Research Project

Pickup and Delivery Problems with Cross-Docking and Transfers

Jochem Tanis - s2227118

Evrim Ursavas - first supervisor

Paul Buijs - second supervisor

Leandro C. Coelho - third supervisor

May 24, 2018

Abstract

In this paper the Pickup and Delivery Problem with Cross-Docking and Transfers (PDPCDT) is introduced. The PDPCDT is generalized model that solves problems with one-to-one supplier and customer ratio. The model extends previous transportation problems such as the Pickup and Delivery Problem (PDP), Vehicle Routing Problem with Cross-Docking (VRPCD), PDP with Cross-Docking (PDPCD), and the Two Period VRPCD with Transfers (2P-VRPCDT). The PDPCDT is able to visit suppliers and customers by two types of routing. By solely Pickup and Delivery (PD) route or VRP routes with use of cross-docking. For the last type of route, transfer of packages between cross-docking stations (CDs) can be performed by the use of a shuttle carrier. This enables the possibility of delivering packages by another CD then it was originally picked up. A Variable MIP Neighbourhood Descent is used as accelerated exact solution algorithm to significantly decrease the computational time needed for the search of optimal or near optimal solutions. The PDPCDT with VMND is able to find the optimal and near optimal solutions for solely VRPCD, PDP, and 2P-VRPCDT types of problems. If customers are randomly clustered on a grid, the PDPCDT outperforms solely VRPCD and PDP, since it is able to use both types of routing. The PDPCDT has a costs reduction up-to 30% compared to the 2P-VRPCDT due to the possibility of performing PD routes.

1. Introduction

This paper presents the Pickup and Delivery Problems with Cross-Docking and Transfers (PDPCDT). The PDPCDT consists of route planning between multiple cross-docking stations (CDs) in which goods can be exchanged among carriers. The PDPCDT consists of two classical transportation problems, the Pickup and Delivery Problem (PDP) and the Vehicle Routing Problem with Cross-Docking (VRPCD). The PDP is an extension on the classical Vehicle Routing Problem (VRP), in which the shortest route must be found for a set of geographical locations. Transportation routes between the origin and destination of packages, constructed by the PDPCDT, enable transportation companies to exchange packages between CDs. This ability to exchange induces significant savings in transportation costs. Through the possibility of direct shipment between supplier and customer, further savings can be made due to the exclusion of a stop at a CD. In what follows, the focus will lie on the research done on the PDP and its extensions and on the in VRP that uses the ability of cross-docking (VRPCD).

The first PDP definition is based on the General Pickup and Delivery Problem by Savelsbergh and Sol (1995), in which packages are directly shipped from suppliers to their associated customer by the same carrier. Berbeglia et al. (2007) has identified different categories of PDPs. Distinction is made in planning based on supplier-customer ratio. Many suppliers delivering to many customers (many-to-many), or a single supplier to a single customer (one-to-one), while keeping in mind the possibilities of shipments between these two examples (one-to-many and others). Furthermore, transportation problems can be expressed as a static version, where transportation request is know upfront, or a dynamic case, in which a transportation request may be integrated at any given time (Berbeglia et al., 2010). This paper shall focus on a static version of the transportation problem with an one-to-one ratio.

Next to the PDP is the VRPCD. The VRPCD is an extension on the classical VRP which enables transportation companies to schedule routes with suppliers and customers at the same time. Guastaroba et al. (2016) provided an extensive review on the main contributions in literature on the use of CDs. VRPCD is a widely used warehousing strategy, at which packages are unloaded from incoming carriers and (almost) directly loaded onto outgoing carriers. The CD functions as a storing unit for a short period of time, typically less than 24 hours (Van Belle et al., 2012). In VRPCD, carriers start and end during their routes at the same CD. In VRPCD, one can choose to synchronize incoming and outgoing carriers. Lee et al. (2006) were one of the first that indicated the problem in supplychain and proposed a heuristic based on tabu search to solve a VRPCD at which carriers arrive simultaneously at the cross-docking station. The tabu search and solutions were improved by Liao et al. (2010). In the model proposed by Wen et al. (2009), simultaneously arrival has been relaxed. The model inserts predetermined time windows for each pickup and delivery, satisfying that pickup is done prior to the delivery and that the packages are at the depots before interchanging them among other carriers. Ma et al. (2011) introduces service times for handling packages in the VRPCD with time window constraints for the transportation requests. Heuristics to increase the amount of request were presented by Santos et al. (2011), Dondo and Cerdá (2013), Morais et al. (2014), and Grangier et al. (2017).

Comparable to the VRPCD is the PDP with Transfer (PDPT). The PDPT is an extension of the PDP that relaxes the constraint that packages are delivered by the same carrier, enabling the transfer of packages between carriers. The PDPT has predefined transfer locations where carriers may interchange packages. Mitrović-Minić and Laporte (2006) were one of the first that developed a PDPT with a single transfer point. Additional research into solution techniques for

PDPT with a single transfer node was performed by Cortés et al. (2010), Masson et al. (2013), and Rais et al. (2014), where the last allows transfers to take place at any node. Masson et al. (2014) introduced the transfer of packages with the use of shuttle services between two transfer points. Additionally, Ghilas et al. (2016) modelled transportation of passengers with time windows and synchronization of routes while using scheduled lines for transfers.

Next to the VRPCD a PDP with Cross-Docking (PDPCD) has been studied. This problem was introduced by Petersen and Ropke (2011), who researched the modelling of pickup, delivery, and pickup-and-delivery requests for a Danish transportation company. In their solution, the opportunity exists of performing a PD request by direct delivery through one carrier or by using the ability of transferring the package similar to the VRPCD. They solve the problem with 500-1000 requests while using a large neighborhood search. Whether to use a CD is based on a simple heuristic that makes a decision based on the ratio between the distance of two types of routing. Subsequently, Santos et al. (2013) developed a branch-and-price algorithm for the same problem without time windows and applied it on problem instances containing up to 30 transportation requests. In their model a fixed amount of carriers is used and the model is tested with varying carriers' capacity and packages' handling time of at the CD. The research indicate that an increase of handling time shows a decrease of using the CD. Extensive research was performed by Nikolopoulou et al. (2017), who presented a local-search algorithm for solving the PDPCD with a specified time window. The algorithm is applied on the instances where PDP or VRPCD outperforms the other in transportation costs, and on in the instances at which PDP and VRPCD have comparable transportation costs. In the algorithm, supplier-customer pairs may be relocated from one route to another and are examined on improvement of the solution. Nikolopoulou et al. (2017) indicate that the PDPCD shows significant reduction in transportation costs when the PDP and VRPCD strategies have comparable performance.

There has been some research into the VRPCD with multiple CDs. Research that has been performed in VRP with multiple CD is done in the supplychain management concerning the transfer between suppliers and manufacturers (Dondo et al., 2011). Musa et al. (2010) provide a model that constructs direct or indirect routes between suppliers and customers. The indirect routes go through a CD before ending at the customers. No routes between suppliers and between customers is considered. An ant colony optimization algorithm enables them to solve instances up to 75 requests and 50 CDs. The Multiple Depot VRP (MDVRP) has been researched immensely since 1988 for deliveries of homogeneous packages. This is comparable to the PDPCDT, since pickup and delivery requests need to be assigned to a CD. Montoya-Torres et al. (2015) has provided an literature review on the MDVRP with constraints such as time windows, split deliveries, homogeneous fleet, and periodic deliveries. Adaptive large neighborhood search framework and a tabu search algorithms have been proposed to increase the number of customers for the MDVRP (Pisinger and Ropke, 2007; Escobar et al., 2014). A local improvement algorithm, for problem sizes up to 200 transportation requests, was presented by Dondo and Cerdá (2009). Miao et al. (2012) proposed a hybrid genetic algorithm for VRPCD with multiple CDs. The objective of their model is minimizing all costs including shipments, penalty, and inventory costs. The model is able to solve 22 transportation requests using up-to six CDs. Further research in the VRP with multiple CDs was performed by Mousavi and Tavakkoli-Moghaddam (2013), their model consists of two stages. Firstly, for finding the best locations to open CDs, and secondly, for constructing transportation routes. A hybrid simulated annealing algorithm is proposed that is capable of solving problem sizes with 55 suppliers, 45 customers, and 28 CDs. Additionally Ahkamiraad and Wang (2018) propose a hybrid of the genetic algorithm and particle swarm optimization (HGP) to give good results for instances with 12 suppliers, 38 deliveries and 5 CDs.

