VEHICLE ROUTING PROBLEM WITH SIMULTANEOUS DELIVERY AND PICKUP AND INTERMEDIARY DELIVERY

by

ERSİN DEMİROK

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

SABANCI UNIVERSITY
Spring 2007

VEHICLE ROUTING PROBLEM WITH SIMULTANEOUS DELIVERY AND PICKUP AND INTERMEDIARY DELIVERY

APPROVED BY:

Assistant Prof. Bülent Çatay ....
(Thesis Supervisor)

Assistant Prof. Tonguç Ünlüyurt ....

Assistant Prof. Dilek Tüzün Aksu...

Assistant Prof. Güvenç Şahin ....

Assistant Prof. Hüsnü Yenigün..

DATE OF APPROVAL: ....

ACKNOWLEDGMENTS

First, I would like to honestly thank Asst. Professor Bülent Çatay for his guidance, motivation, encouragement, and support throughout my research. Also my roommate, classmate and research mate İlkan Sarıgöl deserves my gratitude for his cooperation. I present sincere thanks to my parents and brothers, Mehmet and Erkan, for their endless encouragement, support, and fun when I need. Finally, I thank to my friend Seçil Dönmez for her inestimable support and esprit that has made the accomplishment of this thesis possible.

ABSTRACT

We address a variant of the Vehicle Routing Problem with Backhauls where delivery of the goods picked up from one node to another is allowed along the same vehicle route. The remaining goods in the vehicle are transported back to the depot. Two objectives exist: the primary one is to minimize the total distance traveled; the secondary is to maximize intra-route deliveries. To achieve these goals, we propose a hybrid metaheuristic which consists of an Ant Colony Optimization algorithm for the route construction and a Tabu Search algorithm for the route improvement. To test the performance of our approach, we generate benchmark data based on the well-known problem instances in the literature. Since the variant presented in this paper has not been addressed previously in the literature, only benchmark results with respect to the first objective are available. For this dual objective problem, we attempt to generate a Pareto curve for different levels of the first objective to investigate the trade-off between the two objective functions.

Keywords: Ant colony algorithm, Intermediary delivery, Pick-up and delivery, Simultaneous pick-up and delivery, Tabu search algorithm, Vehicle routing

ÖZET

Araç Rotalama Problemleri, Gezgin Satıcı Probleminin ortaya atılmasından bu yana değerlendirilmektedir. Bu problemler; depo büyüklüğü ve sayısı, araç büyüküğü ve sayısı, zaman bağımlılığı, dağıtım ve geri toplama, toplama çeşitliliği (eşzamanlı gibi), vb açılardan incelenmektedir. Tüm bu problemlerde geçerli olan; "müşteri ziyaret edildiğinde tüm ilgili faaliyetleri gerçekleştirilir" varsayımı, bu çalışmada irdelenmektedir. Çünkü, birçok gerçek vakada; taşıma sadece depodan müşteriye yada müşteriden depoya gerçekleşmemektedir. Bunların yanında müşteriler arası taşıma ihtiyacı da oluşmaktadır. Literatürde, bu tip taşımalar şu şekilde çözülmektedir; önce müşteriden talep alınıp depoya taşınmakta daha sonra depodan alınıp hedef müşteriye götürülmektedir. Çalışmamızda, bu verimsiz yöntem yerine, eşzamanlı Araç Rotalama Problemi için rota içi taşıma yöntemi araştırılmıştır. Makale şu şekilde düzenlenmiştir; ilk bölümde Literatür taraması, ikinci bölümde problem tanımı ve doğrusal model sunumu, üçüncü bölümde çözüm metodu ve algoritması, dördüncü bölümde literatürdeki örneklerden türetilmiş kıyaslama sonuçları ve son bölümde sonuç ve ilerki çalışma konuları sunulmuştur.

