 527.29 | 533.00 | 8.0 | 11.3 |
| 19 | 2 | 786.11 | 117.8 | 527.9 | 796.87 | 766.99 | 775.51 | 780.10 | 42.8 | 51.6 |
| 19 | 3 | 797.54 | 727.2 | 888.1 | 816.77 | 786.43 | 786.43 | 786.43 | 46.5 | 46.7 |
| 19 | 4 | 831.80 | 166.3 | 1548.8 | 819.79 | 797.81 | 799.92 | 804.92 | 66.1 | 68.4 |
| 19 | 5 | 673.53 | 1889.6 | 3600.0 | 674.20 | 656.04 | 657.51 | 662.72 | 67.6 | 71.4 |
| 20 | 1 | 241.97 | 333.0 | 3039.3 | 244.54 | 241.97 | 242.16 | 242.44 | 49.5 | 77.3 |
| 20 | 2 | 570.59 | 339.1 | 2347.0 | 569.20 | 535.12 | 538.40 | 540.96 | 422.2 | 430.3 |
| 20 | 3 | 573.36 | 2532.8 | 3600.0 | 557.72 | 542.61 | 543.93 | 545.27 | 297.1 | 307.8 |
| 20 | 4 | 609.89 | 28.9 | 3600.0 | 576.92 | 551.92 | 554.46 | 559.36 | 227.5 | 227.5 |
| 20 | 5 | 504.84 | 1538.8 | 3600.2 | 503.01 | 481.00 | 484.35 | 487.34 | 318.6 | 330.9 |
| 21 | 1 | 688.18 | 2131.3 | 3600.0 | 687.60 | 690.20 | 691.94 | 694.91 | 26.5 | 47.9 |
| 21 | 2 | 1076.97 | 15.1 | 3600.0 | 1076.24 | 1014.07 | 1023.17 | 1034.83 | 512.6 | 515.4 |
| 21 | 3 | 1196.67 | 13.1 | 3600.0 | 1191.07 | 1140.99 | 1149.79 | 1154.72 | 232.7 | 245.1 |
| 21 | 4 | 1045.60 | 53.4 | 3600.0 | 1019.74 | 1001.14 | 1004.21 | 1011.05 | 398.6 | 404.2 |
| 21 | 5 | 945.60 | 87.7 | 3600.1 | 914.68 | 897.55 | 899.24 | 907.28 | 392.3 | 417.8 |
| 22 | 1 | 740.66 | 2236.5 | 3600.0 | 740.66 | 742.91 | 746.45 | 752.52 | 57.2 | 73.8 |
| 22 | 2 | 1098.61 | 223.0 | 3322.6 | 1088.33 | 1052.78 | 1060.19 | 1066.71 | 303.9 | 327.8 |
| 22 | 3 | 1095.02 | 41.5 | 3600.1 | 1110.73 | 1074.81 | 1081.33 | 1090.61 | 281.7 | 281.8 |
| 22 | 4 | 1134.32 | 350.9 | 3600.1 | 1119.34 | 1089.58 | 1091.48 | 1093.00 | 196.2 | 205.9 |
| 22 | 5 | 1021.87 | 3104.0 | 3600.1 | 986.02 | 958.94 | 967.01 | 975.39 | 321.4 | 380.4 |
| 23 | 1 | 860.47 | 849.1 | 3600.0 | 839.07 | 845.34 | 853.09 | 861.51 | 55.5 | 71.3 |
| 23 | 2 | 1125.85 | 3053.6 | 3293.1 | 1124.60 | 1047.49 | 1061.47 | 1068.97 | 828.4 | 872.2 |
| 23 | 3 | 1151.58 | 2378.4 | 3600.0 | 1141.51 | 1106.63 | 1114.15 | 1123.87 | 272.8 | 272.8 |
| 23 | 4 | 1146.52 | 2779.1 | 3600.0 | 1123.17 | 1089.49 | 1094.84 | 1104.24 | 245.5 | 249.1 |
| 23 | 5 | 994.94 | 1101.8 | 3600.0 | 975.42 | 963.12 | 965.95 | 970.12 | 247.4 | 265.9 |
| 24 | 1 | 1048.91 | 2322.1 | 3524.0 | 1035.33 | 1030.25 | 1042.26 | 1054.78 | 49.8 | 58.9 |
| 24 | 2 | 1285.18 | 1717.6 | 1956.3 | 1234.03 | 1188.21 | 1190.47 | 1193.65 | 164.8 | 187.2 |
| 24 | 3 | 1163.25 | 3185.7 | 3600.0 | 1136.10 | 1116.24 | 1120.51 | 1122.94 | 139.4 | 139.6 |
| 24 | 4 | 1215.46 | 39.2 | 3600.0 | 1160.92 | 1132.36 | 1148.54 | 1162.32 | 143.1 | 148.7 |
| 24 | 5 | 1086.60 | 6.2 | 3600.1 | 1065.41 | 1050.54 | 1055.33 | 1062.60 | 65.8 | 84.2 |
| 25 | 1 | 832.97 | 1981.2 | 3600.0 | 829.45 | 830.82 | 833.88 | 837.46 | 170.3 | 237.8 |
| 25 | 2 | 1501.06 | 128.1 | 3600.0 | 1500.07 | 1418.80 | 1430.78 | 1443.03 | 657.8 | 670.8 |
| 25 | 3 | 1445.56 | 1433.4 | 3600.0 | 1476.14 | 1411.93 | 1422.36 | 1433.80 | 669.3 | 669.4 |
| 25 | 4 | 1543.46 | 270.6 | 3600.0 | 1486.54 | 1445.35 | 1455.26 | 1462.70 | 638.5 | 658.7 |
| 25 | 5 | 1254.27 | 2215.4 | 3600.1 | 1212.73 | 1190.30 | 1192.93 | 1197.58 | 1126.2 | 1191.6 |
| 26 | 1 | 819.56 | 353.8 | 3600.1 | 819.56 | 819.56 | 819.56 | 819.56 | 175.9 | 276.8 |
| 26 | 2 | 1360.54 | 925.9 | 3600.0 | 1387.30 | 1308.76 | 1315.34 | 1322.36 | 821.6 | 865.5 |
| 26 | 3 | 1444.67 | 925.1 | 3600.0 | 1436.55 | 1376.56 | 1385.76 | 1395.30 | 697.7 | 697.7 |
| 26 | 4 | 1478.32 | 2710.3 | 3600.0 | 1491.00 | 1449.34 | 1452.71 | 1457.92 | 805.9 | 806.0 |
| 26 | 5 | 1334.41 | 954.4 | 3600.1 | 1267.68 | 1250.31 | 1254.20 | 1259.24 | 778.4 | 852.2 |
| 27 | 1 | 1099.95 | 986.1 | 3600.0 | 1097.63 | 1100.22 | 1104.42 | 1112.03 | 189.8 | 254.9 |
| 27 | 2 | 1443.10 | 19.4 | 3600.0 | 1402.42 | 1329.95 | 1348.79 | 1363.79 | 754.6 | 780.8 |

| | | TS | | | GTS | ACO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
| 27 | 3 | 1508.84 | 24.5 | 3600.0 | 1476.73 | 1411.27 | 1423.75 | 1430.37 | 499.8 | 499.9 |
| 27 | 4 | 1460.10 | 1373.7 | 3600.1 | 1397.75 | 1358.11 | 1367.68 | 1376.22 | 558.0 | 558.2 |
| 27 | 5 | 1388.66 | 202.8 | 3600.1 | 1309.50 | 1278.24 | 1283.21 | 1288.91 | 761.9 | 813.1 |
| 28 | 1 | 1078.27 | 3209.4 | 3600.4 | 1042.12 | 1050.49 | 1109.78 | 1176.84 | 256.8 | 339.5 |
| 28 | 2 | 2668.85 | 674.5 | 3600.0 | 2856.93 | 2599.42 | 2649.34 | 2697.05 | 9381.7 | 10559.2 |
| 28 | 3 | 2831.04 | 2069.2 | 3600.0 | 2867.46 | 2722.68 | 2735.98 | 2749.69 | 8154.6 | 8430.5 |
| 28 | 4 | 2820.11 | 2262.9 | 3600.1 | 2770.05 | 2709.82 | 2727.08 | 2748.30 | 5635.2 | 6199.8 |
| 28 | 5 | 2633.25 | 2039.9 | 3600.2 | 2453.59 | 2344.47 | 2385.94 | 2416.07 | 7940.2 | 9067.2 |
| 29 | 1 | 1179.01 | 1826.0 | 3600.2 | 1188.15 | 1173.46 | 1214.80 | 1271.35 | 761.4 | 870.9 |
| 29 | 2 | 2465.92 | 358.3 | 3600.0 | 2362.75 | 2231.37 | 2252.73 | 2305.57 | 10527.4 | 10800.4 |
| 29 | 3 | 2359.09 | 1769.8 | 3600.0 | 2249.80 | 2165.92 | 2182.98 | 2219.32 | 8916.2 | 10066.2 |
| 29 | 4 | 2582.33 | 153.9 | 3600.0 | 2427.95 | 2330.43 | 2339.91 | 2346.82 | 9922.2 | 10690.4 |
| 29 | 5 | 2489.99 | 2924.4 | 3600.0 | 2220.32 | 2186.08 | 2189.74 | 2195.81 | 8643.4 | 9879.4 |
| 30 | 1 | 1061.55 | 289.0 | 3600.6 | 1037.05 | 1037.71 | 1047.84 | 1062.37 | 310.2 | 389.9 |
| 30 | 2 | 1999.52 | 807.3 | 3600.0 | 1929.93 | 1825.89 | 1834.44 | 1850.97 | 9090.5 | 10124.9 |
| 30 | 3 | 2011.87 | 1015.7 | 3600.0 | 2038.55 | 1875.75 | 1890.32 | 1903.71 | 7618.6 | 7905.8 |
| 30 | 4 | 2099.18 | 682.0 | 3600.1 | 1965.45 | 1892.17 | 1909.83 | 1928.50 | 8243.9 | 9046.6 |
| 30 | 5 | 1684.32 | 3133.1 | 3600.8 | 1625.42 | 1566.86 | 1577.69 | 1587.22 | 7226.6 | 7885.3 |
| 31 | 1 | 1464.04 | 1783.5 | 3600.2 | 1421.20 | 1341.89 | 1368.25 | 1411.77 | 519.2 | 519.2 |
| 31 | 2 | 2464.92 | 2515.5 | 3600.4 | 2456.28 | 2296.18 | 2315.83 | 2336.05 | 10494.2 | 10665.3 |
| 31 | 3 | 2509.75 | 2790.4 | 3600.3 | 2478.94 | 2347.67 | 2356.66 | 2369.47 | 10007.5 | 10254.6 |
| 31 | 4 | 2756.75 | 2495.1 | 3600.1 | 2585.67 | 2474.74 | 2482.03 | 2489.26 | 7916.6 | 9173.9 |
| 31 | 5 | 2296.52 | 2294.4 | 3600.2 | 2132.92 | 2064.42 | 2077.77 | 2105.15 | 10217.1 | 10766.3 |
| 32 | 1 | 1352.61 | 2501.7 | 3600.4 | 1328.68 | 1334.26 | 1361.25 | 1404.35 | 511.3 | 516.2 |
| 32 | 2 | 2509.03 | 1070.1 | 3600.0 | 2465.17 | 2279.04 | 2308.55 | 2334.66 | 10541.6 | 10800.2 |
| 32 | 3 | 2545.84 | 1042.6 | 3600.4 | 2422.98 | 2318.12 | 2327.42 | 2343.23 | 10165.1 | 10548.0 |
| 32 | 4 | 2522.25 | 3306.8 | 3600.1 | 2432.49 | 2326.62 | 2348.28 | 2372.09 | 10223.6 | 10594.0 |
| 32 | 5 | 2368.60 | 402.7 | 3601.3 | 2086.13 | 2009.11 | 2028.06 | 2054.62 | 10625.4 | 10802.0 |
| 33 | 1 | 1361.51 | 798.9 | 3600.7 | 1328.19 | 1331.69 | 1356.13 | 1396.88 | 483.7 | 503.5 |
| 33 | 2 | 2472.05 | 649.7 | 3600.0 | 2508.68 | 2296.94 | 2312.70 | 2330.56 | 10246.1 | 10636.3 |
| 33 | 3 | 2595.28 | 2008.6 | 3600.0 | 2595.41 | 2432.20 | 2448.11 | 2467.67 | 9663.7 | 10399.5 |
| 33 | 4 | 2664.51 | 3241.0 | 3600.1 | 2601.34 | 2468.23 | 2480.79 | 2493.84 | 8742.0 | 9519.3 |
| 33 | 5 | 2284.15 | 2517.9 | 3600.2 | 2117.72 | 2040.56 | 2051.29 | 2071.49 | 9778.4 | 10461.3 |
| 34 | 1 | 858.94 | 1920.8 | 3606.5 | 719.91 | 712.32 | 715.72 | 727.38 | 612.2 | 639.2 |
| 34 | 2 | 1496.45 | 346.5 | 3600.0 | 1268.93 | 1197.54 | 1207.20 | 1213.66 | 10729.8 | 10800.4 |
| 34 | 3 | 1510.97 | 652.0 | 3600.0 | 1298.48 | 1242.74 | 1249.84 | 1256.89 | 9717.2 | 10532.4 |
| 34 | 4 | 1500.51 | 1416.4 | 3600.1 | 1279.65 | 1234.47 | 1242.79 | 1251.80 | 9686.6 | 10086.3 |
| 34 | 5 | 1356.30 | 2459.7 | 3600.1 | 1086.79 | 1059.45 | 1064.23 | 1072.16 | 10536.4 | 10754.9 |
| 35 | 1 | 992.86 | 798.4 | 3600.3 | 877.04 | 868.12 | 879.56 | 886.01 | 1481.5 | 1507.3 |
| 35 | 2 | 1684.13 | 3493.2 | 3600.1 | 1464.93 | 1400.37 | 1412.37 | 1435.89 | 10446.1 | 10800.4 |
| 35 | 3 | 1761.12 | 204.0 | 3600.1 | 1570.67 | 1489.58 | 1502.05 | 1510.14 | 10530.2 | 10800.3 |
| 35 | 4 | 2138.60 | 2885.0 | 3600.0 | 1634.63 | 1554.79 | 1564.96 | 1584.76 | 10462.7 | 10800.5 |
| 35 | 5 | 1477.35 | 2928.2 | 3601.7 | 1324.89 | 1281.90 | 1290.70 | 1301.78 | 10637.5 | 10800.4 |
| 36 | 1 | 677.16 | 3481.1 | 3600.2 | 594.10 | 617.91 | 626.43 | 634.53 | 2613.6 | 3058.8 |

| | | TS | | | GTS | ACO | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | *Cl.* | $z$ | $sec_h$ | $sec_{tot}$ | $z$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
| 36 | 2 | 1888.66 | 973.7 | 3600.0 | 1854.06 | 1754.17 | 1774.80 | 1792.82 | 10800.3 | 10800.3 |
| 36 | 3 | 2118.57 | 111.2 | 3600.0 | 1965.46 | 1862.33 | 1871.77 | 1886.49 | 10800.5 | 10800.5 |
| 36 | 4 | 1889.55 | 3455.4 | 3600.1 | 1803.86 | 1734.54 | 1745.55 | 1770.54 | 10707.4 | 10800.4 |
| 36 | 5 | 1742.73 | 3307.6 | 3600.2 | 1582.25 | 1540.89 | 1559.06 | 1586.36 | 10712.4 | 10800.6 |
| AVG | | 1131.33 | 757.9 | 1822.2 | 1100.46 | 1065.94 | 1072.44 | 1080.47 | 2058.8 | 2170.4 |