Little research is done in the route planning of carriers where transfer between CDs can occur. Research has been done in the supplychain logistics, where carriers perform round-trips between large and smaller distribution centres (Perboli et al., 2011). Inserting transfer of packages, by the use of shuttle service, was introduced by Buijs et al. (2016). In their research, a Dutch company is examined on Generalized PDP (GPDP) with two CDs. Their solution approach excludes the pickup route of packages and the packages are assigned to a CD station based on their geographical location. Buijs et al. (2016) developed a model that provides routes in two ways: firstly, a package can be delivered through a single route from the CD it is assigned to, and secondly, a package is transferred between CDs by using a shuttle service and is subsequently delivered by a delivery route. Similar to the GPDP, Maknoon and Laporte (2017) have developed a model to the VRPCD which enables transfer between CDs, the VRP-Cross-Docking Selection (VRP-CS). In VRP-CS, carriers can perform a pickup route while passing a CD to pickup packages, continue their route and return to their original CD station. Carriers can do the same during a delivery route, by performing deliveries and pickup packages while passing a CD station. An adaptive large neighborhood heuristic was presented as solution technique. The model uses two shifts, the first for pickups and the second for deliveries. Due to two shifts, synchronization of incoming and outgoing carriers is not important. The model is capable of solving 50 transportation requests while using up-to three CD stations and provides optimal and near optimal solutions within eight minutes. Kroep et al. (2017) introduces the Two Period VRPCD and Transfers (2P-VRPCDT). In their solution, pickups are performed by carriers that start and end at the same CD during the first time period. If costs are lowered by transferring packages between CDs, then a shuttle is used that performs a round-trip between the CD stations. In the second period, delivery routes will be executed. Kroep et al. (2017) model time periods similar to the time shifts in the research of Maknoon and Laporte (2017). The model is capable of solving 12 transportation requests optimally with two CDs, resulting in significant savings compared to no cooperation between CD stations. Table 1 provides an overview of the articles discussed and the type of problem they have researched, the table also indicates where this paper extends previous research done in PDP and VRPCD.

The research will be an extension on the the VRPCD of Maknoon and Laporte (2017) and Kroep et al. (2017) through the possibility of direct shipment between supplier-customer pairs. Furthermore, the research extends the PDPCD given by Santos et al. (2013) and Nikolopoulou et al. (2017) by adding CDs and the possibility of transfers between the CDs, yielding a PDPCDT. The transportation process of the PDPCDT for visiting suppliers and customer shall be performed in two types of routes. Solely PDP or VRPCD, at which CDs can interchange packages with the use of a shuttle that performs round-trips. The PD routes shall occur in a single time window with a maximum driving time. The routes for the VRPCD with solely pickup or delivery shall occur in two time windows with halve of the maximum driving time. The purpose of this research is to indicate the possible savings in transportation costs with the use of direct shipment and transfers between CDs. Due to the complexity of the proposed problem, an accelerated exact algorithm is used to decrease the computational time needed to find optimal solutions. In this research the Variable MIP Neighborhood Descent (VMND) is used as solution technique. This hybrid algorithm, proposed by Larrain et al. (2017), switches between two phases, namely an exact algorithm and a local search phase. Larrain et al. (2017) prove that the VMND is successful against solely branch-and-bound with computational time reduction up-to 40% for a cash logistics problem.

The remainder of this paper is structured as follows. The problem formulation of the PDPCDT will be given in Section 2. This is followed by the explanation of the accelerated exact algorithm, presented in Section 3. Section 4 will present the experiments performed and the computational results gained from these experiments. The last section shall consist of an overall conclusion based on the results.

Table 1: Overview of references

Reference	PDP	VRP	CD	Multiple CD	Transfer	Shuttle
Savelsbergh & Sol (1995)	✓					
Lee et al. (2006)		✓	✓			
Liao et al. (2010)		✓	✓			
Wen et al. (2009)		✓	✓			
Ma et al. (2011)		✓	✓			
Santos et al. (2011)		✓	✓			
Dondo & Cerda (2013)		✓	✓			
Morais et al. (2014)		✓	✓			
Grangier et al. (2017)		✓	✓			
Mitrović-Minić & Laporte (2006)	✓				✓	
Cortés et al. (2010)	✓				✓	
Masson et al. (2013)	✓				✓	
Rais et al. (2014)	✓				✓	
Masson et al. (2014)	✓				✓	✓
Ghilas et al. (2016)	✓				✓	✓
Petersen & Ropke (2011)	✓	✓	✓			
Santos et al. (2013)	✓	✓	✓			
Nikolopoulou et al. (2017)	✓	✓	✓			
Musa et al. (2010)		✓		✓		
Pisinger & Ropke (2007)		✓		✓		
Escobar et al. (2014)		✓		✓		
Dondo & Cerda (2009)		✓		✓		
Miao et al. (2012)		✓	✓	✓		
Mouvasi & Tavakkoli-Moghaddam (2013)		✓	✓	✓		
Ahkamiraad & Wang (2018)		✓	✓	✓		
Buijs et al. (2016)		✓		✓	✓	✓
Maknoon & Laporte (2017)		✓	✓	✓	✓	
Kroep et al. (2017)		✓	✓	✓	✓	✓
PDPCDT	✓	✓	✓	✓	✓	✓

2. Problem Formulation

In order to solve the Pickup and Delivery Problem with Cross-docking and Transfer, a mathematical model is presented. As discussed the model uses two types of routes to visit each supplier, associated customer and round-trips between CD stations can occur. The two types of routes are:

- Type 1: A VRPCD routes with possibility of transfer. In this route a carrier starts at its CD and visits a subset of pickup customers. After visiting the customers, the carrier returns to its originating CD. After returning and package handling, the carrier visits a subset of delivery customers and ends again at its CD.

- Type 2: A PD route. In this route a carrier starts at its CD and visits a subset of pickup and delivery pairs. After visiting the last delivery customer, the carrier returns at its originating CD.

A carrier is only able to perform one type of route. For Type 1 the following scenarios are possible: a carrier can visit a supplier-customer pair during its pickup and subsequent delivery route, and a carrier visits a supplier but not the associated customer since packages have been transferred at the CD. The transferred package can be delivered by a carrier of the same CD or another CD since a shuttle carrier has transferred the package. For Type 2 there is only one scenario possible, the carrier visits a subset of suppliers and associated customers at which the supplier is always visited before its customer. Figure 1 illustrates a transportation route given by the model. Each CD performs a Type 1 route for a pickup and delivery. A transfer between the CD stations is used to interchange the packages. Furthermore, both CDs are performing a Type 2 for supplier-customer pair that are near each other.