Anahtar Kelimeler: Araç Rotalama, Toplama ve Dağıtım, Eşzamanlı Rotalama, Rota içi Taşıma, Arı Kolonisi Algoritması, Tabu Arama Algoritması

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The vehicle routing problem (VRP) deals with delivering goods from a depot to customers in a region using multiple vehicles. Different variants of the problem exist with respect to the depot size and number, vehicle size and number, time-windows, backhauling and linehauling, open routes, etc. Two types of services are of interest in this research: delivery to/pickup from customers and delivery of the picked up goods to customers along the route of the vehicle. These two types of services have been addressed in the literature separately but we are not aware of any research considering both cases simultaneously. In practice, however, these two types of services may be common in cargo carrying, package services, courier deliveries, and freighters. We investigate this problem in the context of VRP with simultaneous delivery and pickup (VRPSDP) structure. This structure handles usual depot-to-customer deliveries and customer-to-depot pickups with customer-to-customer deliveries. Since VRPSDP is a more restricted problem, the solution approach can easily be adapted for other delivery and pickup problems such as VRP with backhauls (VRPB) and VRP with mixed delivery and pickup (VRPMD). The remainder of the paper is organized as follows: Section 2 reviews the related literature. The problem definition and the mathematical formulation are given in Section 3. The proposed solution methodology is discussed in Section 4. The computational study and the results are presented in Section 5. Finally, conclusions and further research directions are in the last section.

CHAPTER 2

LITERATURE REVIEW

In this section, the types of vehicle routing problems are presented and are classified according to their occurrence in the literature. Generally speaking, each new problem is established based on the previous ones. The problems handled are: the open form of the problem, occurrence time and source of the problem, definition the of problem and the most known and recent research about the problem.

The Traveling Salesman Problem (TSP) is the origin of all of the following problems treated since the 1800s. It has first been described by Irish Mathematician William Rowen Hamilton. The TSP is that one salesman travels to customers in a region and sells goods. The objective is to minimize the distance traveled. When the number of salesman is more than one, it's called the Vehicle Routing Problem (VRP). The definition of problem has been made by Clark and Wright (1964) [1]. They investigated the following scenario: a fleet of vehicles deliver goods to customers in a region. In this problem, a single depot is used and the objective is the same as TSP. In addition to the VRP, the capacities of vehicles are investigated. The capacitated vehicle routing problem (CVRP) has been defined in the early sixties and focuses on the delivery of goods to customers in a region where the capacity of vehicles is considered in the single depot model. Toth and Vigo [2] present models, relaxations and exact approaches for the CVRP.

The research on VRP has evolved with the changing needs of the customers. In some cases, customers become the destination. The vehicle routing problem with backhauls (VRPB) has been treated since 1980s as a special case of VRP. A detailed literature review can be first found at Bodin et al. [3]. In this problem, vehicles both deliver and pickup goods from customers in a region with a single depot . The objective is to minimize the distance traveled. Researchers in recent years have paid more attention to the problem. Toth and Vigo [4], Mingozzi et al. [5] propose exact algorithms for the problem by assigning customers into backhaul and linehaul customer subsets. Salhi and Nagy [6] develop cluster insertion based heuristics. Brandão [7] defines a new Tabu search algorithm starting from pseudo-lower bounds for the problem. Gendreau et al. [8] study neighborhood search heuristics. Ropke and Pisinger [9] define a new heuristic called Unified Heuristics and transform the VRPB type problems to a general type to generate better solutions. The VRPB problem is also specified with some additional constraints. The vehicle routing problem with sequence-constrained delivery and pick-up (VRPDP) is investigated since the definition of VRPB problem. In this problem both delivery and pick up of goods from customers in a region is investigated. The vehicles first deliver to the customers rather than pickup from, all customers are visited from a graph with single depot. The constraint of the first deliver then pickup sequence is generated to avoid rearranging the loads on the vehicle. Also the objective is to minimize the distance traveled. Goetschalckx and Blecha [10] define new heuristics, called LHBH, based on a generalized assignment problem that generates the initial solution and routes. Ganesh and Narendran [11] develop a multi phase constructive heuristic that uses the shrink-wrap algorithm and genetic algorithm. A special case of the VRPDP problem is: the vehicle routing problem with precedence constraints relaxed (VRP-PD). The solution is also constructed by the sequence-constrained model but is improved by relaxing the constraint. The problem involves vehicles that both deliver and pickup goods from customers in a region where customers with deliveries and customers with pickups visited without any precedence constraints from a single depot. Since rearranging the loads on the vehicle become less important, new models have been developed for VRPB problems. The vehicle routing problem with mixed delivery and pick-up (VRPMD) is a more general form of VRPB. It was first formulated by Golden et al. [12]. In this formulation, sequence based constraints are never generated. The problem includes the delivery and pick up of goods by vehicles from customers in a region where customers' deliveries and pickups were visited with no precedence in a single depot network while minimizing the distance traveled. Note that the customers are still subgroups of backhaul and linehaul; there is just no precedence between subsets. Wade and Salhi [13] suggest an insertion-type heuristic, where backhaul customers fully mixed with linehaul customers. Then a relaxation in the restriction of the mix of linehaul and backhaul customers is done. In some research, the problem is named as VRPBM: The vehicle routing problem with backhauls and mixed-loads. In addition, VRPBM is also defined as a case where customers can only be backhaul or linehaul.