Table A.3: Detailed Comparison of ACO, TS and GTS (No Detailed Runtimes are Available in Zachariadis, Tarantilis, and Kiranoudis (2007)) for 2|UO|L

| *Inst.* | *Cl.* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec_h}$ | $\overline{sec_{tot}}$ |
|---|---|---|---|---|---|---|
| 1 | 1 | 278.73 | 278.73 | 278.73 | 0.1 | 0.4 |
| 1 | 2 | 278.73 | 278.73 | 278.73 | 3.5 | 5.3 |
| 1 | 3 | 284.52 | 284.52 | 284.52 | 5.1 | 6.7 |
| 1 | 4 | 282.95 | 282.95 | 282.95 | 0.3 | 0.8 |
| 1 | 5 | 280.60 | 283.61 | 285.93 | 0.8 | 1.9 |
| 2 | 1 | 334.96 | 334.96 | 334.96 | 0.0 | 0.3 |
| 2 | 2 | 334.96 | 334.96 | 334.96 | 0.2 | 0.4 |
| 2 | 3 | 352.16 | 352.16 | 352.16 | 0.6 | 0.8 |
| 2 | 4 | 342.00 | 342.00 | 342.00 | 0.4 | 0.6 |
| 2 | 5 | 334.96 | 334.96 | 334.96 | 0.2 | 0.5 |
| 3 | 1 | 358.40 | 358.40 | 358.40 | 0.2 | 0.8 |
| 3 | 2 | 384.93 | 385.04 | 385.29 | 1.6 | 2.5 |
| 3 | 3 | 394.72 | 394.72 | 394.72 | 2.2 | 3.4 |
| 3 | 4 | 368.56 | 368.56 | 368.56 | 1.5 | 3.1 |
| 3 | 5 | 358.40 | 358.40 | 358.40 | 0.5 | 1.3 |
| 4 | 1 | 430.89 | 430.89 | 430.89 | 0.3 | 0.7 |
| 4 | 2 | 430.89 | 430.89 | 430.89 | 2.6 | 2.6 |
| 4 | 3 | 430.89 | 439.19 | 445.49 | 1.7 | 2.5 |
| 4 | 4 | 447.37 | 447.37 | 447.37 | 2.3 | 3.4 |
| 4 | 5 | 430.89 | 430.89 | 430.89 | 1.5 | 2.2 |
| 5 | 1 | 375.28 | 375.28 | 375.28 | 0.3 | 1.0 |
| 5 | 2 | 375.28 | 377.91 | 384.06 | 11.4 | 15.5 |
| 5 | 3 | 379.94 | 382.09 | 383.52 | 13.7 | 16.1 |
| 5 | 4 | 383.88 | 384.11 | 384.27 | 14.9 | 18.4 |
| 5 | 5 | 375.28 | 375.28 | 375.28 | 3.2 | 4.8 |
| 6 | 1 | 495.85 | 495.85 | 495.85 | 0.3 | 0.9 |
| 6 | 2 | 498.16 | 498.16 | 498.16 | 2.2 | 2.9 |
| 6 | 3 | 498.16 | 498.76 | 504.23 | 3.8 | 4.8 |
| 6 | 4 | 498.32 | 498.35 | 498.65 | 5.7 | 7.1 |
| 6 | 5 | 495.85 | 495.85 | 495.85 | 1.5 | 2.7 |
| 7 | 1 | 568.56 | 568.56 | 568.56 | 0.2 | 1.3 |
| 7 | 2 | 716.82 | 716.82 | 716.82 | 4.2 | 7.5 |
| 7 | 3 | 706.99 | 706.99 | 706.99 | 4.0 | 8.5 |
| 7 | 4 | 702.45 | 703.31 | 704.25 | 9.9 | 10.1 |

| *Inst.* | *Cl.* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
|---|---|---|---|---|---|---|
| 7 | 5 | 658.64 | 660.98 | 665.79 | 12.2 | 17.0 |
| 8 | 1 | 568.56 | 568.56 | 568.56 | 0.2 | 1.3 |
| 8 | 2 | 674.20 | 674.20 | 674.20 | 5.0 | 7.9 |
| 8 | 3 | 740.85 | 741.04 | 741.12 | 7.5 | 8.0 |
| 8 | 4 | 701.87 | 713.45 | 719.73 | 10.0 | 11.4 |
| 8 | 5 | 646.46 | 646.46 | 646.46 | 22.1 | 52.8 |
| 9 | 1 | 607.65 | 607.65 | 607.65 | 0.6 | 1.3 |
| 9 | 2 | 607.65 | 612.69 | 620.25 | 1.9 | 3.0 |
| 9 | 3 | 607.65 | 607.65 | 607.65 | 3.6 | 4.6 |
| 9 | 4 | 625.13 | 625.13 | 625.13 | 4.6 | 5.7 |
| 9 | 5 | 607.65 | 607.65 | 607.65 | 2.4 | 3.5 |
| 10 | 1 | 535.80 | 535.80 | 535.80 | 2.4 | 4.2 |
| 10 | 2 | 685.21 | 685.25 | 685.61 | 35.5 | 35.5 |
| 10 | 3 | 617.62 | 626.99 | 638.73 | 34.0 | 35.1 |
| 10 | 4 | 743.73 | 749.03 | 756.40 | 34.6 | 34.9 |
| 10 | 5 | 694.50 | 695.59 | 698.30 | 70.5 | 74.2 |
| 11 | 1 | 505.01 | 505.01 | 505.01 | 0.8 | 3.1 |
| 11 | 2 | 694.60 | 695.83 | 706.87 | 22.0 | 24.3 |
| 11 | 3 | 706.07 | 711.82 | 718.53 | 19.0 | 19.3 |
| 11 | 4 | 802.55 | 815.42 | 820.75 | 26.9 | 27.6 |
| 11 | 5 | 647.14 | 650.94 | 660.29 | 59.5 | 69.1 |
| 12 | 1 | 610.00 | 611.22 | 614.24 | 1.5 | 2.4 |
| 12 | 2 | 618.36 | 625.63 | 627.68 | 3.7 | 4.9 |
| 12 | 3 | 610.23 | 612.74 | 614.59 | 2.3 | 3.3 |
| 12 | 4 | 614.24 | 619.54 | 622.86 | 5.7 | 7.1 |
| 12 | 5 | 610.23 | 611.94 | 614.59 | 5.0 | 5.8 |
| 13 | 1 | 2006.34 | 2006.34 | 2006.34 | 1.3 | 4.1 |
| 13 | 2 | 2526.07 | 2534.89 | 2543.47 | 34.3 | 43.9 |
| 13 | 3 | 2473.36 | 2473.84 | 2477.74 | 31.5 | 39.0 |
| 13 | 4 | 2623.65 | 2643.99 | 2701.50 | 51.3 | 51.9 |
| 13 | 5 | 2434.99 | 2441.77 | 2446.29 | 103.0 | 107.3 |
| 14 | 1 | 837.67 | 837.67 | 837.67 | 4.2 | 6.1 |
| 14 | 2 | 1041.03 | 1041.72 | 1042.38 | 84.4 | 84.4 |
| 14 | 3 | 1021.79 | 1037.82 | 1057.79 | 114.7 | 115.1 |
| 14 | 4 | 988.25 | 991.42 | 992.98 | 108.4 | 114.0 |
| 14 | 5 | 928.95 | 932.85 | 943.02 | 208.9 | 237.5 |
| 15 | 1 | 837.67 | 837.67 | 837.67 | 2.9 | 4.3 |
| 15 | 2 | 1009.87 | 1016.05 | 1028.06 | 78.5 | 78.5 |
| 15 | 3 | 1166.80 | 1171.04 | 1173.26 | 78.1 | 78.3 |
| 15 | 4 | 1247.27 | 1258.24 | 1267.71 | 90.2 | 91.7 |
| 15 | 5 | 1233.09 | 1239.70 | 1251.06 | 203.4 | 207.5 |
| 16 | 1 | 698.61 | 698.61 | 698.61 | 2.0 | 3.9 |
| 16 | 2 | 698.61 | 699.01 | 702.70 | 5.4 | 7.3 |
| 16 | 3 | 698.61 | 699.01 | 702.70 | 6.1 | 8.0 |
| 16 | 4 | 703.35 | 704.12 | 710.98 | 9.5 | 10.9 |

| *Inst.* | *Cl.* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
|---|---|---|---|---|---|---|
| 16 | 5 | 698.61 | 698.61 | 698.61 | 4.5 | 7.2 |
| 17 | 1 | 861.79 | 862.37 | 862.62 | 3.3 | 6.3 |
| 17 | 2 | 863.27 | 863.27 | 863.27 | 3.6 | 6.6 |
| 17 | 3 | 862.62 | 862.62 | 862.62 | 3.3 | 6.3 |
| 17 | 4 | 861.79 | 862.53 | 862.62 | 4.0 | 7.6 |
| 17 | 5 | 861.79 | 863.00 | 864.63 | 3.2 | 7.2 |
| 18 | 1 | 723.54 | 726.18 | 727.74 | 9.4 | 14.8 |
| 18 | 2 | 989.21 | 995.62 | 1005.87 | 148.7 | 155.7 |
| 18 | 3 | 1031.94 | 1035.50 | 1044.08 | 126.9 | 129.0 |
| 18 | 4 | 1129.00 | 1137.26 | 1141.10 | 252.9 | 253.6 |
| 18 | 5 | 926.40 | 927.82 | 931.77 | 433.4 | 461.3 |
| 19 | 1 | 524.61 | 527.29 | 533.00 | 8.0 | 11.4 |
| 19 | 2 | 732.64 | 739.66 | 745.85 | 50.2 | 50.2 |
| 19 | 3 | 770.19 | 781.64 | 790.67 | 63.9 | 64.1 |
| 19 | 4 | 792.61 | 799.24 | 804.27 | 89.4 | 91.4 |
| 19 | 5 | 668.10 | 672.95 | 675.24 | 197.3 | 203.3 |
| 20 | 1 | 241.97 | 242.16 | 242.44 | 51.9 | 79.8 |
| 20 | 2 | 498.01 | 501.58 | 511.26 | 308.1 | 308.3 |
| 20 | 3 | 537.45 | 539.80 | 542.22 | 355.5 | 355.6 |
| 20 | 4 | 553.15 | 556.97 | 560.37 | 403.5 | 404.2 |
| 20 | 5 | 488.72 | 490.28 | 492.28 | 908.6 | 922.8 |
| 21 | 1 | 690.20 | 691.94 | 694.91 | 26.8 | 48.1 |
| 21 | 2 | 998.48 | 1011.64 | 1020.61 | 327.3 | 327.5 |
| 21 | 3 | 1134.36 | 1140.40 | 1149.60 | 321.6 | 321.9 |
| 21 | 4 | 1013.86 | 1021.65 | 1024.98 | 561.6 | 562.0 |
| 21 | 5 | 906.01 | 914.01 | 922.81 | 932.3 | 943.4 |
| 22 | 1 | 742.91 | 746.45 | 752.52 | 58.3 | 75.0 |
| 22 | 2 | 1005.34 | 1014.74 | 1020.83 | 259.5 | 261.7 |
| 22 | 3 | 1060.49 | 1076.62 | 1084.57 | 322.4 | 322.4 |
| 22 | 4 | 1088.21 | 1098.62 | 1108.86 | 354.0 | 354.1 |
| 22 | 5 | 967.03 | 974.65 | 983.47 | 1069.0 | 1074.7 |
| 23 | 1 | 845.34 | 853.09 | 861.51 | 55.9 | 71.7 |
| 23 | 2 | 1025.48 | 1036.99 | 1048.42 | 313.2 | 318.3 |
| 23 | 3 | 1095.87 | 1104.79 | 1117.40 | 321.7 | 321.7 |
| 23 | 4 | 1098.82 | 1105.80 | 1113.02 | 386.9 | 387.3 |
| 23 | 5 | 965.57 | 969.85 | 979.00 | 586.9 | 604.2 |
| 24 | 1 | 1030.25 | 1042.26 | 1054.78 | 49.7 | 58.7 |
| 24 | 2 | 1181.07 | 1187.53 | 1195.98 | 149.6 | 158.3 |
| 24 | 3 | 1098.01 | 1110.11 | 1123.86 | 145.9 | 146.2 |
| 24 | 4 | 1144.08 | 1149.12 | 1157.50 | 191.3 | 191.7 |
| 24 | 5 | 1065.72 | 1069.35 | 1074.59 | 183.6 | 196.6 |
| 25 | 1 | 830.82 | 833.88 | 837.46 | 171.7 | 239.3 |
| 25 | 2 | 1381.61 | 1395.47 | 1404.51 | 665.0 | 670.4 |
| 25 | 3 | 1385.49 | 1401.32 | 1418.46 | 704.3 | 704.4 |
| 25 | 4 | 1443.83 | 1455.36 | 1471.98 | 957.1 | 957.3 |

| *Inst.* | *Cl.* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
|---|---|---|---|---|---|---|
| 25 | 5 | 1209.72 | 1216.98 | 1225.48 | 3271.4 | 3290.0 |
| 26 | 1 | 819.56 | 819.56 | 819.56 | 177.8 | 278.7 |
| 26 | 2 | 1280.85 | 1290.07 | 1306.40 | 701.3 | 701.6 |
| 26 | 3 | 1363.00 | 1378.35 | 1389.24 | 813.5 | 813.6 |
| 26 | 4 | 1452.49 | 1460.29 | 1470.57 | 2124.8 | 2125.1 |
| 26 | 5 | 1261.13 | 1265.84 | 1271.25 | 2121.4 | 2140.4 |
| 27 | 1 | 1100.22 | 1104.42 | 1112.03 | 191.1 | 256.8 |
| 27 | 2 | 1302.23 | 1320.00 | 1329.78 | 609.4 | 618.6 |
| 27 | 3 | 1409.70 | 1418.25 | 1428.89 | 592.2 | 592.4 |
| 27 | 4 | 1367.90 | 1378.70 | 1384.73 | 939.8 | 940.0 |
| 27 | 5 | 1287.06 | 1299.60 | 1315.05 | 1861.5 | 1870.6 |
| 28 | 1 | 1050.49 | 1109.78 | 1176.84 | 269.8 | 356.7 |
| 28 | 2 | 2549.67 | 2577.94 | 2618.90 | 9212.1 | 10076.3 |
| 28 | 3 | 2690.96 | 2719.51 | 2746.08 | 7007.6 | 7991.4 |
| 28 | 4 | 2728.68 | 2737.72 | 2755.44 | 9221.2 | 10521.8 |
| 28 | 5 | 2410.90 | 2429.07 | 2457.50 | 10312.4 | 10800.9 |
| 29 | 1 | 1173.46 | 1214.80 | 1271.35 | 837.8 | 957.4 |
| 29 | 2 | 2198.66 | 2206.63 | 2222.62 | 9339.3 | 10537.9 |
| 29 | 3 | 2125.18 | 2153.09 | 2180.53 | 10034.2 | 10647.7 |
| 29 | 4 | 2318.59 | 2330.64 | 2344.90 | 9921.4 | 10730.4 |
| 29 | 5 | 2200.71 | 2221.29 | 2260.67 | 10214.3 | 10801.3 |
| 30 | 1 | 1037.71 | 1047.84 | 1062.37 | 330.5 | 416.0 |
| 30 | 2 | 1816.36 | 1822.76 | 1831.32 | 7229.7 | 8066.9 |
| 30 | 3 | 1869.51 | 1884.74 | 1900.32 | 6269.0 | 6691.6 |
| 30 | 4 | 1908.29 | 1919.85 | 1935.44 | 8416.9 | 8956.9 |
| 30 | 5 | 1593.29 | 1614.56 | 1636.36 | 10147.1 | 10800.7 |
| 31 | 1 | 1341.89 | 1368.25 | 1411.77 | 555.6 | 555.6 |
| 31 | 2 | 2258.56 | 2275.72 | 2295.22 | 8322.9 | 9004.6 |
| 31 | 3 | 2322.42 | 2339.26 | 2365.50 | 9468.8 | 10442.2 |
| 31 | 4 | 2473.93 | 2490.26 | 2501.32 | 10078.8 | 10609.0 |
| 31 | 5 | 2099.61 | 2114.48 | 2138.30 | 9911.9 | 10800.3 |
| 32 | 1 | 1334.26 | 1361.25 | 1404.35 | 528.1 | 533.3 |
| 32 | 2 | 2256.26 | 2272.01 | 2286.70 | 9603.2 | 10012.2 |
| 32 | 3 | 2296.68 | 2313.26 | 2333.74 | 9733.2 | 10537.0 |
| 32 | 4 | 2365.12 | 2385.49 | 2400.69 | 10372.9 | 10800.7 |
| 32 | 5 | 2050.12 | 2072.90 | 2127.26 | 10175.2 | 10800.3 |
| 33 | 1 | 1331.69 | 1356.13 | 1396.88 | 497.4 | 517.6 |
| 33 | 2 | 2252.56 | 2269.86 | 2299.17 | 9503.8 | 10310.3 |
| 33 | 3 | 2404.31 | 2428.51 | 2445.39 | 9086.5 | 9662.5 |
| 33 | 4 | 2484.35 | 2494.24 | 2508.86 | 10666.2 | 10800.2 |
| 33 | 5 | 2086.42 | 2098.42 | 2117.66 | 10632.4 | 10800.5 |
| 34 | 1 | 712.32 | 715.72 | 727.38 | 638.6 | 666.9 |
| 34 | 2 | 1170.91 | 1177.74 | 1187.91 | 10522.5 | 10753.2 |
| 34 | 3 | 1234.43 | 1242.10 | 1251.99 | 10210.8 | 10584.4 |
| 34 | 4 | 1248.40 | 1256.38 | 1268.16 | 10637.8 | 10800.5 |