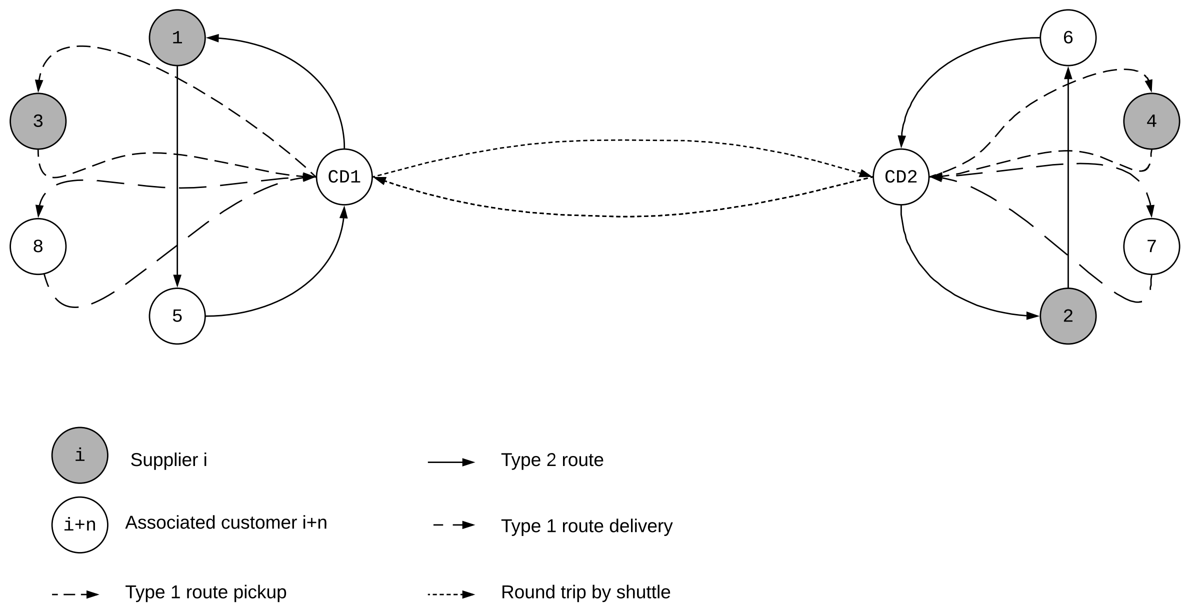

Figure 1: Illustration of the PDPCDT

The mathematical model of the PDPCDT is based on several articles. The use of two types of routes in minimizing transportation costs is inspired by Santos et al. (2013) and the transfer between CDs by Kroep et al. (2017). Constrains developed for the Type 1 are based on the articles from Wen et al. (2009) and Lahyani et al. (2018). The article of Lahyani et al. (2018) present models that enable the in amount of carriers used. Formulations for the Type 2 are based on multi vehicle pickup and delivery problems presented in the survey of Parragh et al. (2008).

In the PDPCDT, a transportation company has m amount of CDs available with K amount of carriers to visit n amount of suppliers and n amount of customers. Each supplier i has an associated customer i+n with a demand q. A CD m consists of two nodes, a node for departure o_1 and a node for arrival o_2. While arriving at suppliers and customers, the carrier has a service time s to pickup or drop-off the package. Each carrier has the same capacity Q, which cannot be exceeded during a transportation route. During the transportation route the carriers cannot exceed the maximum driving time T. The non-negative travelling costs c_{ij} and c_{de} are the costs of a carrier to travel from node i to j and the shuttle carrier's costs to travel between cross-dock d and e. The following sets of nodes can be made. Set V_p consists of all suppliers, V_d of all customers, V_c of all CDs, V_f of all CD arrival and departure nodes, V_{pd} consists of all suppliers and customers, and V of all nodes. For the two types of routes separate sets of nodes are made. V_1 for Type 1 and V_2 for Type 2. All sets of nodes, parameters, and further explanation about the variables are given below.

Sets of nodes

$\mathrm{V}_p = \{1,..,n\}$	Set of all supplier nodes
$\mathrm{V}_d = \{n+1,..,2n\}$	Set of all associated customer nodes
$\mathrm{V}_c = \{1,..,m\}$	Set of all CDs
$\mathrm{V}_f = \{2n+1,..,2n+2m\}$	Set of all CD nodes
$\mathrm{V}_{pd} = \{1,..,2n\}$	Set of all customers
$\mathrm{V} = \{1,..,2n+2m\}$	Set of all nodes
$\mathrm{V}_1 = \{1,..,n,2n+1,..,2n+2m\} \quad and \quad \{n+1,..,2n+2m\}$	Set of all nodes for Type 1
$\mathrm{V}_2 = \{1,..,2n+2m\}$	Set of all nodes for Type 2

Parameters

c_{ij}	Travelling costs between node i and j
c_{de}	Travelling costs between cross-dock d and i
s_i	Service time at node i
q_i	Demand of node i
K	Amount of homogeneous vehicles at the CDs
C	Maximum capacity of the carriers
T	Maximum driving time
M	A large value for capacity
W	A large value for time

Decision variables

x_{ij}^{dk}	1 if edge between i and j is used by carrier k of cross-dock d for Type 1
α_{ij}^{dk}	1 if edge between i and j is used by carrier k of cross-dock d for Type 2
ω_{de}	1 if shuttle is used between cross-dock d and e
y_i^{dk}	1 if node i is visited by Type 1 , cross-dock d and carrier k
z_i^{dk}	1 if node i is visited by Type 2, cross-dock d and carrier k
r_i^{de}	1 if node i is transferred between cross-dock d and e
v_i^{dk}	Departure time of carrier k of cross-dock d at node i by Type 1
w_i^{dk}	Departure time of carrier k of cross-dock d at node i by Type 2
Q_i^{dk}	Truckload of carrier k of cross-dock d at node i

Minimize

$$\sum_{i,j\in V}\sum_{d\in V_d}\sum_{k\in K} c_{ij} x_{ij}^{dk} + \sum_{i,j\in V}\sum_{d\in V_d}\sum_{k\in K} c_{ij} \alpha_{ij}^{dk} + \sum_{d,e\in V_d} c_{de}\omega_{de} \tag{1}$$

Subject to

$$\sum_{d\in V_c}\sum_{k\in K} y_i^{dk} + \sum_{d\in V_c}\sum_{k\in K} z_i^{dk} = 1 \qquad \forall i \in V_{pd} \tag{2}$$

$$\sum_{j\in V}\sum_{k\in K} x_{ij}^{dk} + \sum_{j\in V}\sum_{k\in K} \alpha_{ij}^{dk} = 1 \qquad \forall i \in V_{pd} \tag{3}$$

$$\sum_{j\in V}\sum_{d\in Vc}\sum_{k\in K} x_{ji}^{dk} + \sum_{j\in V}\sum_{k\in K} \alpha_{ji}^{dk} = 1 \qquad \forall i \in V_{pd} \tag{4}$$

$$y_i^{dk} \le y_{o_1}^{dk} \qquad \forall i \in V_{pd}, d \in V_c, k \in K \tag{5}$$

$$z_i^{dk} \le z_{o_1}^{dk} \qquad \forall i \in V_{pd}, d \in V_c, k \in K \tag{6}$$

$$y_{o_1}^{dk} \le \sum_{i,j\in V_1} x_{ij}^{dk} \qquad \forall d \in V_c, k \in K \tag{7}$$

$$z_{o_1}^{dk} \le \sum_{i,j\in V_2} \alpha_{ij}^{dk} \qquad \forall d \in V_c, k \in K \tag{8}$$

$$\sum_{j\in V_1} x_{ij}^{dk} - \sum_{j\in V_1} x_{ji}^{dk} = 0 \qquad \forall i \in V_{pd}, d \in V_c, k \in K \tag{9}$$

$$\sum_{j\in V_1} x_{ij}^{dk} + \sum_{j\in V_1} x_{ji}^{dk} = 2y_i^{dk} \qquad \forall i \in V_{pd}, d \in V_c, k \in K \tag{10}$$

$$\sum_{j\in V_2} \alpha_{ij}^{dk} - \sum_{j\in V_2} \alpha_{ji}^{dk} = 0 \qquad \forall i \in V_{pd}, d \in V_c, k \in K \tag{11}$$