| *Inst.* | *Cl.* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
|---|---|---|---|---|---|---|
| 34 | 5 | 1080.20 | 1088.92 | 1101.34 | 10619.0 | 10800.8 |
| 35 | 1 | 868.12 | 879.56 | 886.01 | 1501.5 | 1528.9 |
| 35 | 2 | 1376.90 | 1383.44 | 1389.76 | 10491.2 | 10800.1 |
| 35 | 3 | 1483.92 | 1492.12 | 1501.41 | 10699.1 | 10800.2 |
| 35 | 4 | 1572.54 | 1593.17 | 1614.70 | 10797.5 | 10800.3 |
| 35 | 5 | 1319.32 | 1337.03 | 1367.74 | 10760.6 | 10800.7 |
| 36 | 1 | 617.91 | 626.43 | 634.53 | 2696.2 | 3130.2 |
| 36 | 2 | 1728.73 | 1739.95 | 1759.42 | 10774.2 | 10800.4 |
| 36 | 3 | 1841.12 | 1854.43 | 1868.47 | 10671.5 | 10800.3 |
| 36 | 4 | 1757.91 | 1773.41 | 1794.67 | 10800.6 | 10800.6 |
| 36 | 5 | 1592.36 | 1607.09 | 1624.92 | 10801.2 | 10801.2 |
| AVG | | 1063.53 | 1070.86 | 1080.15 | 2156.6 | 2264.5 |

Table A.4: Results Obtained by ACO for 2|RN|L

| *Inst.* | *Cl.* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
|---|---|---|---|---|---|---|
| 1 | 1 | 278.73 | 278.73 | 278.73 | 0.1 | 0.4 |
| 1 | 2 | 278.73 | 279.31 | 284.52 | 1.5 | 2.4 |
| 1 | 3 | 284.23 | 284.29 | 284.52 | 2.3 | 2.9 |
| 1 | 4 | 282.95 | 282.95 | 282.95 | 0.1 | 0.5 |
| 1 | 5 | 278.73 | 279.58 | 280.60 | 0.3 | 0.7 |
| 2 | 1 | 334.96 | 334.96 | 334.96 | 0.0 | 0.3 |
| 2 | 2 | 334.96 | 334.96 | 334.96 | 0.1 | 0.4 |
| 2 | 3 | 352.16 | 352.16 | 352.16 | 0.4 | 0.6 |
| 2 | 4 | 342.00 | 342.00 | 342.00 | 0.1 | 0.4 |
| 2 | 5 | 334.96 | 334.96 | 334.96 | 0.1 | 0.4 |
| 3 | 1 | 358.40 | 358.40 | 358.40 | 0.2 | 0.8 |
| 3 | 2 | 380.35 | 384.01 | 384.93 | 1.3 | 2.1 |
| 3 | 3 | 390.55 | 393.47 | 394.72 | 0.9 | 1.9 |
| 3 | 4 | 362.41 | 362.41 | 362.41 | 0.5 | 1.2 |
| 3 | 5 | 358.40 | 358.40 | 358.40 | 0.4 | 1.0 |
| 4 | 1 | 430.89 | 430.89 | 430.89 | 0.3 | 0.7 |
| 4 | 2 | 430.89 | 430.89 | 430.89 | 1.1 | 1.7 |
| 4 | 3 | 430.89 | 430.89 | 430.89 | 1.2 | 1.5 |
| 4 | 4 | 447.37 | 447.37 | 447.37 | 1.7 | 2.4 |
| 4 | 5 | 430.89 | 430.89 | 430.89 | 1.0 | 1.5 |
| 5 | 1 | 375.28 | 375.28 | 375.28 | 0.3 | 1.0 |
| 5 | 2 | 375.28 | 375.28 | 375.28 | 10.4 | 13.8 |
| 5 | 3 | 379.94 | 379.94 | 379.94 | 4.0 | 7.7 |
| 5 | 4 | 383.88 | 384.98 | 387.54 | 6.9 | 10.2 |
| 5 | 5 | 375.28 | 375.28 | 375.28 | 1.2 | 2.0 |
| 6 | 1 | 495.85 | 495.85 | 495.85 | 0.3 | 0.9 |
| 6 | 2 | 495.85 | 495.85 | 495.85 | 1.4 | 2.2 |
| 6 | 3 | 498.16 | 498.16 | 498.16 | 1.9 | 3.0 |
| 6 | 4 | 498.32 | 498.35 | 498.65 | 3.3 | 4.6 |

| *Inst.* | *Cl.* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
|---|---|---|---|---|---|---|
| 6 | 5 | 495.85 | 495.85 | 495.85 | 0.8 | 1.5 |
| 7 | 1 | 568.56 | 568.56 | 568.56 | 0.2 | 1.3 |
| 7 | 2 | 715.02 | 715.02 | 715.02 | 1.6 | 3.6 |
| 7 | 3 | 674.23 | 678.30 | 678.75 | 3.1 | 4.4 |
| 7 | 4 | 702.45 | 702.45 | 702.45 | 3.0 | 4.8 |
| 7 | 5 | 657.77 | 657.77 | 657.77 | 3.2 | 5.4 |
| 8 | 1 | 568.56 | 568.56 | 568.56 | 0.2 | 1.3 |
| 8 | 2 | 674.20 | 675.25 | 684.76 | 3.0 | 4.0 |
| 8 | 3 | 738.43 | 740.31 | 741.12 | 1.4 | 4.3 |
| 8 | 4 | 692.47 | 692.47 | 692.47 | 3.6 | 4.6 |
| 8 | 5 | 609.90 | 609.90 | 609.90 | 11.0 | 25.5 |
| 9 | 1 | 607.65 | 607.65 | 607.65 | 0.6 | 1.3 |
| 9 | 2 | 607.65 | 607.65 | 607.65 | 1.7 | 2.3 |
| 9 | 3 | 607.65 | 607.65 | 607.65 | 1.8 | 3.3 |
| 9 | 4 | 625.13 | 627.54 | 633.42 | 2.9 | 3.9 |
| 9 | 5 | 607.65 | 607.65 | 607.65 | 1.1 | 2.0 |
| 10 | 1 | 535.80 | 535.80 | 535.80 | 2.3 | 4.1 |
| 10 | 2 | 684.42 | 684.42 | 684.42 | 24.8 | 24.8 |
| 10 | 3 | 615.68 | 619.40 | 620.33 | 16.2 | 21.9 |
| 10 | 4 | 703.64 | 705.41 | 709.57 | 10.0 | 11.3 |
| 10 | 5 | 686.12 | 687.14 | 691.20 | 18.9 | 21.8 |
| 11 | 1 | 505.01 | 505.01 | 505.01 | 0.8 | 3.1 |
| 11 | 2 | 686.18 | 686.66 | 687.38 | 15.4 | 19.2 |
| 11 | 3 | 706.94 | 706.94 | 706.94 | 7.1 | 9.5 |
| 11 | 4 | 782.31 | 782.35 | 782.37 | 14.1 | 15.9 |
| 11 | 5 | 624.82 | 624.82 | 624.82 | 7.5 | 15.0 |
| 12 | 1 | 610.00 | 611.22 | 614.24 | 1.5 | 2.4 |
| 12 | 2 | 610.00 | 613.68 | 624.31 | 3.1 | 4.3 |
| 12 | 3 | 614.24 | 614.27 | 614.61 | 1.6 | 2.7 |
| 12 | 4 | 614.24 | 616.22 | 619.21 | 2.5 | 4.7 |
| 12 | 5 | 614.24 | 615.09 | 615.61 | 3.0 | 4.0 |
| 13 | 1 | 2006.34 | 2006.34 | 2006.34 | 1.3 | 4.0 |
| 13 | 2 | 2504.53 | 2504.53 | 2504.53 | 30.2 | 30.2 |
| 13 | 3 | 2450.19 | 2450.41 | 2451.28 | 20.5 | 20.5 |
| 13 | 4 | 2583.72 | 2584.43 | 2590.85 | 13.7 | 20.7 |
| 13 | 5 | 2334.78 | 2334.78 | 2334.78 | 24.9 | 30.1 |
| 14 | 1 | 837.67 | 837.67 | 837.67 | 4.1 | 6.0 |
| 14 | 2 | 1032.01 | 1032.60 | 1037.88 | 50.8 | 61.4 |
| 14 | 3 | 992.64 | 996.20 | 996.79 | 29.3 | 34.5 |
| 14 | 4 | 981.90 | 981.90 | 981.90 | 34.0 | 46.9 |
| 14 | 5 | 914.02 | 920.90 | 921.72 | 49.9 | 57.0 |
| 15 | 1 | 837.67 | 837.67 | 837.67 | 2.8 | 4.3 |
| 15 | 2 | 1008.09 | 1009.58 | 1015.06 | 47.4 | 48.5 |
| 15 | 3 | 1153.41 | 1158.30 | 1167.82 | 47.7 | 51.8 |
| 15 | 4 | 1218.17 | 1238.01 | 1240.21 | 36.7 | 43.7 |

| *Inst.* | *Cl.* | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
|---|---|---|---|---|---|---|
| 15 | 5 | 1160.20 | 1167.61 | 1178.65 | 41.7 | 48.2 |
| 16 | 1 | 698.61 | 698.61 | 698.61 | 2.0 | 3.9 |
| 16 | 2 | 698.61 | 700.52 | 704.08 | 4.6 | 6.0 |
| 16 | 3 | 698.61 | 699.06 | 703.18 | 4.5 | 6.2 |
| 16 | 4 | 703.35 | 703.35 | 703.35 | 4.8 | 6.8 |
| 16 | 5 | 698.61 | 698.61 | 698.61 | 3.8 | 5.7 |
| 17 | 1 | 861.79 | 862.37 | 862.62 | 3.3 | 6.2 |
| 17 | 2 | 863.27 | 863.27 | 863.27 | 2.8 | 6.1 |
| 17 | 3 | 862.62 | 862.62 | 862.62 | 2.6 | 6.0 |
| 17 | 4 | 862.62 | 862.62 | 862.62 | 3.4 | 6.7 |
| 17 | 5 | 861.79 | 861.79 | 861.79 | 4.7 | 7.3 |
| 18 | 1 | 723.54 | 726.18 | 727.74 | 9.1 | 14.5 |
| 18 | 2 | 988.91 | 988.91 | 988.91 | 87.3 | 100.2 |
| 18 | 3 | 1025.35 | 1028.19 | 1032.56 | 68.4 | 68.5 |
| 18 | 4 | 1110.48 | 1112.84 | 1122.30 | 92.4 | 96.5 |
| 18 | 5 | 924.04 | 924.13 | 924.66 | 102.9 | 114.8 |
| 19 | 1 | 524.61 | 527.29 | 533.00 | 7.9 | 11.2 |
| 19 | 2 | 730.16 | 731.61 | 736.22 | 33.7 | 34.1 |
| 19 | 3 | 753.66 | 753.66 | 753.66 | 29.5 | 29.6 |
| 19 | 4 | 776.66 | 788.18 | 792.90 | 42.6 | 43.4 |
| 19 | 5 | 651.97 | 655.15 | 657.10 | 44.1 | 49.7 |
| 20 | 1 | 241.97 | 242.16 | 242.44 | 49.5 | 77.1 |
| 20 | 2 | 492.90 | 495.39 | 496.63 | 208.7 | 216.6 |
| 20 | 3 | 517.61 | 517.65 | 518.01 | 173.1 | 182.6 |
| 20 | 4 | 546.91 | 548.23 | 549.11 | 166.5 | 166.6 |
| 20 | 5 | 476.97 | 478.30 | 479.15 | 280.3 | 301.8 |
| 21 | 1 | 690.20 | 691.94 | 694.91 | 26.9 | 48.2 |
| 21 | 2 | 979.90 | 985.65 | 993.50 | 221.2 | 223.3 |
| 21 | 3 | 1111.94 | 1117.63 | 1122.90 | 195.0 | 200.1 |
| 21 | 4 | 975.80 | 985.85 | 991.34 | 317.8 | 318.0 |
| 21 | 5 | 890.16 | 893.88 | 897.22 | 289.5 | 314.9 |
| 22 | 1 | 742.91 | 746.45 | 752.52 | 58.1 | 74.8 |
| 22 | 2 | 988.15 | 988.71 | 991.18 | 174.8 | 177.8 |
| 22 | 3 | 1045.95 | 1054.87 | 1058.67 | 171.0 | 177.6 |
| 22 | 4 | 1058.29 | 1063.40 | 1067.75 | 124.8 | 142.3 |
| 22 | 5 | 944.10 | 949.44 | 954.63 | 278.7 | 305.0 |
| 23 | 1 | 845.34 | 853.09 | 861.51 | 55.6 | 71.4 |
| 23 | 2 | 1005.13 | 1015.91 | 1022.02 | 190.3 | 195.1 |
| 23 | 3 | 1071.28 | 1083.49 | 1088.28 | 204.7 | 211.5 |
| 23 | 4 | 1078.74 | 1081.84 | 1089.89 | 180.6 | 183.0 |
| 23 | 5 | 943.60 | 952.95 | 963.41 | 190.7 | 202.5 |
| 24 | 1 | 1030.25 | 1042.26 | 1054.78 | 49.5 | 58.5 |
| 24 | 2 | 1172.00 | 1172.88 | 1179.57 | 111.1 | 111.1 |
| 24 | 3 | 1088.09 | 1097.45 | 1107.07 | 78.4 | 90.3 |
| 24 | 4 | 1106.91 | 1116.36 | 1123.77 | 94.4 | 105.6 |