$$\sum_{j\in V_2} \alpha_{ij}^{dk} + \sum_{j\in V_2} \alpha_{ji}^{dk} = 2z_i^{dk} \qquad \forall i \in V_{pd}, d \in V_c, k \in K \tag{12}$$

$$\sum_{i\in V_p} x_{o_1 i}^{dk} = y_{o_1}^{dk} \qquad \forall d \in V_c, k \in K \tag{13}$$

$$\sum_{i\in V_d} x_{o_1 i}^{dk} = \sum_{j\in V_p} x_{o_1 j}^{dk} \qquad \forall d \in V_c, k \in K \tag{14}$$

$$\sum_{j\in V_p} x_{jo_2}^{dk} = \sum_{j\in V_1} x_{jo_2}^{dk} = \sum_{j\in V_p} x_{o_1 j}^{dk} \qquad \forall d\in V_c, k\in K \tag{15}$$

$$\sum_{j\in V_p} \alpha_{o_1 j}^{dk} = \sum_{j\in V_d} \alpha_{jo_2}^{dk} = z_{o_1}^{dk} \qquad \forall d\in V_c, k\in K \tag{16}$$

$$\sum_{j\in V_2} \alpha_{ij}^{dk} - \sum_{j\in V_2} \alpha_{i+n,j}^{dk} = 0 \qquad \forall i\in V_p, d\in V_c, k\in K \tag{17}$$

$$v_{o_1}^{dk} = w_{o_1}^{dk} = 0 \qquad \forall d\in V_c, k\in K \tag{18}$$

$$v_j^{dk} \geq v_i^{dk} + (s_j + c_{ij})x_{ij}^{dk} - W(1 - x_{ij}^{dk}) \qquad \forall i,j\in V_1, i\neq j, d\in V_c, k\in K \tag{19}$$

$$w_j^{dk} \geq w_i^{dk} + (s_j + c_{ij})\alpha_{ij}^{dk} - W(1 - \alpha_{ij}^{dk}) \qquad \forall i,j\in V_2, i\neq j, d\in V_c, k\in K \tag{20}$$

$$\sum_{d\in V_c}\sum_{k\in K} w_{i+n}^{dk} \geq \sum_{d\in V_c}\sum_{k\in K} w_i^{dk} \qquad \forall i\in V_p \tag{21}$$

$$v_i^{dk} + (c_{ij} + s)y_i^{dk} \leq T y_i^{dk} \qquad \forall i,j\in V, i\neq j, d\in V_c, k\in K \tag{22}$$

$$w_i^{dk} + (c_{ij} + s)z_i^{dk} \leq T z_i^{dk} \qquad \forall i,j\in V, i\neq j, d\in V_c, k\in K \tag{23}$$

$$\sum_{j\in V_{pd}}\sum_{i\in V} q_j x_{ij}^{dk} \leq C \qquad \forall d\in V_c, k\in K \tag{24}$$

$$Q_j^{dk} \geq Q_i^{dk} + q_j - M(1 - \alpha_{ij}^{dk}) \qquad \forall i,j\in V_2, d\in V_c, k\in K \tag{25}$$

$$max\{0, q_i\} \leq Q_i^{dk} \leq min\{C, C + q_i\} \qquad \forall i,j\in V_{pd}, d\in V_c, k\in K \tag{26}$$

$$\sum_{k\in K} y_i^{dk} - \sum_{k\in K} y_{i+n}^{ek} - r_{de}^i = 1 \qquad \forall i\in V_p, d,e\in V_c, d\neq e \tag{27}$$

$$\sum_{i\in V_p} r_{de}^i q_i \leq C\omega_{de} \qquad \forall d,e\in V_c, d\neq e \tag{28}$$

$$\omega_{de} = \omega_{ed} \qquad \forall d,e\in V_c, d\neq e \tag{29}$$

$$x_{ij}^{dk}, \alpha_{ij}^{dk} \in \{0,1\} \qquad \forall i,j\in V, d\in V_c, k\in K \tag{30}$$

$$y_i^{dk}, z_i^{dk} \in \{0,1\} \qquad \forall i\in V, d\in V_c \tag{31}$$

$$r_{de}^i \in \{0,1\} \qquad \forall i\in V, d,e\in V_c, d\neq e \tag{32}$$

$$\omega_{de} \in \{0,1\} \qquad \forall d,e\in V_c, d\neq e \tag{33}$$

$$v_i^{dk}, w_i^{dk}, Q_i^{dk} \geq 0 \qquad \forall i\in V, d\in V_c, k\in K \tag{34}$$

The objective of the model (1) is to minimize the traveling costs of the carriers, consisting of the traveling between nodes and the transfer by the shuttle between the CD stations. Constraints (2), (3), and (4) ensure that the each customer is visited exactly once and that each node has a single arrival and departure of a carrier by one of the two types of routes. Constraints (5)-(8) ensure that nodes can only be visited by carriers that are in use. Constraints (9)-(12) maintain the flow conservation of carriers through the nodes. Constraints (13)-(16) ensure that if a carrier is used, it starts and ends at a CD node. Constraint (17) ensures if a supplier is served by a type 2 route carrier, that the same carrier visits the associated delivery. Constraint (18) sets the departure time at the CDs at zero. Constraints (19) and (20) calculate the carrier's departure time at the node by adding the service time and the travelling time between the node and its predecessor to the departure time of its predecessor. In this case a big value W is used, since the constraints only apply if x_{ij}^{dk} or α_{ij}^{dk} is equal to 1. These constraints also function as subtour-elimination constraints. Constraint (21) ensures that the customers is visited after the associated supplier. Constraints (22) and (23) ensure that the arrival time at the CDs does not exceed the maximum driving time and sets departure time to zero if the node is not visited by that carrier. Constraints (24), (25) and (26) compute the capacity of the carrier when it leaves a node and ensures that it does not exceed maximum capacity. Constraint (27) indicates if a package is delivered by the same CD at which it was picked up or that a shuttle carrier is required for the package, since it will be delivered from another CD. Constraint (28) ensures that maximum capacity of the shuttle is not exceeded and indicates if the shuttle is needed. Constraint (29) ensures that the shuttle performs a round-trip. Constraints (30)-(34) define the domain of the decisions variables.

Valid inequalites were proposed by Lahyani et al. (2018) to decrease the computational time needed for achieving a feasible solution.

$$x_{ij}^{dk} = x_{ji}^{dk} = 0 \qquad \forall i \in V_p, j \in V_d, d \in V_c, k \in K \tag{35}$$

$$x_{ie}^{dk} = x_{ei}^{dk} = \alpha_{ie}^{dk} = \alpha_{ei}^{dk} = 0 \qquad \forall i \in V_{pd}, d, e \in V_c, d \neq e, k \in K \tag{36}$$

$$y_e^{dk} = z_e^{dk} = 0 \qquad \forall d, e \in V_c, d \neq e, k \in K \tag{37}$$

$$\alpha_{o_1 j}^{dk} = 0 \qquad \forall j \in V_d, d \in V_c, k \in K \tag{38}$$

$$\alpha_{io_2}^{dk} = \alpha_{jo_1}^{dk} = \alpha_{o_2 j}^{dk} = x_{jo_1}^{dk} = x_{o_2 j}^{dk} = 0 \qquad \forall i \in V_p, j \in V, d \in V_c, k \in K \tag{39}$$

$$x_{ij}^{dk} + x_{ji}^{dk} \leq 1 \qquad \forall i, j \in V, d \in V_c, k \in K \tag{40}$$

$$\alpha_{ij}^{dk} + \alpha_{ji}^{dk} \leq 1 \qquad \forall i, j \in V, d \in V_c, k \in K \tag{41}$$

$$\sum_{i \in V_1} x_{ij}^{dk} = y_j^{dk} \qquad \forall j \in V_{pd}, d \in V_c, k \in K \tag{42}$$

$$\sum_{i \in V_2} \alpha_{ij}^{dk} = y_j^{dk} \qquad \forall j \in V_{pd}, d \in V_c, k \in K \tag{43}$$

$$x_{ii}^{dk} = \alpha_{ii}^{dk} = 0 \qquad \forall i \in V, d \in V_c, k \in K \tag{44}$$

Constraint (35) forbids that Type 1 can visit a customer after a supplier and vice versa. Constraint (36) and constraint (37) forbid that a vehicle housed at a CD can start or end at another CD. Constraint (38) forbids that a customer can be visited after the departure of the CD. Constraint (39) forbids that a supplier can be prior to the arrival at a CD, a carrier can arrive at the arrival CD node, and nodes can be visited after the arrival CD node. Constraints (40) and (41) ensure that there will be no round trips between two nodes. Constraints (42) and (43) ensure that variables for the type of route are equal to each other. Constraint (44) ensures that no route can be made between the same node.