| $Inst.$ | $Cl.$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
|---|---|---|---|---|---|---|
| 24 | 5 | 1048.66 | 1052.74 | 1061.39 | 63.3 | 74.4 |
| 25 | 1 | 830.82 | 833.88 | 837.46 | 171.1 | 238.6 |
| 25 | 2 | 1373.56 | 1380.38 | 1394.47 | 493.2 | 509.0 |
| 25 | 3 | 1371.59 | 1376.95 | 1382.75 | 428.8 | 429.1 |
| 25 | 4 | 1412.70 | 1419.26 | 1426.20 | 494.7 | 501.0 |
| 25 | 5 | 1173.47 | 1179.77 | 1185.20 | 893.8 | 935.7 |
| 26 | 1 | 819.56 | 819.56 | 819.56 | 175.4 | 276.6 |
| 26 | 2 | 1267.66 | 1268.34 | 1270.96 | 471.1 | 501.7 |
| 26 | 3 | 1344.56 | 1354.90 | 1363.63 | 552.2 | 552.2 |
| 26 | 4 | 1412.80 | 1416.61 | 1420.54 | 526.0 | 569.3 |
| 26 | 5 | 1228.92 | 1233.55 | 1238.62 | 636.3 | 659.5 |
| 27 | 1 | 1100.22 | 1104.42 | 1112.03 | 190.6 | 256.8 |
| 27 | 2 | 1284.87 | 1296.63 | 1315.20 | 441.6 | 441.6 |
| 27 | 3 | 1375.97 | 1382.40 | 1389.41 | 351.6 | 381.5 |
| 27 | 4 | 1320.85 | 1331.04 | 1346.12 | 413.4 | 413.5 |
| 27 | 5 | 1258.43 | 1262.09 | 1272.83 | 522.9 | 573.1 |
| 28 | 1 | 1050.49 | 1109.78 | 1176.84 | 256.4 | 338.9 |
| 28 | 2 | 2530.75 | 2546.01 | 2589.87 | 5139.6 | 6125.7 |
| 28 | 3 | 2595.84 | 2635.79 | 2663.40 | 3975.1 | 4484.4 |
| 28 | 4 | 2622.84 | 2659.68 | 2682.85 | 3165.8 | 3787.6 |
| 28 | 5 | 2325.75 | 2337.02 | 2395.18 | 5019.8 | 5643.4 |
| 29 | 1 | 1173.46 | 1214.80 | 1271.35 | 761.2 | 870.3 |
| 29 | 2 | 2182.32 | 2195.13 | 2212.87 | 8629.7 | 9267.3 |
| 29 | 3 | 2094.01 | 2101.95 | 2113.25 | 6279.1 | 7088.7 |
| 29 | 4 | 2266.64 | 2279.39 | 2296.42 | 4867.9 | 5699.6 |
| 29 | 5 | 2153.98 | 2164.11 | 2182.53 | 6442.0 | 6971.7 |
| 30 | 1 | 1037.71 | 1047.84 | 1062.37 | 310.9 | 390.6 |
| 30 | 2 | 1780.14 | 1788.82 | 1800.85 | 4817.2 | 4966.8 |
| 30 | 3 | 1828.38 | 1834.73 | 1841.35 | 4049.4 | 4176.8 |
| 30 | 4 | 1848.10 | 1854.08 | 1862.89 | 5026.6 | 5378.9 |
| 30 | 5 | 1544.52 | 1552.89 | 1564.27 | 4812.9 | 5051.5 |
| 31 | 1 | 1341.89 | 1368.25 | 1411.77 | 522.0 | 522.1 |
| 31 | 2 | 2231.85 | 2244.81 | 2260.14 | 6690.5 | 7489.4 |
| 31 | 3 | 2277.22 | 2286.41 | 2294.68 | 5911.7 | 6260.5 |
| 31 | 4 | 2397.19 | 2409.30 | 2445.63 | 5118.5 | 5366.7 |
| 31 | 5 | 2014.44 | 2029.75 | 2050.53 | 7824.9 | 8194.7 |
| 32 | 1 | 1334.26 | 1361.25 | 1404.35 | 512.2 | 517.2 |
| 32 | 2 | 2219.87 | 2232.14 | 2241.97 | 6708.4 | 6999.2 |
| 32 | 3 | 2255.07 | 2265.66 | 2272.48 | 7341.2 | 8002.5 |
| 32 | 4 | 2270.38 | 2287.09 | 2301.87 | 5206.7 | 5807.5 |
| 32 | 5 | 1975.87 | 1992.88 | 2018.06 | 9218.0 | 9852.8 |
| 33 | 1 | 1331.69 | 1356.13 | 1396.88 | 485.2 | 505.0 |
| 33 | 2 | 2208.08 | 2219.10 | 2231.33 | 7992.1 | 8435.0 |
| 33 | 3 | 2360.35 | 2374.40 | 2385.74 | 6213.0 | 6782.9 |
| 33 | 4 | 2388.44 | 2408.24 | 2418.28 | 6524.1 | 6812.8 |

| $Inst.$ | $Cl.$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ |
|---|---|---|---|---|---|---|
| 33 | 5 | 1998.98 | 2006.65 | 2017.90 | 7657.7 | 7769.0 |
| 34 | 1 | 712.32 | 715.72 | 727.38 | 612.1 | 639.1 |
| 34 | 2 | 1150.01 | 1156.44 | 1164.82 | 8234.1 | 8851.3 |
| 34 | 3 | 1199.33 | 1209.67 | 1217.13 | 5747.3 | 6520.3 |
| 34 | 4 | 1204.03 | 1213.01 | 1223.47 | 7260.3 | 7298.7 |
| 34 | 5 | 1033.44 | 1042.15 | 1047.81 | 7583.4 | 8281.9 |
| 35 | 1 | 868.12 | 879.56 | 886.01 | 1441.1 | 1466.1 |
| 35 | 2 | 1354.63 | 1358.30 | 1364.25 | 10435.1 | 10800.5 |
| 35 | 3 | 1437.40 | 1444.62 | 1453.41 | 9810.7 | 10263.4 |
| 35 | 4 | 1508.25 | 1510.32 | 1514.54 | 9618.7 | 10139.2 |
| 35 | 5 | 1262.86 | 1270.27 | 1285.17 | 10664.7 | 10746.7 |
| 36 | 1 | 617.91 | 626.43 | 634.53 | 2555.0 | 2981.1 |
| 36 | 2 | 1692.62 | 1704.19 | 1724.77 | 10754.1 | 10800.2 |
| 36 | 3 | 1785.20 | 1796.30 | 1808.05 | 10769.3 | 10800.2 |
| 36 | 4 | 1677.32 | 1684.70 | 1691.07 | 10606.1 | 10800.3 |
| 36 | 5 | 1508.48 | 1517.59 | 1531.79 | 10632.5 | 10800.5 |
| AVG | | 1045.03 | 1050.41 | 1057.27 | 1534.0 | 1631.4 |

Table A.5: Results Obtained by ACO for 2|UN|L

# B Detailed Results for the 3L-CVRP

| | TS | | ACO | | | | | %*gap* ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | $z$ | $sec_h$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 316.32 | 129.5 | 304.13 | 305.35 | 307.18 | 11.2 | 12.0 | -3.85 | -3.47 | -2.89 |
| 2 | 350.58 | 5.3 | 334.96 | 334.96 | 334.96 | 0.1 | 0.6 | -4.46 | -4.46 | -4.46 |
| 3 | 447.73 | 461.1 | 399.68 | 409.79 | 415.87 | 88.5 | 121.8 | -10.73 | -8.47 | -7.12 |
| 4 | 448.48 | 181.1 | 440.68 | 440.68 | 440.68 | 3.9 | 5.4 | -1.74 | -1.74 | -1.74 |
| 5 | 464.24 | 75.8 | 450.93 | 453.19 | 454.14 | 22.7 | 30.9 | -2.87 | -2.38 | -2.18 |
| 6 | 504.46 | 1167.9 | 498.32 | 501.47 | 504.39 | 17.5 | 18.4 | -1.22 | -0.59 | -0.01 |
| 7 | 831.66 | 181.1 | 792.13 | 797.47 | 801.73 | 51.4 | 67.4 | -4.75 | -4.11 | -3.60 |
| 8 | 871.77 | 156.1 | 820.67 | 820.67 | 820.67 | 56.2 | 78.6 | -5.86 | -5.86 | -5.86 |
| 9 | 666.10 | 1468.5 | 635.50 | 635.50 | 635.50 | 15.3 | 16.3 | -4.59 | -4.59 | -4.59 |
| 10 | 911.16 | 714.0 | 840.75 | 841.12 | 842.75 | 241.2 | 246.7 | -7.73 | -7.69 | -7.51 |
| 11 | 819.36 | 396.4 | 818.87 | 821.04 | 829.84 | 172.4 | 199.8 | -0.06 | 0.21 | 1.28 |
| 12 | 651.58 | 268.1 | 626.37 | 629.07 | 634.65 | 46.2 | 48.2 | -3.87 | -3.45 | -2.60 |
| 13 | 2928.34 | 1639.1 | 2739.80 | 2739.80 | 2739.80 | 235.4 | 308.8 | -6.44 | -6.44 | -6.44 |
| 14 | 1559.64 | 3451.6 | 1466.84 | 1472.26 | 1481.07 | 623.8 | 642.8 | -5.95 | -5.60 | -5.04 |
| 15 | 1452.34 | 2327.4 | 1367.58 | 1405.48 | 1426.51 | 621.0 | 656.8 | -5.84 | -3.23 | -1.78 |
| 16 | 707.85 | 2550.3 | 698.92 | 698.92 | 698.92 | 12.8 | 14.8 | -1.26 | -1.26 | -1.26 |
| 17 | 920.87 | 2142.5 | 868.59 | 870.33 | 874.24 | 11.8 | 14.9 | -5.68 | -5.49 | -5.06 |
| 18 | 1400.52 | 1452.9 | 1255.64 | 1261.07 | 1262.88 | 2122.2 | 2209.8 | -10.34 | -9.96 | -9.83 |
| 19 | 871.29 | 1822.3 | 777.18 | 781.29 | 784.77 | 614.3 | 623.6 | -10.80 | -10.33 | -9.93 |
| 20 | 732.12 | 790.0 | 604.28 | 611.26 | 617.13 | 3762.3 | 3901.0 | -17.46 | -16.51 | -15.71 |
| 21 | 1275.20 | 2370.3 | 1110.09 | 1124.55 | 1136.59 | 5140.0 | 5180.6 | -12.95 | -11.81 | -10.87 |
| 22 | 1277.94 | 1611.3 | 1194.18 | 1197.43 | 1207.22 | 2233.6 | 2290.3 | -6.55 | -6.30 | -5.53 |
| 23 | 1258.16 | 6725.6 | 1158.51 | 1171.77 | 1178.66 | 3693.4 | 3727.6 | -7.92 | -6.87 | -6.32 |
| 24 | 1307.09 | 6619.3 | 1136.80 | 1148.70 | 1160.00 | 1762.8 | 1791.5 | -13.03 | -12.12 | -11.25 |
| 25 | 1570.72 | 5630.9 | 1429.64 | 1436.32 | 1444.02 | 8619.7 | 8817.1 | -8.98 | -8.56 | -8.07 |
| 26 | 1847.95 | 4123.7 | 1611.78 | 1616.99 | 1624.01 | 6651.2 | 6904.3 | -12.78 | -12.50 | -12.12 |
| 27 | 1747.52 | 7127.2 | 1560.70 | 1573.50 | 1600.30 | 10325.8 | 10483.9 | -10.69 | -9.96 | -8.42 |
| AVG | 1042.26 | 2058.9 | 960.87 | 966.66 | 972.54 | 1746.6 | 1793.1 | -6.98 | -6.43 | -5.89 |

Table B.1: Detailed Results for $3|R|N|1|75|L$ with $bot = 0$

| | TS | | ACO | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $Inst.$ | $z$ | $sec_h$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 316.32 | 129.5 | 305.32 | 305.32 | 305.32 | 10.6 | 16.2 | -3.48 | -3.48 | -3.48 |
| 2 | 350.58 | 5.3 | 334.96 | 334.96 | 334.96 | 0.2 | 0.7 | -4.46 | -4.46 | -4.46 |
| 3 | 447.73 | 461.1 | 404.96 | 413.90 | 418.67 | 172.2 | 184.8 | -9.55 | -7.56 | -6.49 |
| 4 | 448.48 | 181.1 | 440.68 | 440.68 | 440.68 | 4.1 | 5.7 | -1.74 | -1.74 | -1.74 |
| 5 | 464.24 | 75.8 | 450.93 | 451.67 | 453.99 | 24.0 | 31.9 | -2.87 | -2.71 | -2.21 |
| 6 | 504.46 | 1167.9 | 498.32 | 501.35 | 504.39 | 16.3 | 17.5 | -1.22 | -0.62 | -0.01 |
| 7 | 831.66 | 181.1 | 785.14 | 787.00 | 789.38 | 47.0 | 56.2 | -5.59 | -5.37 | -5.08 |
| 8 | 871.77 | 156.1 | 822.06 | 822.06 | 822.06 | 66.6 | 73.1 | -5.70 | -5.70 | -5.70 |
| 9 | 666.10 | 1468.5 | 635.50 | 642.91 | 662.67 | 16.4 | 17.3 | -4.59 | -3.48 | -0.51 |
| 10 | 911.16 | 714.0 | 844.53 | 847.41 | 848.13 | 221.2 | 234.0 | -7.31 | -7.00 | -6.92 |
| 11 | 819.36 | 396.4 | 817.09 | 817.21 | 817.68 | 138.6 | 185.9 | -0.28 | -0.26 | -0.21 |
| 12 | 651.58 | 268.1 | 638.42 | 639.39 | 646.00 | 41.8 | 46.6 | -2.02 | -1.87 | -0.86 |
| 13 | 2928.34 | 1639.1 | 2741.92 | 2751.64 | 2755.81 | 246.8 | 309.2 | -6.37 | -6.03 | -5.89 |
| 14 | 1559.64 | 3451.6 | 1475.27 | 1486.75 | 1495.69 | 629.8 | 639.0 | -5.41 | -4.67 | -4.10 |
| 15 | 1452.34 | 2327.4 | 1365.15 | 1419.05 | 1452.76 | 556.4 | 618.5 | -6.00 | -2.29 | 0.03 |
| 16 | 707.85 | 2550.3 | 698.92 | 698.92 | 698.92 | 13.3 | 15.6 | -1.26 | -1.26 | -1.26 |
| 17 | 920.87 | 2142.5 | 870.96 | 872.55 | 874.91 | 12.2 | 15.5 | -5.42 | -5.25 | -4.99 |
| 18 | 1400.52 | 1452.9 | 1257.62 | 1264.29 | 1273.28 | 2273.8 | 2289.8 | -10.20 | -9.73 | -9.09 |
| 19 | 871.29 | 1822.3 | 779.75 | 781.92 | 787.33 | 565.4 | 578.8 | -10.51 | -10.26 | -9.64 |
| 20 | 732.12 | 790.0 | 610.13 | 616.95 | 624.46 | 3794.8 | 3837.8 | -16.66 | -15.73 | -14.71 |
| 21 | 1275.20 | 2370.3 | 1134.75 | 1143.08 | 1152.35 | 5396.0 | 5498.4 | -11.01 | -10.36 | -9.63 |
| 22 | 1277.94 | 1611.3 | 1200.63 | 1212.38 | 1224.43 | 2371.7 | 2411.5 | -6.05 | -5.13 | -4.19 |
| 23 | 1258.16 | 6725.6 | 1163.80 | 1168.35 | 1173.13 | 3788.0 | 3847.9 | -7.50 | -7.14 | -6.76 |
| 24 | 1307.09 | 6619.3 | 1153.55 | 1163.57 | 1178.88 | 1970.1 | 1992.2 | -11.75 | -10.98 | -9.81 |
| 25 | 1570.72 | 5630.9 | 1439.80 | 1455.63 | 1465.01 | 8898.0 | 9034.7 | -8.34 | -7.33 | -6.73 |
| 26 | 1847.95 | 4123.7 | 1617.64 | 1622.72 | 1628.64 | 6804.1 | 6976.4 | -12.46 | -12.19 | -11.87 |
| 27 | 1747.52 | 7127.2 | 1571.21 | 1595.17 | 1635.07 | 10482.0 | 10713.8 | -10.09 | -8.72 | -6.43 |
| AVG | 1042.26 | 2058.9 | 965.15 | 972.48 | 980.17 | 1798.6 | 1838.8 | -6.59 | -5.97 | -5.29 |