3. Variable MIP Neighborhood Descent Algorithm

To decrease the computational time needed to find the optimal or near optimal solution, a solution technique is proposed. In this research the Variable MIP Neighborhood Descent, proposed by Larrain et al. (2017), is used as solution technique. The MILP of the PDPCDT, presented in the problem formulation, is solved by an exact method. In this research the exact method will be branch-and-bound. If a feasible solution is found or a specified time window has expired, the process is paused and the found solution is given as upper-bound to the local search and the decision variables x_{ij}^{dk}, α_{ij}^{dk} and ω_{de} are stored, since these variables influence the objective value. The local search iterates through multiple neighborhoods that are predetermined upfront while using the solution found as upper-bound for the neighborhoods. The local search goes to the next neighborhood if a better solution is found or a maximum computing time has been met. By creating neighborhoods, the MILP has less difficulty in solving the problem, and therefore, it is quicker in finding a better solution or exploring the entire neighborhood. If the local search does not improve the best solution through all its neighborhoods, the last solution is used for the branch-and-bound and the branching is continued. If the exact method finds a new solution that is not optimal, than this value is used as new best for the local search again. The local search will only go through all the neighborhoods if a new best value has been found. The procedure of the VMND is illustrated in Figure 2.

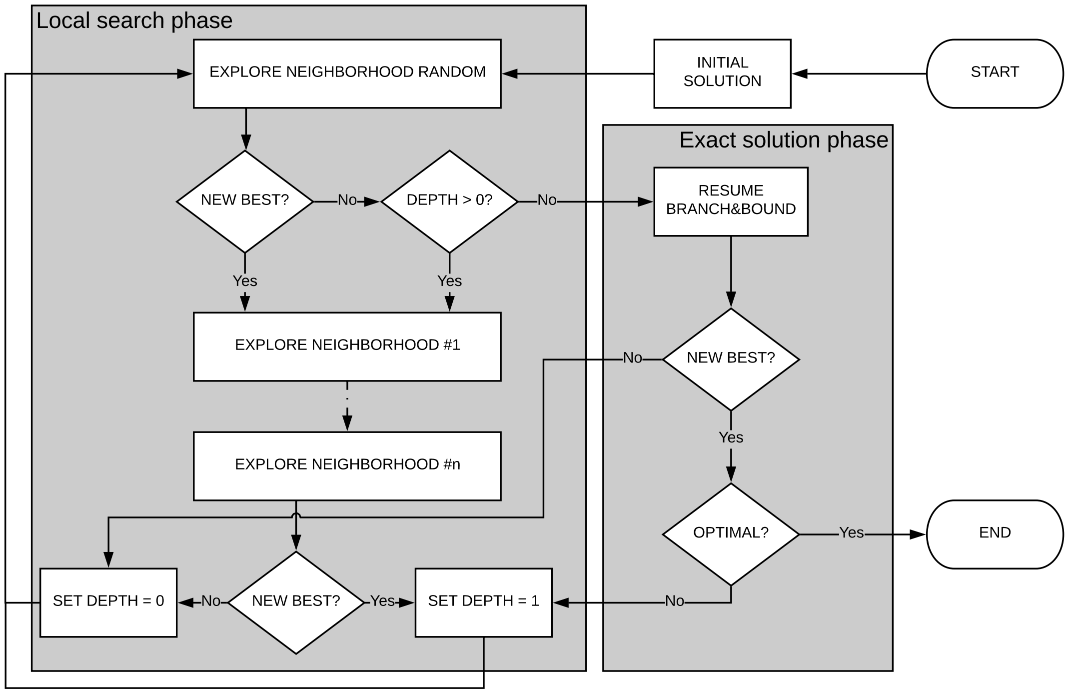

Figure 2: Illustration of the VMND

3.1 Exact solution algorithm

The exact solution algorithm in this research will be the branch-and-bound technique. If the VMND starts searching for a feasible solution, it will use this algorithm on the MILP defined in section 2. If the exact solution method finds a feasible solution, the solution is given to the neighborhoods. If no solution is found within a predefined time window, the exact solution algorithm will be paused. After all neighborhoods have been searched, the exact solution algorithm continues solving with the best found upper-bound until it finds an improved solution or a predefined amount of time is met.

3.2 Local search

The local search phase in the VMND iterates through n amount of neighborhoods. When an initial solution has been found as the first feasible solution, the local search starts with the iteration through the neighborhoods surrounding the best found solution. If a better solution is found in a neighborhood, then this solution will be used for the next neighborhood and will be given as new upper-bound to the branch-and-bound phase. The neighborhoods are made on their similarities. If a neighborhood is used, then this neighborhood shall impose fixed values from the found solution on the decision variables x_{ij}^{dk} and α_{ij}^{dk} that are in- or outside of the neighborhood. Fixing decision variables to a certain value is done by adding constraints in the neighborhood that ensure that these values are met. Within the neighborhood, the local search tries to find a better solution which meets the constraints imposed in the neighborhood, but also the constraints stated in the problem formulation. Within the local search the following neighborhoods are used:

1. Carriers: In this neighborhood carrier k would be set free for decision variables x_{ij}^{dk} and α_{ij}^{dk}. The other carriers will be fixed to the values that were found in the previous best solution.

2. Cross-docks: This neighborhood sets CDs d free and fixes all the other CDs to the values found in the best solution for x_{ij}^{dk} and α_{ij}^{dk}.

3. Periods: During this neighborhood search, the values of the pickup route or the delivery route will be fixed or free. This is only for Type 1 variables, x_{ij}^{dk}.

4. Random: In this neighborhood n random nodes will be set free and the other nodes will be fixed to the values found in the current best solution.

5. Cluster: Based on the distance between the nodes and node i, the best 33% will be selected as a cluster. The cluster will function as a neighborhood and the nodes in this cluster will be set free. All other nodes are fixed.

6. Distance: This neighborhood selects the 33% of the decision variables that have the least travel costs between them. These decision variables are than fixed and all other decision variables are set free.

4. Computational Experiments

Several experiments are performed to study the performance of the VMND against normal branch-and-bound techniques, and to evaluate the performance of the PDPCDT against the models proposed by Kroep et al. (2017) and Nikolopoulou et al. (2017). The model is constructed with JAVA 9.0.4 and is executed on a Intel Xeon E5 2680v3 CPU running at 2.5GHz, while using CPLEX 12.8 as solver. The computational time of the experiments vary between 1 and 2 hours, at each experiment it will be stated what the computational time is.

This chapter is structured as follows. In section 4.1 the benchmark instances will be explained and all parameters will be handled. Section 4.2 will give more explanation about the experiments that are performed and how the results should be interpreted and shall also give an overview of the results with explanation.

4.1 Instances

During the computational experiments two types of instances will be used. The instances generated by Kroep et al. (2017) and the instances generated by Nikolopoulou et al. (2017).