Table B.2: Detailed Results for $3|R|N|1|75|L$ with $bot = 1$

| | TS | | ACO | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $Inst.$ | $z$ | $sec_h$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 316.32 | 1.8 | 310.00 | 310.42 | 310.84 | 9.1 | 11.3 | -2.00 | -1.87 | -1.73 |
| 2 | 334.96 | 13.0 | 334.96 | 334.96 | 334.96 | 0.1 | 0.5 | 0.00 | 0.00 | 0.00 |
| 3 | 430.02 | 1137.9 | 380.87 | 380.87 | 380.87 | 50.1 | 72.1 | -11.43 | -11.43 | -11.43 |
| 4 | 440.68 | 62.0 | 440.68 | 440.68 | 440.68 | 3.9 | 4.6 | 0.00 | 0.00 | 0.00 |
| 5 | 462.59 | 60.9 | 439.65 | 440.33 | 445.52 | 27.2 | 31.1 | -4.96 | -4.81 | -3.69 |
| 6 | 498.32 | 90.6 | 495.85 | 495.89 | 496.29 | 8.6 | 10.9 | -0.50 | -0.49 | -0.41 |
| 7 | 801.03 | 112.1 | 761.36 | 763.33 | 781.10 | 48.0 | 56.3 | -4.95 | -4.71 | -2.49 |
| 8 | 864.54 | 686.0 | 816.33 | 816.33 | 816.33 | 47.3 | 59.1 | -5.58 | -5.58 | -5.58 |
| 9 | 677.06 | 8.5 | 632.70 | 632.70 | 632.70 | 13.1 | 14.3 | -6.55 | -6.55 | -6.55 |
| 10 | 843.33 | 3109.3 | 832.58 | 833.79 | 834.33 | 188.2 | 193.7 | -1.27 | -1.13 | -1.07 |
| 11 | 819.36 | 1135.4 | 800.09 | 801.98 | 804.81 | 152.6 | 167.4 | -2.35 | -2.12 | -1.78 |
| 12 | 669.16 | 3042.4 | 614.24 | 614.31 | 614.59 | 26.9 | 28.9 | -8.21 | -8.20 | -8.16 |
| 13 | 2753.91 | 3371.4 | 2649.84 | 2670.54 | 2692.28 | 269.9 | 311.1 | -3.78 | -3.03 | -2.24 |
| 14 | 1518.26 | 2599.5 | 1438.25 | 1450.03 | 1458.78 | 537.5 | 573.0 | -5.27 | -4.49 | -3.92 |
| 15 | 1414.19 | 574.8 | 1343.34 | 1351.57 | 1357.28 | 682.6 | 699.9 | -5.01 | -4.43 | -4.02 |
| 16 | 711.11 | 3080.6 | 698.92 | 698.92 | 698.92 | 11.1 | 12.9 | -1.71 | -1.71 | -1.71 |
| 17 | 920.87 | 2138.3 | 868.59 | 871.75 | 874.91 | 9.8 | 12.4 | -5.68 | -5.33 | -4.99 |
| 18 | 1433.51 | 3100.9 | 1202.61 | 1205.29 | 1209.59 | 1681.4 | 1705.7 | -16.11 | -15.92 | -15.62 |
| 19 | 853.05 | 3768.5 | 754.33 | 758.40 | 760.00 | 527.0 | 564.5 | -11.57 | -11.10 | -10.91 |
| 20 | 653.47 | 2364.9 | 572.85 | 576.26 | 578.43 | 3299.2 | 3339.6 | -12.34 | -11.82 | -11.48 |
| 21 | 1185.67 | 6244.3 | 1085.57 | 1092.61 | 1100.50 | 3728.1 | 3894.0 | -8.44 | -7.85 | -7.18 |
| 22 | 1243.22 | 3562.1 | 1170.30 | 1176.67 | 1181.63 | 1956.1 | 1998.3 | -5.87 | -5.35 | -4.95 |
| 23 | 1248.25 | 648.4 | 1115.34 | 1125.75 | 1140.77 | 2385.9 | 2441.8 | -10.65 | -9.81 | -8.61 |
| 24 | 1187.68 | 3896.4 | 1128.12 | 1131.25 | 1140.33 | 902.8 | 937.1 | -5.01 | -4.75 | -3.99 |
| 25 | 1673.08 | 7187.8 | 1406.11 | 1418.26 | 1435.44 | 7581.3 | 7732.4 | -15.96 | -15.23 | -14.20 |
| 26 | 1775.52 | 6887.7 | 1596.15 | 1608.30 | 1613.44 | 6395.5 | 6565.0 | -10.10 | -9.42 | -9.13 |
| 27 | 1662.10 | 6205.3 | 1498.53 | 1514.96 | 1535.35 | 5556.4 | 5691.1 | -9.84 | -8.85 | -7.63 |
| AVG | 1014.49 | 2410.8 | 940.30 | 945.04 | 950.76 | 1337.0 | 1375.1 | -6.49 | -6.15 | -5.68 |

Table B.3: Detailed Results for $3|R|N|0|75|L$ with $bot = 0$

| | TS | | ACO | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $Inst.$ | $z$ | $sec_h$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 316.32 | 1.8 | 310.84 | 310.84 | 310.84 | 10.5 | 15.0 | -1.73 | -1.73 | -1.73 |
| 2 | 334.96 | 13.0 | 334.96 | 334.96 | 334.96 | 0.1 | 0.5 | 0.00 | 0.00 | 0.00 |
| 3 | 430.02 | 1137.9 | 396.27 | 397.70 | 402.22 | 66.6 | 96.6 | -7.85 | -7.52 | -6.46 |
| 4 | 440.68 | 62.0 | 440.68 | 440.68 | 440.68 | 3.2 | 4.7 | 0.00 | 0.00 | 0.00 |
| 5 | 462.59 | 60.9 | 439.65 | 440.53 | 443.61 | 21.1 | 29.3 | -4.96 | -4.77 | -4.10 |
| 6 | 498.32 | 90.6 | 498.16 | 499.56 | 500.87 | 9.4 | 12.1 | -0.03 | 0.25 | 0.51 |
| 7 | 801.03 | 112.1 | 776.56 | 779.28 | 781.22 | 41.7 | 54.1 | -3.05 | -2.72 | -2.47 |
| 8 | 864.54 | 686.0 | 823.65 | 823.65 | 823.65 | 33.5 | 61.9 | -4.73 | -4.73 | -4.73 |
| 9 | 677.06 | 8.5 | 632.70 | 632.70 | 632.70 | 12.9 | 14.6 | -6.55 | -6.55 | -6.55 |
| 10 | 843.33 | 3109.3 | 829.78 | 832.05 | 836.14 | 197.2 | 205.6 | -1.61 | -1.34 | -0.85 |
| 11 | 819.36 | 1135.4 | 796.47 | 800.02 | 801.54 | 154.7 | 163.1 | -2.79 | -2.36 | -2.17 |
| 12 | 669.16 | 3042.4 | 614.24 | 614.31 | 614.59 | 25.7 | 27.4 | -8.21 | -8.20 | -8.16 |
| 13 | 2753.91 | 3371.4 | 2699.81 | 2699.81 | 2699.81 | 121.3 | 269.6 | -1.96 | -1.96 | -1.96 |
| 14 | 1518.26 | 2599.5 | 1398.57 | 1426.99 | 1447.07 | 519.7 | 575.8 | -7.88 | -6.01 | -4.69 |
| 15 | 1414.19 | 574.8 | 1351.91 | 1365.88 | 1388.99 | 595.4 | 644.1 | -4.40 | -3.42 | -1.78 |
| 16 | 711.11 | 3080.6 | 698.92 | 698.92 | 698.92 | 11.4 | 13.4 | -1.71 | -1.71 | -1.71 |
| 17 | 920.87 | 2138.3 | 866.40 | 870.02 | 874.91 | 10.3 | 12.9 | -5.92 | -5.52 | -4.99 |
| 18 | 1433.51 | 3100.9 | 1215.92 | 1223.14 | 1227.68 | 1793.2 | 1811.5 | -15.18 | -14.68 | -14.36 |
| 19 | 853.05 | 3768.5 | 765.81 | 769.51 | 775.45 | 548.8 | 566.9 | -10.23 | -9.79 | -9.10 |
| 20 | 653.47 | 2364.9 | 579.76 | 580.95 | 585.91 | 3205.3 | 3238.4 | -11.28 | -11.10 | -10.34 |
| 21 | 1185.67 | 6244.3 | 1069.31 | 1072.86 | 1079.46 | 3768.4 | 3805.1 | -9.81 | -9.51 | -8.96 |
| 22 | 1243.22 | 3562.1 | 1182.95 | 1187.41 | 1191.14 | 2058.7 | 2108.7 | -4.85 | -4.49 | -4.19 |
| 23 | 1248.25 | 648.4 | 1111.91 | 1120.10 | 1134.37 | 2357.9 | 2391.3 | -10.92 | -10.27 | -9.12 |
| 24 | 1187.68 | 3896.4 | 1122.12 | 1132.61 | 1141.00 | 1028.2 | 1051.8 | -5.52 | -4.64 | -3.93 |
| 25 | 1673.08 | 7187.8 | 1420.58 | 1430.10 | 1441.85 | 7825.3 | 7953.3 | -15.09 | -14.52 | -13.82 |
| 26 | 1775.52 | 6887.7 | 1593.70 | 1605.62 | 1621.45 | 6171.0 | 6336.0 | -10.24 | -9.57 | -8.68 |
| 27 | 1662.10 | 6205.3 | 1497.32 | 1508.19 | 1514.24 | 5529.2 | 5668.5 | -9.91 | -9.26 | -8.90 |
| AVG | 1014.49 | 2410.8 | 943.29 | 948.09 | 953.53 | 1337.8 | 1375.3 | -6.16 | -5.78 | -5.31 |

Table B.4: Detailed Results for $3|R|N|0|75|L$ with $bot = 1$

| | TS | | ACO | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $Inst.$ | $z$ | $sec_h$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 301.74 | 40.2 | 301.74 | 301.74 | 301.74 | 3.5 | 5.0 | 0.00 | 0.00 | 0.00 |
| 2 | 334.96 | 3.7 | 334.96 | 334.96 | 334.96 | 0.1 | 0.4 | 0.00 | 0.00 | 0.00 |
| 3 | 392.44 | 639.1 | 362.27 | 362.27 | 362.27 | 20.0 | 25.1 | -7.69 | -7.69 | -7.69 |
| 4 | 430.88 | 12.3 | 430.89 | 430.89 | 430.89 | 1.9 | 3.0 | 0.00 | 0.00 | 0.00 |
| 5 | 422.90 | 461.5 | 436.73 | 436.73 | 436.73 | 21.2 | 27.0 | 3.27 | 3.27 | 3.27 |
| 6 | 495.85 | 204.5 | 498.16 | 498.38 | 500.43 | 6.7 | 8.4 | 0.47 | 0.51 | 0.92 |
| 7 | 732.51 | 7.1 | 747.30 | 747.30 | 747.30 | 16.6 | 23.4 | 2.02 | 2.02 | 2.02 |
| 8 | 810.65 | 180.9 | 778.88 | 778.88 | 778.88 | 6.9 | 14.8 | -3.92 | -3.92 | -3.92 |
| 9 | 630.13 | 1518.7 | 630.13 | 634.51 | 647.04 | 4.9 | 6.4 | 0.00 | 0.70 | 2.68 |
| 10 | 827.11 | 1541.4 | 799.84 | 801.62 | 805.65 | 97.3 | 109.2 | -3.30 | -3.08 | -2.59 |
| 11 | 767.22 | 3440.3 | 769.94 | 769.94 | 769.94 | 94.1 | 106.6 | 0.35 | 0.35 | 0.35 |
| 12 | 635.15 | 3306.0 | 612.81 | 613.99 | 614.59 | 11.6 | 13.8 | -3.52 | -3.33 | -3.24 |
| 13 | 2549.68 | 1308.9 | 2649.25 | 2674.69 | 2690.30 | 139.9 | 203.2 | 3.91 | 4.90 | 5.52 |
| 14 | 1389.31 | 2621.9 | 1338.92 | 1346.25 | 1364.09 | 553.7 | 588.8 | -3.63 | -3.10 | -1.82 |
| 15 | 1346.44 | 2489.3 | 1281.01 | 1315.69 | 1321.00 | 525.1 | 556.4 | -4.86 | -2.28 | -1.89 |
| 16 | 703.57 | 2673.6 | 698.61 | 698.61 | 698.61 | 3.9 | 6.1 | -0.70 | -0.70 | -0.70 |
| 17 | 906.42 | 697.5 | 866.40 | 871.39 | 874.24 | 5.3 | 8.8 | -4.42 | -3.86 | -3.55 |
| 18 | 1331.71 | 2000.8 | 1187.39 | 1197.00 | 1201.50 | 1043.8 | 1194.1 | -10.84 | -10.12 | -9.78 |
| 19 | 781.77 | 67.8 | 730.83 | 736.46 | 738.08 | 318.4 | 342.1 | -6.52 | -5.80 | -5.59 |
| 20 | 629.77 | 1355.3 | 557.75 | 562.24 | 564.00 | 1428.2 | 1514.6 | -11.44 | -10.72 | -10.44 |
| 21 | 1210.78 | 2771.5 | 1039.14 | 1045.60 | 1052.22 | 1738.0 | 1805.9 | -14.18 | -13.64 | -13.10 |
| 22 | 1160.67 | 5004.2 | 1109.30 | 1112.46 | 1125.98 | 1176.1 | 1212.4 | -4.43 | -4.15 | -2.99 |
| 23 | 1153.60 | 1003.1 | 1057.92 | 1062.85 | 1065.54 | 1196.6 | 1230.3 | -8.29 | -7.87 | -7.63 |
| 24 | 1154.51 | 1998.3 | 1087.43 | 1088.79 | 1089.84 | 349.8 | 373.2 | -5.81 | -5.69 | -5.60 |
| 25 | 1420.51 | 3146.1 | 1296.65 | 1308.84 | 1326.61 | 4282.0 | 4487.5 | -8.72 | -7.86 | -6.61 |
| 26 | 1605.54 | 516.6 | 1540.39 | 1550.31 | 1562.11 | 2712.0 | 2858.5 | -4.06 | -3.44 | -2.71 |
| 27 | 1556.33 | 7154.0 | 1434.56 | 1456.28 | 1468.66 | 2510.8 | 2629.5 | -7.82 | -6.43 | -5.63 |
| AVG | 951.19 | 1709.8 | 910.34 | 916.25 | 921.23 | 676.6 | 716.8 | -3.86 | -3.41 | -2.99 |

Table B.5: Detailed Results for $3|U|N|1|75|L$ with $bot = 0$

| | TS | | ACO | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $Inst.$ | $z$ | $sec_h$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 301.74 | 40.2 | 301.74 | 301.74 | 301.74 | 3.4 | 4.7 | 0.00 | 0.00 | 0.00 |
| 2 | 334.96 | 3.7 | 334.96 | 334.96 | 334.96 | 0.1 | 0.4 | 0.00 | 0.00 | 0.00 |
| 3 | 392.44 | 639.1 | 362.27 | 362.27 | 362.27 | 17.7 | 22.3 | -7.69 | -7.69 | -7.69 |
| 4 | 430.88 | 12.3 | 430.89 | 430.89 | 430.89 | 1.8 | 3.1 | 0.00 | 0.00 | 0.00 |
| 5 | 422.90 | 461.5 | 437.27 | 437.27 | 437.27 | 19.6 | 25.6 | 3.40 | 3.40 | 3.40 |
| 6 | 495.85 | 204.5 | 498.16 | 498.30 | 498.65 | 6.3 | 8.4 | 0.47 | 0.49 | 0.56 |
| 7 | 732.51 | 7.1 | 745.07 | 745.07 | 745.07 | 19.4 | 25.2 | 1.71 | 1.71 | 1.71 |
| 8 | 810.65 | 180.9 | 778.88 | 778.88 | 778.88 | 8.6 | 15.4 | -3.92 | -3.92 | -3.92 |
| 9 | 630.13 | 1518.7 | 630.13 | 635.11 | 647.04 | 4.2 | 5.5 | 0.00 | 0.79 | 2.68 |
| 10 | 827.11 | 1541.4 | 803.92 | 803.92 | 803.92 | 89.3 | 103.2 | -2.80 | -2.80 | -2.80 |
| 11 | 767.22 | 3440.3 | 771.05 | 771.05 | 771.05 | 57.2 | 103.3 | 0.50 | 0.50 | 0.50 |
| 12 | 635.15 | 3306.0 | 614.24 | 614.24 | 614.24 | 13.8 | 16.4 | -3.29 | -3.29 | -3.29 |
| 13 | 2549.68 | 1308.9 | 2646.75 | 2665.97 | 2695.66 | 164.4 | 193.7 | 3.81 | 4.56 | 5.73 |
| 14 | 1389.31 | 2621.9 | 1338.92 | 1349.12 | 1360.42 | 544.0 | 571.1 | -3.63 | -2.89 | -2.08 |
| 15 | 1346.44 | 2489.3 | 1295.82 | 1307.14 | 1315.32 | 517.9 | 596.9 | -3.76 | -2.92 | -2.31 |
| 16 | 703.57 | 2673.6 | 698.61 | 698.61 | 698.61 | 4.1 | 6.0 | -0.70 | -0.70 | -0.70 |
| 17 | 906.42 | 697.5 | 866.40 | 870.16 | 873.82 | 4.9 | 8.8 | -4.42 | -4.00 | -3.60 |
| 18 | 1331.71 | 2000.8 | 1183.86 | 1188.14 | 1203.10 | 1205.1 | 1238.1 | -11.10 | -10.78 | -9.66 |
| 19 | 781.77 | 67.8 | 717.25 | 724.80 | 729.60 | 335.6 | 361.9 | -8.25 | -7.29 | -6.67 |
| 20 | 629.77 | 1355.3 | 567.13 | 568.62 | 570.82 | 1512.2 | 1561.3 | -9.95 | -9.71 | -9.36 |
| 21 | 1210.78 | 2771.5 | 1025.53 | 1029.72 | 1033.97 | 1746.1 | 1813.7 | -15.30 | -14.95 | -14.60 |
| 22 | 1160.67 | 5004.2 | 1101.16 | 1102.38 | 1104.33 | 1190.6 | 1225.4 | -5.13 | -5.02 | -4.85 |
| 23 | 1153.60 | 1003.1 | 1061.00 | 1065.93 | 1072.28 | 1128.2 | 1167.2 | -8.03 | -7.60 | -7.05 |
| 24 | 1154.51 | 1998.3 | 1089.41 | 1090.98 | 1095.03 | 336.8 | 357.1 | -5.64 | -5.50 | -5.15 |
| 25 | 1420.51 | 3146.1 | 1291.57 | 1307.41 | 1319.82 | 4416.3 | 4529.2 | -9.08 | -7.96 | -7.09 |
| 26 | 1605.54 | 516.6 | 1537.74 | 1545.34 | 1555.94 | 2767.0 | 2927.4 | -4.22 | -3.75 | -3.09 |
| 27 | 1556.33 | 7154.0 | 1431.56 | 1440.59 | 1456.71 | 2457.2 | 2623.7 | -8.02 | -7.44 | -6.40 |
| AVG | 951.19 | 1709.8 | 909.68 | 913.65 | 918.94 | 687.8 | 722.8 | -3.89 | -3.58 | -3.18 |

Table B.6: Detailed Results for $3|U|N|1|75|L$ with $bot = 1$

| | TS | | ACO | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $Inst.$ | $z$ | $sec_h$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 301.74 | 4.7 | 302.02 | 302.17 | 303.51 | 4.8 | 6.4 | 0.09 | 0.14 | 0.59 |
| 2 | 334.96 | 0.3 | 334.96 | 334.96 | 334.96 | 0.1 | 0.4 | 0.00 | 0.00 | 0.00 |
| 3 | 388.10 | 33.7 | 390.52 | 391.18 | 391.37 | 38.2 | 54.2 | 0.62 | 0.79 | 0.84 |
| 4 | 430.88 | 22.4 | 440.91 | 440.91 | 440.91 | 2.4 | 3.8 | 2.33 | 2.33 | 2.33 |
| 5 | 435.93 | 54.5 | 441.87 | 441.87 | 441.87 | 19.4 | 37.1 | 1.36 | 1.36 | 1.36 |
| 6 | 498.16 | 66.9 | 495.85 | 495.85 | 495.85 | 7.3 | 8.5 | -0.46 | -0.46 | -0.46 |
| 7 | 762.63 | 26.8 | 771.07 | 771.07 | 771.07 | 50.9 | 64.4 | 1.11 | 1.11 | 1.11 |
| 8 | 799.38 | 71.7 | 803.77 | 803.77 | 803.77 | 42.0 | 52.0 | 0.55 | 0.55 | 0.55 |
| 9 | 640.94 | 764.3 | 631.09 | 631.09 | 631.09 | 12.2 | 13.1 | -1.54 | -1.54 | -1.54 |
| 10 | 803.18 | 1881.6 | 769.36 | 780.25 | 802.16 | 213.6 | 221.6 | -4.21 | -2.85 | -0.13 |
| 11 | 772.55 | 2781.2 | 749.66 | 750.81 | 756.26 | 159.2 | 219.5 | -2.96 | -2.81 | -2.11 |
| 12 | 616.95 | 1799.4 | 614.24 | 614.52 | 614.59 | 19.3 | 22.0 | -0.44 | -0.39 | -0.38 |
| 13 | 2591.84 | 104.1 | 2560.16 | 2593.43 | 2609.51 | 271.5 | 294.4 | -1.22 | 0.06 | 0.68 |
| 14 | 1348.19 | 1972.6 | 1348.92 | 1363.92 | 1376.75 | 748.2 | 839.0 | 0.05 | 1.17 | 2.12 |
| 15 | 1352.20 | 2830.9 | 1324.59 | 1325.27 | 1325.41 | 672.2 | 709.0 | -2.04 | -1.99 | -1.98 |
| 16 | 704.80 | 1339.4 | 698.92 | 698.92 | 698.92 | 7.9 | 9.9 | -0.83 | -0.83 | -0.83 |
| 17 | 920.87 | 2129.1 | 866.40 | 870.02 | 874.24 | 6.9 | 10.1 | -5.92 | -5.52 | -5.06 |
| 18 | 1245.57 | 1848.4 | 1202.52 | 1204.26 | 1208.39 | 2200.7 | 2234.0 | -3.46 | -3.32 | -2.98 |
| 19 | 734.77 | 6772.3 | 736.89 | 739.96 | 743.07 | 558.7 | 574.0 | 0.29 | 0.71 | 1.13 |
| 20 | 610.63 | 779.9 | 559.29 | 566.47 | 582.76 | 3153.2 | 3225.2 | -8.41 | -7.23 | -4.56 |
| 21 | 1188.60 | 4715.0 | 1051.43 | 1056.78 | 1065.65 | 4198.2 | 4252.4 | -11.54 | -11.09 | -10.34 |
| 22 | 1172.20 | 4950.0 | 1115.38 | 1123.47 | 1139.52 | 1980.0 | 2019.4 | -4.85 | -4.16 | -2.79 |
| 23 | 1106.43 | 2446.2 | 1070.84 | 1074.77 | 1083.04 | 2346.8 | 2379.1 | -3.22 | -2.86 | -2.11 |
| 24 | 1113.80 | 5720.5 | 1097.16 | 1100.47 | 1103.07 | 754.0 | 775.7 | -1.49 | -1.20 | -0.96 |
| 25 | 1375.99 | 2159.2 | 1334.51 | 1340.36 | 1358.17 | 7979.1 | 8105.5 | -3.01 | -2.59 | -1.30 |
| 26 | 1579.47 | 2351.4 | 1543.69 | 1557.75 | 1573.15 | 5829.1 | 5987.6 | -2.27 | -1.38 | -0.40 |
| 27 | 1536.49 | 3201.9 | 1441.49 | 1457.25 | 1473.58 | 4903.6 | 5033.3 | -6.18 | -5.16 | -4.09 |
| AVG | 939.53 | 1882.5 | 914.72 | 919.69 | 926.02 | 1340.0 | 1376.0 | -2.14 | -1.75 | -1.16 |

Table B.7: Detailed Results for $3|R|N|1|0|L$ with $bot = 0$

| | TS | | ACO | | | | | %*gap* ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| *Inst.* | $z$ | $sec_h$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 301.74 | 4.7 | 301.66 | 301.66 | 301.66 | 2.7 | 5.7 | -0.03 | -0.03 | -0.03 |
| 2 | 334.96 | 0.3 | 334.96 | 334.96 | 334.96 | 0.1 | 0.4 | 0.00 | 0.00 | 0.00 |
| 3 | 388.10 | 33.7 | 365.08 | 370.39 | 378.35 | 31.5 | 42.7 | -5.93 | -4.56 | -2.51 |
| 4 | 430.88 | 22.4 | 430.89 | 430.89 | 430.89 | 2.5 | 3.4 | 0.00 | 0.00 | 0.00 |
| 5 | 435.93 | 54.5 | 428.98 | 429.32 | 429.66 | 38.1 | 45.5 | -1.59 | -1.52 | -1.44 |
| 6 | 498.16 | 66.9 | 495.85 | 495.85 | 495.85 | 5.1 | 7.0 | -0.46 | -0.46 | -0.46 |
| 7 | 762.63 | 26.8 | 752.81 | 752.81 | 752.81 | 48.9 | 67.7 | -1.29 | -1.29 | -1.29 |
| 8 | 799.38 | 71.7 | 788.74 | 788.74 | 788.74 | 49.9 | 58.1 | -1.33 | -1.33 | -1.33 |
| 9 | 640.94 | 764.3 | 630.13 | 641.35 | 647.04 | 10.3 | 13.2 | -1.69 | 0.06 | 0.95 |
| 10 | 803.18 | 1881.6 | 762.69 | 763.35 | 768.86 | 266.2 | 269.7 | -5.04 | -4.96 | -4.27 |
| 11 | 772.55 | 2781.2 | 723.60 | 723.98 | 727.11 | 225.7 | 240.4 | -6.34 | -6.29 | -5.88 |
| 12 | 616.95 | 1799.4 | 614.24 | 614.24 | 614.24 | 14.1 | 17.4 | -0.44 | -0.44 | -0.44 |
| 13 | 2591.84 | 104.1 | 2559.67 | 2559.74 | 2560.04 | 323.8 | 367.6 | -1.24 | -1.24 | -1.23 |
| 14 | 1348.19 | 1972.6 | 1337.26 | 1337.26 | 1337.26 | 1061.3 | 1071.7 | -0.81 | -0.81 | -0.81 |
| 15 | 1352.20 | 2830.9 | 1317.28 | 1320.83 | 1325.36 | 884.3 | 983.8 | -2.58 | -2.32 | -1.98 |
| 16 | 704.80 | 1339.4 | 698.92 | 698.92 | 698.92 | 6.7 | 8.9 | -0.83 | -0.83 | -0.83 |
| 17 | 920.87 | 2129.1 | 863.27 | 864.06 | 871.24 | 7.3 | 9.7 | -6.25 | -6.17 | -5.39 |
| 18 | 1245.57 | 1848.4 | 1164.16 | 1175.92 | 1188.92 | 3173.4 | 3231.9 | -6.54 | -5.59 | -4.55 |
| 19 | 734.77 | 6772.3 | 721.81 | 729.26 | 730.15 | 684.3 | 713.8 | -1.76 | -0.75 | -0.63 |
| 20 | 610.63 | 779.9 | 547.93 | 555.34 | 562.28 | 3795.7 | 3847.0 | -10.27 | -9.05 | -7.92 |
| 21 | 1188.60 | 4715.0 | 1008.19 | 1014.79 | 1031.78 | 4832.0 | 4873.5 | -15.18 | -14.62 | -13.19 |
| 22 | 1172.20 | 4950.0 | 1092.81 | 1098.11 | 1107.56 | 2291.4 | 2326.9 | -6.77 | -6.32 | -5.51 |
| 23 | 1106.43 | 2446.2 | 1052.31 | 1059.85 | 1070.90 | 2699.7 | 2757.4 | -4.89 | -4.21 | -3.21 |
| 24 | 1113.80 | 5720.5 | 1082.45 | 1088.36 | 1094.48 | 766.7 | 785.7 | -2.81 | -2.28 | -1.73 |
| 25 | 1375.99 | 2159.2 | 1307.80 | 1317.83 | 1334.87 | 9057.0 | 9188.7 | -4.96 | -4.23 | -2.99 |
| 26 | 1579.47 | 2351.4 | 1523.19 | 1537.01 | 1545.62 | 6464.8 | 6631.4 | -3.56 | -2.69 | -2.14 |
| 27 | 1536.49 | 3201.9 | 1422.66 | 1432.11 | 1442.90 | 5042.0 | 5162.2 | -7.41 | -6.79 | -6.09 |
| AVG | 939.53 | 1882.5 | 901.09 | 905.07 | 910.09 | 1547.6 | 1582.6 | -3.70 | -3.29 | -2.77 |

Table B.8: Detailed Results for $3|R|N|1|0|L$ with $bot = 1$

| | TS | | ACO | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $Inst.$ | $z$ | $sec_h$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 297.65 | 3.4 | 297.65 | 297.65 | 297.65 | 1.0 | 1.8 | 0.00 | 0.00 | 0.00 |
| 2 | 334.96 | 0.6 | 334.96 | 334.96 | 334.96 | 0.1 | 0.3 | 0.00 | 0.00 | 0.00 |
| 3 | 362.27 | 448.1 | 362.27 | 362.27 | 362.27 | 16.2 | 19.5 | 0.00 | 0.00 | 0.00 |
| 4 | 430.88 | 11.1 | 430.89 | 430.89 | 430.89 | 0.5 | 1.2 | 0.00 | 0.00 | 0.00 |
| 5 | 395.64 | 0.5 | 406.50 | 406.50 | 406.50 | 9.6 | 20.7 | 2.74 | 2.74 | 2.74 |
| 6 | 495.85 | 14.7 | 495.85 | 495.85 | 495.85 | 1.2 | 1.8 | 0.00 | 0.00 | 0.00 |
| 7 | 742.23 | 1.8 | 732.52 | 732.52 | 732.52 | 18.1 | 28.4 | -1.31 | -1.31 | -1.31 |
| 8 | 735.14 | 104.9 | 735.14 | 735.14 | 735.14 | 13.3 | 19.8 | 0.00 | 0.00 | 0.00 |
| 9 | 630.13 | 977.8 | 630.13 | 630.13 | 630.13 | 3.7 | 5.1 | 0.00 | 0.00 | 0.00 |
| 10 | 717.90 | 410.7 | 711.45 | 711.45 | 711.45 | 92.6 | 106.9 | -0.90 | -0.90 | -0.90 |
| 11 | 718.24 | 208.1 | 718.25 | 718.25 | 718.25 | 81.9 | 121.0 | 0.00 | 0.00 | 0.00 |
| 12 | 614.60 | 1302.7 | 610.23 | 612.63 | 614.24 | 7.5 | 8.7 | -0.71 | -0.32 | -0.06 |
| 13 | 2316.56 | 2317.3 | 2391.77 | 2391.77 | 2391.77 | 174.5 | 311.9 | 3.25 | 3.25 | 3.25 |
| 14 | 1276.60 | 2121.3 | 1214.14 | 1222.17 | 1240.31 | 425.9 | 569.7 | -4.89 | -4.26 | -2.84 |
| 15 | 1196.55 | 2916.4 | 1177.69 | 1182.86 | 1184.71 | 645.0 | 765.4 | -1.58 | -1.14 | -0.99 |
| 16 | 698.61 | 863.0 | 698.61 | 698.61 | 698.61 | 2.8 | 4.5 | 0.00 | 0.00 | 0.00 |
| 17 | 906.42 | 753.2 | 861.79 | 862.18 | 863.27 | 3.1 | 7.0 | -4.92 | -4.88 | -4.76 |
| 18 | 1124.33 | 2198.9 | 1111.77 | 1112.18 | 1112.46 | 1484.6 | 1617.3 | -1.12 | -1.08 | -1.06 |
| 19 | 680.29 | 1390.3 | 668.94 | 671.60 | 672.61 | 414.4 | 437.1 | -1.67 | -1.28 | -1.13 |
| 20 | 529.00 | 7007.5 | 509.83 | 515.39 | 518.09 | 1436.7 | 1521.3 | -3.62 | -2.57 | -2.06 |
| 21 | 1004.40 | 6262.5 | 946.21 | 951.87 | 955.08 | 2105.7 | 2180.9 | -5.79 | -5.23 | -4.91 |
| 22 | 1068.96 | 2078.7 | 1026.05 | 1030.12 | 1033.35 | 1218.4 | 1261.8 | -4.01 | -3.63 | -3.33 |
| 23 | 1012.51 | 4314.1 | 970.96 | 971.05 | 971.95 | 1231.7 | 1277.4 | -4.10 | -4.09 | -4.01 |
| 24 | 1063.61 | 1052.5 | 1047.50 | 1057.39 | 1063.67 | 184.7 | 208.4 | -1.51 | -0.58 | 0.01 |
| 25 | 1371.32 | 500.9 | 1205.10 | 1207.97 | 1209.84 | 3986.1 | 4140.4 | -12.12 | -11.91 | -11.78 |
| 26 | 1557.12 | 1075.0 | 1445.06 | 1453.39 | 1459.31 | 2843.6 | 3079.7 | -7.20 | -6.66 | -6.28 |
| 27 | 1378.52 | 3983.2 | 1327.39 | 1333.16 | 1344.48 | 2208.3 | 2302.5 | -3.71 | -3.29 | -2.47 |
| AVG | 876.31 | 1567.4 | 854.39 | 856.67 | 858.86 | 689.3 | 741.5 | -1.97 | -1.75 | -1.55 |