4.1.1 Data Set I

The data set provided by Kroep et al. (2017) consist of 432 different data-sets. The suppliers of the instances have a one-to-one relation with the customers with a demand ranging from [0, 30]. The suppliers have a positive demand and their associated customer has a negative demand. All instances identify the amount of customers, ranging from $V_{pd} \in \{24, 28, 32, 36\}$. The travelling costs between the nodes is given by the Euclidean distance. The maximum route duration (T) per period is equal to 250. Customers are scattered in a $[0, 100]^2$ grid. All instances have two CD stations with three different locations, denoted as: D1, D2, and D3. In these three parameter settings, the distance between the CDs increases. In all of the instances, one of the two CDs is in the western part of the grid and the other CD is in the eastern part of the grid. Furthermore,there is no bound on the amount of homogeneous carriers stationed at the CDs. The customers are scattered in the grid in four different classes. These classes are Random (R), Semi-Clustered (SC), Clustered (C), and Fully Clustered (FC). For the R class, the nodes are uniformly distributed on the grid. In the SC class, all nodes are still uniformly distributed on the y-coordinates of the grid. Regarding to the x-coordinate of the nodes, half of the customers are distributed on the western part of the grid and the other half is distributed on the eastern part of the grid, with some overlap in the middle. In the C class, half of the suppliers is located in the north-west of the grid and their associated customer is located in the south-east. The other half of the suppliers are located in the south-east and their associated customers is located in the north-west. In the FC class, half of the suppliers are located north-west of the western CD station and there associated customers are located south-east of the eastern CDs. For the other half of suppliers and associated customers, it is the other way around. Furthermore, parameters for the capacity of the carriers and shuttle carrier are given in the instances. Table 2 gives an overview of the parameters used in the instances generated by Kroep et al. (2017).

Table 2: Parameter level values used for Kroep et al. (2017) their instances

Parameter Level	Vehicle Capacity (Q)	Depots Locations (D)
1	70	[20,50] & [80,50]
2	100	[20,30] & [80,70]
3	120	[20,10] & [80,90]

4.1.2 Data Set II

The data set obtained from Nikolopoulou et al. (2017) consists of 2915 different instances. All instances contain a unique pickup and delivery node with a one-to-one relation. The instances have a single CD station at which the carriers start and end. The instances are identified by the amount of customers they have, making $V_{pd} \in \{24, 56, 72, 96\}$. Within this research only the 24 and 56 amount of customers will be examined. All nodes are positioned in a $[0, 100]^2$ grid. The demand of the customers are integer numbers ranging between $[1, 30]$. Suppliers are identified as positive demands and their associated customer with a negative demand. The service time at all the nodes is equal to 2. The travel costs between two nodes is given by the Euclidean distances. At all the instances, there is no bound on the amount of homogeneous carriers that can be used. The nodes are distributed throughout the grid in three classes. First is the Random (R), second is the Clustered (C), and last is the Random-Clustered (RC). In the R class, the nodes are randomly scattered on the grid. For the C class, all nodes are distributed in four clusters on the grid. In the last class, RC, half of the nodes are randomly distributed on the grid and the other half is clustered. Furthermore, four other parameters are set in these instances. These parameters are the vehicle capacity Q, maximum route duration T, Depot's location D, and the unloading/reloading time at the CD. The values of these parameters are give in Table 3.

Table 3: Parameter level values used for Nikolopoulou et al. (2017) their instances.

Parameter Level	Vehicle Capacity (Q)	Maximum route duration (T)	Depot's location (D)	Unloading/Reloading time (seconds)
1	70	250	[50,50]	5
2	100	265	[50,60]	7
3	120	280	[50,70]	9

4.2 Computational experiments

Four experiments will be done to examine the performance of the PDPCDT model and the VMND. These experiments show that the PDPCDT is a generalized model, which can be used in different ways. Furthermore, the experiments shall indicate the benefits of using two types of routes with ability of transfer between CD stations against the models proposed by Kroep et al. (2017) and Nikolopoulou et al. (2017).

4.2.1 Comparison of the 2P-VRPCDT with PDPCDT

The first experiment compares the results of the proposed model, without Type 2, to the results generated by the 2P-VRPCD of Kroep et al. (2017) with transfer between the CDs. In this case both models are similar, resulting in similar near optimal or optimal values. Only difference lies in the amount of shuttle carriers available. During this same experiment, the PDPCDT model is executed with and without the VMND solution technique, to illustrate the profits and performance of applying a VMND as solution technique. For this experiment Data Set I will be used and the computational time for 2P-VRPCD is 3 hours, and for the PDPCDT with and without VMND the computational time will be 1 hour.

In Table 4 an overview is presented of the average transportation costs per customer amount and distribution class of Data Set I. The table shows large differences in the average results of the PDPCDT with and without VMND, at which the model with VMND outperforms the model without VMND. The PDPCDT without VMND was able to find 383 feasible solutions and the PDPCDT with VMND was able to find feasible solutions for all instances in one hour. The reason that the PDPCDT with VMND is able to find a feasible solutions for all instances is because the VMND receives a solution that is always possible if no solution is found within a certain time limit. After receiving the first feasible solution the VMND is able to iterate through the neighborhoods.

Table 4: Comparison of 2P-VRPCDT with and without VMND

	2P-VRPCDT				PDPCDT without VMND				PDPCDT with VMND			
Customers	24	28	32	36	24	28	32	36	24	28	32	36
C	727.85	799.63	900.48	975.30	777.73	999.85	1,489.38	2,266.15	729.11	804.74	916.81	991.48
FC	591.26	661.22	768.70	839.48	757.78	1,195.72	1,921.90	2,040.45	635.15	709.19	844.89	971.48
R	883.78	904.67	1,079.48	1,136.22	933.85	1,081.56	1,972.12	2,280.77	893.48	912.19	1,089.93	1,154.22
SC	832.48	944.11	1,042.44	1,097.44	1,602.91	1,263.50	1,700.04	2,410.25	846.22	959.15	1,063.15	1,129.30
Average	758.84	827.41	947.78	1,012.11	1,018.07	1,135.16	1,770.86	2,249.41	775.99	846.31	978.69	1,061.62

Table 4 also gives the performances of the PDPCDT with VMND compared to the averages of the 2P-VRPCDT. For the classes C, R, and SC, the PDPCDT has almost same averages as the

2P-VRPCDT. All averages lie in a range of 2% compared to the 2P-VRPCDT. Difference in the averages are probably caused by the computational time needed to find the optimal solution. The 2P-VRPCDT is a model with a large amount of variables, and therefore, making it a very complex model to find a optimal solution within 1 hour, even with the VMND. A bigger difference in averages can be found in the FC class of distribution. Cause for this difference is the amount of shuttle carriers available in the PDPCDT model. The 2P-VRPCDT model has multiple shuttle carriers available for performing round trips and the PDPCDT only uses a single shuttle carrier. In this distribution class, the model is pushed to use shuttle carriers since the customers pairs lie at the opposites of the grid. Meaning that the capacity limit of the shuttle carrier is met in the PDPCDT model, and therefore, other routing must be used.

4.2.2 Comparison of solely VRPCD and PDP with PDPCDT

In the second experiment, the PDPCDT model will executed solely with Type 1 or Type 2 on Data Set II. In this experiment, extra constraints on the Type 1 will be imposed to make it similar to the VRPCD model proposed in the article of Nikolopoulou et al. (2017). These extra constraints are similar to constraints 9-17 of Wen et al. (2009). The difference lies in the time spend at the CD station for unloading and reloading of packages, which is stated in Table 3. Furthermore, extra intermediate CDs are made to for the constraints to work. The solutions given by the PDPCDT will be compared to the solutions provided by Nikolopoulou et al. (2017) for solely VR-PCD (Type 1) and solely PDP (Type 2). For 24 customers the computational time of the PDPCDT was 1 hour and for 56 customers it is doubled to 2 hours. The increase of customers has a large effect on the amount of variables, and therefore, more time is needed for the VMND to find a near optimal or optimal solution.

Four runs in total where executed with the difference in service time of handling at the CD and type of route. Table 5 indicates the averages of 24 and 56 customers for the VRPCD with different handling time values at the cross-docking stations and the classes of node distribution. There is a slight difference in the averages since Nikolopoulou et al. (2017) do not provide an exact mathematical model in their article. Which might cause the difference in the total travelling costs. Furthermore, they use a solution technique without an exact solver. Resulting in near optimal solutions instead of optimal solutions for some of the instances.

The PDPCDT model with VMND is able to find feasible, similar and cheaper solutions for the instances with an amount of 24 customers. When the amount of customers doubles, the model has more difficulty with finding the near optimal solutions due to the increase of complexity. Therefore, it is chosen to increase the computational time to increase the probability of finding better solutions. The PDPCDT model provides an average travelling costs in a range of 2% compared to the average travelling costs provided by the model of Nikolopoulou et al. (2017).

Similar as to the VRPCD, the PDPCDT model is executed on the Data Set II to provide solely Type 2 solutions. Table 6 provides an overview of the average solutions based on customers and distribution classes. The table shows similarities in the averages and the total averages are in a range of 1.5% compared to each other. Reasons for the differences in averages might be similar to the previous experiment with the VRPCD, caused by the solution technique of Nikolopoulou et al. (2017) and the assumptions in how their mathematical model is constructed. For 56 customers the computational time has been doubled to find better solutions, also in this experiment it can be seen that an increase in customers results in a negative difference in average for travelling costs due to the increase of complexity.

Table 5: Comparison of Type 1 routes

		VRPCD		PDPCDT	
Customers		24	56	24	56
	R	530.84	822.07	527.73	845.93
S1	C	437.25	807.70	436.66	813.61
	RC	443.72	839.88	440.17	885.03
	R	532.19	823.13	524.85	845.09
S2	C	440.87	809.98	435.93	817.74
	RC	444.90	842.47	440.25	866.72
	R	534.36	828.81	526.36	838.90
S3	C	442.21	817.00	435.58	815.27
	RC	446.42	847.23	442.56	855.42
	Average	472.53	826.48	467.79	842.63

Table 6: Comparison of Type 2 routes

	PDP		PDPCDT	
Customers	24	56	24	56
R	379.98	601.52	352.11	554.07
C	356.89	643.72	371.13	690.83
RC	364.43	669.24	364.16	693.16
Average	367.10	638.16	362.46	646.02

4.2.3 Best solutions of Nikolopoulou et al. (2017) against PDPCDT

The third experiment shall consist of comparing the best average results of VRPCD and PDP, based on the solutions provided by Nikolopoulou et al. (2017), with the PDPCDT model. The solutions provided by Nikolopoulou et al. (2017) show that the average values for the different distribution classes and amount of customers are lower for solely PDP. Therefore, the PDPCDT will be compared to the PDP.

Table 7 gives an overview of the average results based on customer amount and distribution class. Compared to the previous experiment where the PDPCDT can only perform Type 2 routes, the current model performs worse in all classes expect for 24 customers distributed Randomly Clustered. Here the PDPCDT performs better then the PDP average. Referring to the previous experiment and Table 6, it can also be seen that the PDPCDT outperforms itself in this class. Meaning that in this distribution class, the PDPCDT uses the possibility of performing Type 1 for delivering packages. Resulting in a decrease of travelling costs. A remark must be made that no transfer was used, since Data Set II only has a single CD.

Table 7: Computational results PDP and PDPCDT

	PDP		PDPCDT	
Customers	24	56	24	56
R	379.98	601.52	398.40	717.51
C	356.89	643.72	385.40	713.75
RC	364.43	669.24	362.15	730.19
Average	367.10	638.16	381.98	720.49

The reason for the differences in the other averages is probably correlated with the complexity of the problem, which is consistent throughout all the experiments. If complexity increases, the model with VMND has more difficulty in finding a near optimal solution or optimal solution. In total the PDPCDT model was able to find 26 significant better solutions than solely PDP, all improved solutions were found with a 24 customers amount.

4.2.4 2P-VRPCDT against PDPCDT

In this experiment the solutions provided by the PDPCDPT model, described in the Problem Formulation with transfer and the PDP routes, will be compared to the 2P-VRPCDT of Kroep et al. (2017). An overview will be given of the transportation costs generated by the 2P-VRPCDT and the PDPCDT. Both models were executed on Data Set I with a maximum computational time of 3 hours for the 2P-VRPCDT and 1 hour for the PDPCDT.

In this experiment the average results per class of customer distribution and amount of customers are compared between the 2P-VRPCDT and PDPCDT. Table 8 gives an overview of the average results for both models. The PDPCDT scores better on almost all classes, but has a higher average on 32 and 36 customers in the FC class. Reason for this is the amount of shuttle carriers available between the depots and the distribution of the customers. The distribution of the customers gives a preference to the use of a shuttle, since the supplier-customer pair are at the opposites of the grid. In the PDPCDT a single shuttle carrier can be used and in the 2P-VRPCDT there are multiple carriers available. An increase of customers in this distribution class means that the capacity limit of the shuttle carrier is met in the PDPCDT model and another way or routing, without transfer, is necessary. This is similar to the results of the experiment in section 4.2.1.

Table 8: Computational results of 2P-VRPCDT and PDPCDT

	2P-VRPCDT				PDPCDT			
Customers	24	28	32	36	24	28	32	36
C	727.85	799.63	900.48	975.30	**546.37**	**633.15**	**812.92**	**911.07**
FC	591.26	661.22	**768.70**	**839.48**	**527.89**	**607.67**	830.37	1,008.42
R	883.78	904.67	1,079.48	1,136.22	**614.96**	**714.00**	**833.59**	**1,026.08**
SC	832.48	944.11	1,042.44	1,097.44	**599.96**	**711.70**	**827.52**	**934.62**
Average	758.84	827.41	947.78	1,012.11	**572.30**	**666.63**	**826.10**	**970.05**

Table 9 provides an overview of the savings in transportation costs between the PDPCDT and the 2P-VRPCDT. The table shows that most savings can be achieved with 24 customers and that an increase in customers shows a decrease in savings. This is because an increase of customers increases the complexity and size of the problem. The PDPCDT uses an accelerated exact algorithm which experiences more difficulty with an increase of customers since it still uses branch-and-bound within the neighborhoods to find better solutions.

Table 9: Savings in percentages

Customers	24	28	32	36
C	24.93%	20.82%	9.72%	6.58%
FC	10.72%	8.10%	-8.02%	-20.12%
R	30.42%	21.08%	22.78%	9.69%
SC	27.93%	24.62%	20.62%	14.84%
Average	23.50%	18.65%	11.27%	2.75%

5. Conclusion

In this paper a Pickup and Delivery Problem with Cross-Docking and Transfers (PDPCDT) has been studied with the aim of minimizing transportation costs for a set of suppliers and customers. The PDPCDT is a generalized model that is able to solve solely PDP, PDPCD, VRPCD, 2P-VRPCDT and is able to combine the last two with PD routes. In order to tackle a complex problem as the PDPCDT, a Variable MIP Neighborhood Descent (VMND) is used as accelerated exact algorithm. The PDPCDT with VMND shows significant improvements in finding solutions compared to the PDPCDT without VMND. Yet the VMND experiences difficulty when complexity and problem sizes increases. Furthermore, the PDPCDT with VMND is able to find similar results, in a range of 2%, for 2P-VRPCDT when the shuttle carrier's capacity is not met. Also the PDPCDT with VMND for solely VRPCD and PDP show similar results, within a range of 2% and 1.5% sequentially. The PDPCDT shows decrease in transportation costs, compared to solely PDP, when suppliers and customers are distributed randomly clustered on a grid. Compared to the 2P-VRPCDT, the PDPCDT is able to induce significant costs savings up-to 30% for 24 customers. These savings decrease when customer amount increases due to the increase of problem's complexity.