Table B.9: Detailed Results for $3|U|N|0|0|L$ with $bot = 0$

| | TS | | ACO | | | | | %$gap$ ACO - TS | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $Inst.$ | $z$ | $sec_h$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ | $\overline{sec}_h$ | $\overline{sec}_{tot}$ | $z_{min}$ | $\overline{z}$ | $z_{max}$ |
| 1 | 297.65 | 3.4 | 297.65 | 297.65 | 297.65 | 0.8 | 1.9 | 0.00 | 0.00 | 0.00 |
| 2 | 334.96 | 0.6 | 334.96 | 334.96 | 334.96 | 0.1 | 0.3 | 0.00 | 0.00 | 0.00 |
| 3 | 362.27 | 448.1 | 362.27 | 362.27 | 362.27 | 20.5 | 23.8 | 0.00 | 0.00 | 0.00 |
| 4 | 430.88 | 11.1 | 430.89 | 430.89 | 430.89 | 0.6 | 1.4 | 0.00 | 0.00 | 0.00 |
| 5 | 395.64 | 0.5 | 395.64 | 395.64 | 395.64 | 12.3 | 20.7 | 0.00 | 0.00 | 0.00 |
| 6 | 495.85 | 14.7 | 495.85 | 495.85 | 495.85 | 1.1 | 1.8 | 0.00 | 0.00 | 0.00 |
| 7 | 742.23 | 1.8 | 732.52 | 732.52 | 732.52 | 21.6 | 37.4 | -1.31 | -1.31 | -1.31 |
| 8 | 735.14 | 104.9 | 735.14 | 735.14 | 735.14 | 13.6 | 23.9 | 0.00 | 0.00 | 0.00 |
| 9 | 630.13 | 977.8 | 630.13 | 630.13 | 630.13 | 4.3 | 5.3 | 0.00 | 0.00 | 0.00 |
| 10 | 717.90 | 410.7 | 716.18 | 733.46 | 739.34 | 108.9 | 147.3 | -0.24 | 2.17 | 2.99 |
| 11 | 718.24 | 208.1 | 718.25 | 718.25 | 718.25 | 93.4 | 134.6 | 0.00 | 0.00 | 0.00 |
| 12 | 614.60 | 1302.7 | 610.23 | 613.51 | 614.59 | 6.5 | 9.2 | -0.71 | -0.18 | 0.00 |
| 13 | 2316.56 | 2317.3 | 2319.72 | 2319.72 | 2319.72 | 207.2 | 320.8 | 0.14 | 0.14 | 0.14 |
| 14 | 1276.60 | 2121.3 | 1205.36 | 1215.44 | 1233.96 | 516.8 | 580.1 | -5.58 | -4.79 | -3.34 |
| 15 | 1196.55 | 2916.4 | 1165.87 | 1167.71 | 1171.24 | 760.7 | 836.6 | -2.56 | -2.41 | -2.12 |
| 16 | 698.61 | 863.0 | 698.61 | 698.61 | 698.61 | 2.5 | 4.5 | 0.00 | 0.00 | 0.00 |
| 17 | 906.42 | 753.2 | 861.79 | 862.28 | 862.62 | 3.4 | 6.9 | -4.92 | -4.87 | -4.83 |
| 18 | 1124.33 | 2198.9 | 1096.74 | 1099.48 | 1101.31 | 1750.0 | 2091.1 | -2.45 | -2.21 | -2.05 |
| 19 | 680.29 | 1390.3 | 672.84 | 673.26 | 674.51 | 485.6 | 512.2 | -1.10 | -1.03 | -0.85 |
| 20 | 529.00 | 7007.5 | 512.63 | 516.70 | 519.98 | 2013.5 | 2142.0 | -3.09 | -2.33 | -1.71 |
| 21 | 1004.40 | 6262.5 | 942.19 | 945.70 | 954.21 | 2405.6 | 2455.3 | -6.19 | -5.84 | -5.00 |
| 22 | 1068.96 | 2078.7 | 1021.02 | 1022.26 | 1025.27 | 1391.6 | 1440.8 | -4.48 | -4.37 | -4.09 |
| 23 | 1012.51 | 4314.1 | 977.26 | 977.45 | 978.81 | 1410.1 | 1452.0 | -3.48 | -3.46 | -3.33 |
| 24 | 1063.61 | 1052.5 | 1048.58 | 1052.41 | 1056.03 | 169.1 | 189.6 | -1.41 | -1.05 | -0.71 |
| 25 | 1371.32 | 500.9 | 1198.25 | 1204.40 | 1207.27 | 4953.9 | 5092.3 | -12.62 | -12.17 | -11.96 |
| 26 | 1557.12 | 1075.0 | 1432.82 | 1441.54 | 1453.74 | 3401.0 | 3559.4 | -7.98 | -7.42 | -6.64 |
| 27 | 1378.52 | 3983.2 | 1326.35 | 1329.46 | 1331.02 | 2533.3 | 2674.3 | -3.78 | -3.56 | -3.45 |
| AVG | 876.31 | 1567.4 | 849.62 | 852.10 | 854.65 | 825.5 | 880.2 | -2.29 | -2.03 | -1.79 |

Table B.10: Detailed Results for $3|U|N|0|0|L$ with $bot = 1$

# C Acknowledgement

This project was financially supported by Oestereichische Nationalbank (OENB) under grant #11984.

# Bibliography

Alfa, A. S., S. S. Heragu, and M. Chen (1991). "A 3-opt based simulated annealing algorithm for vehicle routing problems". In: *Computers and Industrial Engineering* 21. Pp. 635–639.

Alvarez-Valdes, R., F. Parreño, and J.M. Tamarit (2008). "Reactive GRASP for the strip-packing problem". In: *Computers & Operations Research* 35. Pp. 1065–1083.

Alvim, A.C.F., F. Glover, C.C. Ribeiro, and D.J. Aloise (2004). "A hybrid improvement heuristic for the one-dimensional bin packing problem". In: *Journal of Heuristics* 10.2. Pp. 205–229.

Beasley, J.E. (1983). "Roue-first cluster-second methods for vehicle routing". In: *Omega* 11. Pp. 403–408.

Berger, Jean and Mohamed Barkaoui (2003). "A Hybrid Genetic Algorithm for the Capacitated Vehicle Routing Problem". In: *Genetic and Evolutionary Computation - GECCO 2003*. Pp. 646–656.

Boschetti, M.A. (2004). "New lower bounds for the three-dimensional finite bin packing problem". In: *Discrete Applied Mathematics* 140. Pp. 241–258.

Bramel, J. and D. Simchi-Levi (2002). "Set-covering-based algorithms for the capacitated VRP". In: *The vehicle routing problem*. Ed. by P. Toth and D. Vigo. Philadelphia: Society for Industrial and Applied Mathematics. Pp. 85–108.

Brugger, B., K. F. Doerner, R. F. Hartl, and M. Reimann (2004). "AntPacking - An Ant Colony Optimization Approach for the One-Dimensional Bin Packing Problem". In: Lecture Notes in Computer Science.

Bullnheimer, B., R.F. Hartl, and Strauss C. (1999). "An improved Ant System algorithm for theVehicle Routing Problem". In: *Annals of Operations Research* 89. Pp. 319–328.

Burke, E.K., G. Kendall, and G. Whitwell (2004). "A New Placement Heuristic for the Orthogonal Stock-Cutting Problem". In: *Operations Research* 35.4. Pp. 655–671.

Chen, C. S., S. M. Lee, and Q. S. Shen (1995). "An analytical model for the container loading problem". In: *European Journal of Operational Research* 80. Pp. 68–76.

Chen, Zhi-Long and Warren B. Powell (1999). "Solving Parallel Machine Scheduling Problems by Column Generation". In: *INFORMS Journal on Computing* 11. Pp. 78–94.

Christofides, N., A. Mingozzi, and P. Toth (1979). “The vehicle routing problem”. In: *Combinatorial Optimization*. Ed. by N. Christofides, A. Mingozzi, P. Toth, and C. Sandi. Chichester: Wiley. Pp. 315–338.

Clarke, G. and J.V. Wright (1964). “Scheduling of vehicles from a central depot to a number of delivery points”. In: *Operations Research* 12. Pp. 568–581.

Cordeau, J. F., M. Gendreau, G. Laporte, J. Y. Potvin, and F. Semet (2002). “A guide to vehicle routing heuristics”. In: *Operational Research Society* 53. Pp. 512–522.

Crainic, T.G., G. Perboli, and R. Tadei (2006). *Extreme-Point-based Heuristics for the Three-Dimensional Bin Packing problem*. Technical Report ”Operations Research”/02/06. DAUIN, Politecnico di Torino.

Dell’Amico, M. and S. Martello (1995). “Optimal Scheduling of Tasks on Identical Parallel Processors”. In: *ORSA Journal on Computing* 7. Pp. 191–200.

— (2002). “A lower bound for the non-oriented two-dimensional bin packing problem”. In: *Discrete Applied Mathematics* 118. Pp. 13–24.

Deneubourg, J. L., S. Aron, Goss S., and J. M. Pasteels (1990). “The self-organizing exploratory pattern of the argentine ant”. In: *Journal of Insect Behavior* 3. Pp. 159–168.

Doerner, K.F., G. Fuellerer, M. Gronalt, R. Hartl, and M. Iori (2007). “Metaheuristics for Vehicle Routing Problems with Loading Constraints”. In: *Networks* 49.4. Pp. 294–307.

Domschke, W. and A. Drexl (1996). *Logistik: Standorte*. Oldenbourg.

Dorigo, Marco and Thomas Stützle (2004). *Ant Colony Optimization*. Massachusetts Institute of Technology.

EuroStat (2008). *Anteil von Straße am gesamten inländischen Güterverkehr*. URL: `http://epp.eurostat.ec.europa.eu/portal/page?_pageid=1996,39140985&_dad=portal&_schema=PORTAL&screen=detailref&language=de&product=Yearlies_new_transport&root=Yearlies_new_transport/G/en033` (visited on 01/2008/08).

Fallahi, A. E., C. Prins, and R. W. Calvo (2008). “A Memetic Algorithm and a Tabu Search for the Multi-Compartment Vehicle Routing Problem”. In: *Computers and Operations Research* 35. Pp. 1725–1741.

Fekete, S.P., E. Koehler, and J. Teich (2001). “Higher-Dimensional Packing with Order Constraints”. In: *Lecture Notes in Computer Science* 2125. Pp. 300–312.

Fekete, S.P. and J. Schepers (1997). *On more-dimensional packing II: Bounds*. Technical Report ZPR97-289. `http://www.zaik.uni-koeln.de/~paper/preprints.html?show=zpr97-289`. Mathematisches Institut, Universität zu Köln.

— (2004). “A General Framework for Bounds for Higher-Dimensional Orthogonal Packing Problems”. In: *Mathematical Methods of Operations Research* 60. Pp. 311–329.

Fekete, S.P. and J. Scherpers (1998). “New classes of lower bounds for bin-packing problem”. In: *Lecture Notes in Computer Science* 1412. Pp. 257–270.

Fekete, S.P., J. Scherpers, and J. van der Veen (2007). “An Exact Algorithm for Higher-Dimensional Orthogonal Packing”. In: *Operation Research* 55. Pp. 59–587.

Fisher, M. L. and R. Jaikumar (1981). “A generalized assignment heuristic for the vehicle routing problem”. In: *Networks* 11. Pp. 109–124.

Fleszar, K. and K. S. Hindi (2002). “New heuristics for one-dimensional bin-packing”. In: *Computers & Operations Research* 29.7. Pp. 821–839.

Frangioni, A., E. Necciari, and M. G. Scutella (2004). “A Multi-Exchange Neighborhood for Minimum Makespan Parallel Machine Scheduling Problems”. In: *Journal of Combinatorial Optimization* 8.2. Pp. 195–220.

Fuellerer, M., K.F. Doerner, R. Hartl, and M. Iori (2007). “Ant Colony Optimization for the Two-dimensional Loading Vehicle Routing Problem”. In: *Computers & Operations Research.* (to appear).

Fukasawa, R., H. Longo, J. Lysgaard, M. Poggi de Arago, M. Reis, E. Uchoa, and R. F. Werneck (2006). “Robust Branch-and-Cut-and-Price for the Capacitated Vehicle Routing Problem”. In: *Mathematical Programming* 106.3. Pp. 491–511.

Gendreau, M., A. Hertz, and G. Laporte (1994). “A tabu search heuristic for the vehicle routing problem”. In: *Management Science* 40. Pp. 1276–1290.

Gendreau, M., G. Laporte, and J. Y. Potvin (2002). “Metaheuristics for the Capacitated VRP”. In: *The Vehicle Routing Problem.* Ed. by P. Toth and D. Vigo. Philadelphia: SIAM Monographs on Discrete Mathematics and Applications. Pp. 129–154.

Gendreau, M., M. Iori, G. Laporte, and S. Martello (2006). “A tabu search algorithm for a routing and container loading problem”. In: *Transportation Science* 40. Pp. 342–350.

— (2008). “A Tabu Search heuristic for the vehicle routing problem with two-dimensional loading constraints”. In: *Networks* 51. Pp. 4–18.

Gillett, B.E. and L.R. Miller (1974). “A heuristic algorithm for the vehicle dispatch problem”. In: *Operations Research* 22. Pp. 340–349.

Glover, F. and G.A. Kochenberger (2003). *Handbook of Metaheuristics.* Boston: Kluwer Academic Publishers.

Glover, F. and M. Laguna (1997). *Tabu Search.* Boston: Kluwer Academic Publishers.