Further research in the PDPCDT can be performed in improving the solutions by adjusting the VMND or imposing another solution technique. The current PDPCDT uses a single shuttle carriers. Therefore, adjusting the model with additional shuttle carriers shall indicate better performance on problems where suppliers and customers are fully clustered.

Bibliography

Ahkamiraad, A. and Wang, Y. (2018). Capacitated and multiple cross-docked vehicle routing problem with pickup, delivery, and time windows. *Computers & Industrial Engineering*, 119:76 – 84.

Berbeglia, G., Cordeau, J.-F., Gribkovskaia, I., and Laporte, G. (2007). Static pickup and delivery problems: a classification scheme and survey. *Top*, 15(1):1–31.

Berbeglia, G., Cordeau, J.-F., and Laporte, G. (2010). Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1):8–15.

Buijs, P., Alvarez, J. A. L., Veenstra, M., and Roodbergen, K. J. (2016). Improved collaborative transport planning at dutch logistics service provider fritom. *Interfaces*, 46(2):119–132.

Cortés, C. E., Matamala, M., and Contardo, C. (2010). The pickup and delivery problem with transfers: Formulation and a branch-and-cut solution method. *European Journal of Operational Research*, 200(3):711–724.

Dondo, R. and Cerdá, J. (2013). A sweep-heuristic based formulation for the vehicle routing problem with cross-docking. *Computers & Chemical Engineering*, 48:293–311.

Dondo, R., Méndez, C. A., and Cerdá, J. (2011). The multi-echelon vehicle routing problem with cross docking in supply chain management. *Computers & Chemical Engineering*, 35(12):3002–3024.

Dondo, R. G. and Cerdá, J. (2009). A hybrid local improvement algorithm for large-scale multi-depot vehicle routing problems with time windows. *Computers & Chemical Engineering*, 33(2):513–530.

Escobar, J. W., Linfati, R., Toth, P., and Baldoquin, M. G. (2014). A hybrid granular tabu search algorithm for the multi-depot vehicle routing problem. *Journal of Heuristics*, 20(5):483–509.

Ghilas, V., Demir, E., and Van Woensel, T. (2016). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows and scheduled lines. *Computers & Operations Research*, 72:12–30.

Grangier, P., Gendreau, M., Lehuédé, F., and Rousseau, L.-M. (2017). A matheuristic based on large neighborhood search for the vehicle routing problem with cross-docking. *Computers & Operations Research*, 84:116–126.

Guastaroba, G., Speranza, M. G., and Vigo, D. (2016). Intermediate facilities in freight transportation planning: a survey. *Transportation Science*, 50(3):763–789.

Kroep, F., Buijs, P., Coelho, L. C., and Roodbergen, K. J. (2017). The two-period vehicle routing problem with crossdocking and transfers. Technical Report CIRRELT-2017-53, Montreal, Canada.

Lahyani, R., Coelho, L. C., and Renaud, J. (2018). Alternative formulations and improved bounds for the multi-depot fleet size and mix vehicle routing problem. *OR Spectrum*, 40(1):125–157.

Larrain, H., Coelho, L. C., and Cataldo, A. (2017). A variable mip neighborhood descent algorithm for managing inventory and distribution of cash in automated teller machines. *Computers & Operations Research*, 85:22–31.

Lee, Y. H., Jung, J. W., and Lee, K. M. (2006). Vehicle routing scheduling for cross-docking in the supply chain. *Computers & Industrial Engineering*, 51(2):247–256.

Liao, C.-J., Lin, Y., and Shih, S. C. (2010). Vehicle routing with cross-docking in the supply chain. *Expert Systems with Applications*, 37(10):6868–6873.

Ma, H., Miao, Z., Lim, A., and Rodrigues, B. (2011). Crossdocking distribution networks with setup cost and time window constraint. *Omega*, 39(1):64–72.

Maknoon, Y. and Laporte, G. (2017). Vehicle routing with cross-dock selection. *Computers & Operations Research*, 77:254–266.

Masson, R., Lehuédé, F., and Péton, O. (2013). An adaptive large neighborhood search for the pickup and delivery problem with transfers. *Transportation Science*, 47(3):344–355.

Masson, R., Ropke, S., Lehuédé, F., and Péton, O. (2014). A branch-and-cut-and-price approach for the pickup and delivery problem with shuttle routes. *European Journal of Operational Research*, 236(3):849–862.

Miao, Z., Fu, K., and Yang, F. (2012). A hybrid genetic algorithm for the multiple crossdocks problem. *Mathematical Problems in Engineering*, 2012:1–18.

Mitrović-Minić, S. and Laporte, G. (2006). The pickup and delivery problem with time windows and transshipment. *INFOR: Information Systems and Operational Research*, 44(3):217–227.

Montoya-Torres, J. R., Franco, J. L., Isaza, S. N., Jiménez, H. F., and Herazo-Padilla, N. (2015). A literature review on the vehicle routing problem with multiple depots. *Computers & Industrial Engineering*, 79:115–129.

Morais, V. W., Mateus, G. R., and Noronha, T. F. (2014). Iterated local search heuristics for the vehicle routing problem with cross-docking. *Expert Systems with Applications*, 41(16):7495–7506.

Mousavi, S. M. and Tavakkoli-Moghaddam, R. (2013). A hybrid simulated annealing algorithm for location and routing scheduling problems with cross-docking in the supply chain. *Journal of Manufacturing Systems*, 32(2):335 – 347.

Musa, R., Arnaout, J.-P., and Jung, H. (2010). Ant colony optimization algorithm to solve for the transportation problem of cross-docking network. *Computers & Industrial Engineering*, 59(1):85–92.

Nikolopoulou, A. I., Repoussis, P. P., Tarantilis, C. D., and Zachariadis, E. E. (2017). Moving products between location pairs: cross-docking versus direct-shipping. *European Journal of Operational Research*, 256(3):803–819.

Parragh, S. N., Doerner, K. F., and Hartl, R. F. (2008). A survey on pickup and delivery problems. *Journal für Betriebswirtschaft*, 58(1):21–51.

Perboli, G., Tadei, R., and Vigo, D. (2011). The two-echelon capacitated vehicle routing problem: models and math-based heuristics. *Transportation Science*, 45(3):364–380.

Petersen, H. and Ropke, S. (2011). The pickup and delivery problem with cross-docking opportunity. *Computational Logistics*, 6971:101–113.

Pisinger, D. and Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research*, 34(8):2403–2435.

Rais, A., Alvelos, F., and Carvalho, M. S. (2014). New mixed integer-programming model for the pickup-and-delivery problem with transshipment. *European Journal of Operational Re-*

search, 235(3):530–539.

Santos, F. A., Mateus, G. R., and Da Cunha, A. S. (2011). A novel column generation algorithm for the vehicle routing problem with cross-docking. In *Network Optimization*, volume 6701, pages 412–425. Springer, Berlin.

Santos, F. A., Mateus, G. R., and Da Cunha, A. S. (2013). The pickup and delivery problem with cross-docking. *Computers & Operations Research*, 40(4):1085–1093.

Savelsbergh, M. W. P. and Sol, M. (1995). The general pickup and delivery problem. *Transportation Science*, 29(1):17–29.

Van Belle, J., Valckenaers, P., and Cattrysse, D. (2012). Cross-docking: State of the art. *Omega*, 40(6):827–846.

Wen, M., Larsen, J., Clausen, J., Cordeau, J.-F., and Laporte, G. (2009). Vehicle routing with cross-docking. *Journal of the Operational Research Society*, 60(12):1708–1718.