Golden, B. L., E. A. Wasil, J. P. Kelley, and K. M. Chao (1998). “The impact of metaheuristics on solving the vehicle routing problem: algorithms, problem sets, and computational results”. In: *Fleet management and logistics.* Ed. by T. G. Crainic and G. G. Laporte. Kluwer.

Goss, S., S. Aron, J. L. Deneubourg, and J. M. Pasteels (1989). “Self-organized shortcuts in the Argentine ant”. In: *Naturwissenschaften* 76. Pp. 579–581.

Hansen, P. and N. Mladenovic (1997). “Variable neighborhood search”. In: *Computer and Operations Research* 24. Pp. 1097–1100.

— (2001). “Variable neighboorhod search: Principles and applications”. In: *European Journal of Operational Research* 130. Pp. 449–467.

Iori, M., J.J. Salazar-González, and D. Vigo (2006). “An exact approach for the vehicle routing problem with two-dimensional loading constraints”. In: *Transportation Science* 40. Pp. 342–350.

Kindervater, G.A.P. and M.W.P. Savelsbergh (1997). “Vehicle routing: handling edges exchanges windows.”. In: *Local Search in Combinatorial Optimization*. Ed. by E. Aarts and J.K. Lenstra. Chichester: John Wiley & Sons Ltd. Pp. 337–360.

Kirkpatrick, S., C. D. Gelatt, and M. P. Vecchi (1983). “Optimization by Simulated Annealing”. In: *Science* 220. Pp. 671–680.

Laporte, G. and F. Semet (2002). “Classical Heuristics for the Capacitated VRP”. In: *The Vehicle Routing Problem*. Ed. by P. Toth and D. Vigo. Philadelphia: SIAM Monographs on Discrete Mathematics and Applications. Pp. 109–128.

Letchford, A. N. and J. J. Salazar-González (2006). “Projection results for vehicle routing”. In: *Mathematical Programming* 105.2 – 3. Pp. 251–274.

Lin, S. (1965). “Computer solution of traveling salesman problem”. In: *Bell System Technical Journal* 44. Pp. 2245–2269.

Lodi, A., S. Martello, and M. Monaci (2002). “Two-Dimensional Packing Problems: a Survey”. In: *European Journal of Operational Research* 141. Pp. 241–252.

Lodi, A., S. Martello, and D. Vigo (1999). “Heuristic and Metaheuristic Approaches for a Class of Two-Dimensional Bin Packing Problems”. In: *INFORMS Journal on Computing* 11. Pp. 345–357.

Martello, S., M. Monaci, and D. Vigo (2003). “An Exact Approach to the Strip Packing Problem”. In: *INFORMS Journal on Computing* 15.3. Pp. 310–319.

Martello, S., D. Pisinger, and D. Vigo (2000). “The Three-Dimensional Bin Packing Problem”. In: *Operations Research* 48. Pp. 256–267.

Martello, S. and P. Toth (1990). *Knapsack Problems: Algorithms and Computer Implementations*. Chichester: John Wiley & Sons.

Martello, S. and D. Vigo (1998). “Exact Solution of the Two-Dimensional Finite Bin Packing Problem”. In: *Management Science* 44. Pp. 388–399.

Martello, S., D. Pisinger, D. Vigo, E. den Boef, and J. Korst (2007). “Algorithm 864: Algorithms for General and Robot-Packable Variants of the Three-Dimensional Bin

Packing Problem". In: *ACM Transactions on Mathematical Software* 33.1. Article 7, 12 pages. P. 7.

Moscato, P. and C. Cotta (2003). "A Gentle Introduction to Memetic Algorithms". In: *Handbook of Metaheuristics*. Ed. by F. Glover and G.A. Kochenberger. Boston: Kluwer Academic Publishers. Pp. 105–144.

Naddef, D. and G. Rinaldi (2002). "Branch-and-Cut Algorithms for the Capacitated VRP". In: *The Vehicle Routing Problem*. Ed. by P. Toth and D. Vigo. Philadelphia: SIAM Monographs on Discrete Mathematics and Applications. Pp. 53–84.

Oliveira, J.F. and A. Moura (2008). "An integrated approach to the vehicle routing and container loading problems". In: *OR Spectrum*. (to appear). URL: `http://www.springerlink.com/content/35hg2n207q1u0563`.

Or, I. (1976). "Traveling Salesman-Type Combinatorial Problems and their Relation to the Logistics of Regional Blood Banking". PhD thesis. Northwestern Universit, Evanston, IL.

Osman, I.H. (1993). "Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem". In: *Annals of Operations Research* 41. Pp. 421–451.

Parragh, Sophie N., Karl F. Doerner, and Richard F. Hartl (2008*a*). "A survey on pickup and delivery problems Part I: Transportation between customers and depot Volume 58, Number 1 / April 2008". In: *Journal für Betriebswirtschaft* 58.1. Pp. 21–51.

— (2008*b*). "A survey on pickup and delivery problems Part II: Transportation between pickup and delivery locations". In: *Journal für Betriebswirtschaft* 58.2. Pp. 81–117.

Pearl, Judea (1984). *Heuristics: Intelligent Search Strategies for Computer Solving*. Addison-Wesley Publishing Company.

Polacek, Michael, Richard F. Hartl, Karl Doerner, and Marc Reimann (2004). "A Variable Neighborhood Search for the Multi Depot Vehicle Routing Problem with Time Windows". In: *Journal of Heuristics* 10. Pp. 613–627.

Potvin, J. Y. and S. Bengio (1996). "The Vehicle Routing Problem with Time Windows Part II: Genetic Search". In: *INFORMS Journal on Computing* 8. Pp. 165–172.

Prins, C. (2004). "A Simple and Effective Evolutionary Algorithm for the Vehicle Routing Problem". In: *Computers & Operations Research* 31. Pp. 1985–2002.

Reimann, M., K.F. Doerner, and R.F. Hartl (2004). "D-ants: Savings based ants divide and conquer the vehicle routing problem". In: *Computers & Operations Research* 31.4. Pp. 563–591.

Reimann, M., M. Stummer, and K.F. Doerner (2002). "A savings based ant system for the vehicle routing problem". In: *Proceedings of the Genetic and Evolutionary Computation Conference 2002*. Morgan Kaufmann. Pp. 1317–1325.

Robuste, F., C. F. Daganzo, and R. Souleyrette (1990). "Implementing vehicle routing models". In: *Transportation Research B* 24. Pp. 263–286.

Scholl, A., R. Klein, and C. Juergens (1997). "BISON: a fast hybrid procedure for exactly solving the one-dimensional bin packing problem". In: *Computers and Operations Research* 24.7. Pp. 627–645.

Singh, Alok and Ashok K. Gupta (2007). "Two heuristics for the one-dimensional bin-packing problem". In: *OR Spectrum* 29. Pp. 765–781.

Taillard, E.D. (1993). "Parallel iterative search methods for vehicle routing problems". In: *Networks* 23. Pp. 661–673.

Toth, P. and D. Vigo (2002*a*). "Branch-and-bound algorithms for the capacitated VRP". In: *The vehicle routing problem*. Ed. by P. Toth and D. Vigo. Philadelphia: Society for Industrial and Applied Mathematics. Pp. 29–51.

— (2002*b*). *The Vehicle Routing Problem*. Philadelphia: SIAM Monographs on Discrete Mathematics and Applications.

— (2003). "The granular Tabu Search (and its application to the vehicle routing problem)". In: *INFORMS Journal on Computing* 15.4. Pp. 333–346.

Tricoire, F., K. Doerner, R. F. Hartl, and M. Iori (2007). *Variable Neighbourhood Search for the Multi-Pile Vehicle Routing Problem*. Tech. rep. Universita di Bologna - D.E.I.S. - Operations Research. URL: `http://www.or.deis.unibo.it/research_pages/tech_rep/techrep_2007.html`.

Vanderbeck, F. (1999). "Computational study of a column generation algorithm for bin packing and cutting stock problems". In: *Mathematical Programming* 86.3. Pp. 565–594.

Zachariadis, E.E., C.D. Tarantilis, and C.T. Kiranoudis (2007). "A Guided Tabu Search for the Vehicle Routing Problem with Two-Dimensional Loading Constraints". In: *European Journal of Operational Research*. (to appear).

# Abstract in English

Two of the most important problems in distribution logistics concern the loading of the freight into the vehicles, and the successive routing of the vehicles along the road network, with the aim of satisfying the demands of the clients.
In the combinatorial optimization field, these two loading and routing problems have been studied intensively but separately yielding a large number of publications either for routing or packing problems. Only in recent years some attention has been brought to their combined optimization. The obvious advantage is that, by considering the information on the freight to be loaded, one can construct more appropriate routes for the vehicles. The counterpart is that the combinatorial difficulty of the problem increases consistently. One must not forget that both the vehicle routing problem and the bin packing problem are NP hard problems!

This thesis presents three different problems concerning the combination of routing and loading (packing) problems.

- The Multi-Pile Vehicle Routing Problem (MP-VRP) incorporates an interesting loading problem situated between one dimensional and two dimensional bin packing. This problem has been inspired by a real world application of an Austrian wood distributing company.

- The Capacitated Vehicle Routing Problem with Two-Dimensional Loading Constraints (2L-CVRP) augments the classical Capacitated Vehicle Routing Problem by requiring the satisfaction of general two dimensional loading constraints. This means that customers order items represented by rectangles that have to be feasibly placed on the rectangular shaped loading surface of the used vehicles.

- The most general packing problem to be integrated is the Three Dimensional Bin Packing Problem (3DBPP) resulting in the Capacitated Vehicle Routing Problem with Three-Dimensional Loading Constraints (3L-CVRP). Here the order of each customer consists of cuboid shaped items that have to be feasibly placed on the loading space of the vehicle. A feasible placement is influenced by additional constraints that extend the classical 3DBPP.

Concerning the literature solving these problems with exact methods it becomes clear that this is only possible to some very limited extent (e.g.: the MP-VRP can be solved up to 50, the 2L-CVRP can be solved exact up to 30 customers, for the 3L-CVRP no exact approach exists). Nevertheless for real world applications the problem instances are much larger which justifies the use of (meta-)heuristics.
The rank-based Ant System was modified and extended to solve the combined problem by integrating different packing routines. The designed methods outperform the existing techniques for the three different problem classes.
The influence of different loading constraints on the objective value is investigated/is intensively studied to support the decision makers of companies facing similar problems.

# Abstract in German

Zwei der wichtigsten Problemstellungen in der Transportlogistik behandeln einerseits das Verladen von Produkten auf LKWs und andererseits die ressourceneffiziente Belieferung der Kunden auf dem gegebenen Straßennetz.
Bis dato wurden diese zwei Probleme mit Hilfe von kombinatorischer Optimierung getrennt von einander behandelt und es existieren zahlreiche Publikationen zu beiden Themen in den einschlägigen Fachzeitschriften. Erst in den letzten drei Jahren wurde einem integrierten Ansatz, der beide Problemstellungen zu einem Optimierungsproblem vereint betrachtet. Somit werden die Bestellungen einzelner Kunden nicht bloß über ihre Gewichte, sondern auch über ihre Abmessungen definiert. Der klare Vorteil dieses Ansatzes liegt darin, dass die einzelnen LKW Routen auch tatsächlich so gefahren werden können, da die tatsächliche Beladung auch berücksichtigt wurde. Andererseits steigt die kombinatorische Komplexität drastisch, weil das kapazitierte Vehicle Routing Problem (CVRP) mit Bin Packing Problemen (BPP) kombiniert wird und beide Probleme für sich alleine NP schwer sind.
Diese Dissertation präsentiert drei verschiedene Probleme, die sich neben der Frage welches Fahrzeug beliefert welchen Kunden auch der Frage widmet, wie die bestellten Produkte auf den LKW geladen werden können.

- Das Multi-Pile Vehicle Routing Problem (MP-VRP) bindet in das klassische CVRP eine Beladekomponente ein, die zwischen eindimensionalem und zweidimensionalem Bin Packing Problem angesiedelt ist. Die Problemstellungen wurden durch einen österreichischen Holzzulieferer motiviert.

- Beim kapazitierten Vehicle Routing Problem mit zweidimensionalen Beladenebenbedingungen (2L-CVRP) bestellt jeder Kunden rechteckige Objekte, welche auf der rechteckigen Beladefläche des LKWs untergebracht werden müssen.

- Das allgemeinste Beladeproblem stellt das dreidimensionale Bin Packing Problem dar. Hier bestellt jeder Kunde dreidimensionale Objekte, welche auf dem dreidimensionalen Laderaum des LKWs untergebracht werden müssen. Das klassische dreidimensionale Bin Packing Problem wird durch zusätzliche Beladenebenbedingungen erweitert.

Momentan gibt es zu diesen kombinierten Problemen nur wenige Publikationen. Exakte Ansätze gibt es momentan nur zwei, einen für das MP-VRP (hier können Probleme bis zu 50 Kunden gelöst werden) und für das 2L-CVRP (hier können Probleme bis zu 30 Kunden exakt gelöst werden). Für Realweltanwendungen müssen jedoch Heuristiken gefunden werden, welche größere Probleminstanzen lösen können. In dieser Arbeit wird für alle drei Problemstellungen ein Ameisenalgorithmus verwendet und mit bestehenden Lösungsansätzen aus dem Bereich der Tabu-Suche (TS) verglichen. Es wird gezeigt, dass der präsentierte Ameisenansatz für die zur Verfügung stehenden Benchmarkinstanzen die besten Ergebnisse liefert. Darüber hinaus wird der Einfluss verschiedener Beladenebenbedingungen auf die Lösungsgüte untersucht, was eine wichtige Entscheidungsgrundlage für Unternehmen darstellt.

# Günther Füllerer

## Personal Data

day of birth — December $22^{nd}$, 1980 in St. Pölten, Austria

citizenship — Austria

## Education

2000–2005 — **Master Program in International Business Administration**, *University of Vienna*, Vienna, Austria.

2005–2008 — **Doctoral Program in International Business Administration**, *University of Vienna*, Vienna, Austria.

## Master thesis

title — *Packing and Routing with Ant Colony Optimization*

supervisor — Richard F. Hartl

## PhD thesis

title — *Vehicle Routing with Multi-Dimensional Loading Constraints*

supervisor — Richard F. Hartl

## Experience

### Vocational

2005–2008 — **Research Assistant**, *University of Vienna*, Vienna.

## Languages

German — mother tongue

English — advanced, fluent in written and spoken

Russian — fluent

## Computer skills

operating systems — Windows XP, Windows Vista, Linux

programming — C++, Java

*Museumsstraße 16* • *3451 Rust*

✉ *guenther@fuellerer.info* • www.fuellerer.info

web PHP, HTML, Typo3
optimization XPRESS, CPLEX

## Talks & Presentations

EWI 2007 **Ant Colony Optimization for the Two-dimensional Loading Vehicle Routing Problem**, *Fuellerer, M.; Doerner, K.; Hartl, R. and Iori, M.*, Estoril, Portugal.

MIC 2007 **Ant Colony Optimization for the Two-dimensional Loading Vehicle Routing Problem**, *Fuellerer, M.; Doerner, K.; Hartl, R. and Iori, M.*, Canada.

## Publications

**Metaheuristics for Vehicle Routing Problems with Loading Constraints**, *Doerner, K.; Fuellerer, G.; Gronalt, M.; Hartl, R. and Iori, M.*, *Networks*, 2007, 49.
294-307

**Ant Colony Optimization for the Two-dimensional Loading Vehicle Routing Problem**, *Fuellerer, M.; Doerner, K.; Hartl, R. and Iori, M.*, *Computers & Operations Research*, 2008, to appear.

## Teaching

2005–2008 Implementation of Optimization Algorithms in C++

## Interests

research routing, packing, metaheuristics
hobbies skiing, cycling, web design

*Museumsstraße 16 • 3451 Rust*
✉ *guenther@fuellerer.info* • www.fuellerer.info