In a single customer base, both delivery and pickup requests occur and these requests are handled simultaneously. The traveling salesman problem with simultaneous delivery and pickup (TSP-SPD) is involved with one salesman who both delivers and pickups goods simultaneously from customers in a region. In this problem, any customer may be both backhaul and linehaul customer. The problem's objective is to minimize the distance traveled. When the number of salesmen (more commonly, vehicles) is more than one, the problem becomes: VRP with simultaneous delivery and pickup (VRPSDP). The problem is a more general form of the TSPSDP problem. The problem was introduced in 1989 by Min [14] as a case study dealing with a public library distribution system. In this incidence, vehicles deliver and pick up goods simultaneously from customers in a region with a single depot. After the declaration of the problem, researchers concentrated on methodologies with good results. They also disregarded the problem of rearranging the loads on the vehicle. Nagy and Salhi [15] find a solution to the corresponding VRP problem and modify this solution to make it feasible for the VRPSDP and VRPMD by using an integrated heuristics. Dethloff [16], [17] creates the VRPSDP solution by a heuristic construction procedure which suggested a dependence on VRPBM insertion heuristics. Crispim and Brandão [18], Bianchessi and Righini [19] and Montané and Galvão [20] research constructive algorithms. These are local search algorithms and hybrid algorithms based on metaheuristic (like Tabu search) for better solutions of the problem.

There are also different application areas for all previous problems. The multi-depot vehicle routing problem's (MDVRP) first heuristic solutions are suggested by Tillman et al. [21] in the early 1970s. In this problem, vehicles still deliver goods to customers in a region. Differently multiple depots exist. Cordeau et al. [22] suggest heuristic solution to the problem. Also Nagy and Salhi [15] suggest a methodology to transform solution of VRPSDP to solutions in a multiple depot systems. Pisinger and Ropke [23] also convert all problem variants into a rich pickup and delivery model and solve the problem with Unified Heuristics. Furthermore the Time-Window is also an implementation of all given problems. The vehicle routing problem with time windows (VRPTW) is developed after the definition of VRP problems. Time-Window constraints are focused on delivery of goods to customers in a region where customers should be visited in a given time interval in single depot manner. The given time interval means, delivery start time and finish time are specifically defined for each customer. The distance traveled is minimized while considering time windows. Solomon [24] generated benchmark data for VRPTW problems. Rousseau and Gendrau [25] present operators searching large neighborhoods in order to solve the problem. Also, Azi et al. [26] define a method based on an elementary shortest path algorithm by proposing resource constraints. One further problem type is VRPTWB, which means the vehicle routing problem with time windows and backhauls. The problem is also focused on the delivery of goods to customers in a region where customers should be visited in a given time interval in single depot manner. In addition, backhauls are allowed where backhauls occur in a given time interval. Gelinas et al. [27] propose a new branching strategy for branch-and-bound approaches based on column generation. Duhamel et al. [28] cluster customers in two subsets, backhaul and linehaul customers and defines Tabu search algorithm for the solution. Cheung and Hang [29] define the problem as heterogeneous vehicles, multiple trips per vehicle, penalty for early arrivals and develop two optimization based heuristics.

In some real cases, the need of different vehicles reveals new problem types. The site-dependent vehicle routing problem (SDVRP) has firstly been developed by Nag et al. [30] in the late 1980s. The problem involves a fleet of vehicles which delivers goods to customers in a region where customers are associated with vehicle types and there is one depot. Associating a customer with vehicle types mean more than one type of vehicle exists and customer sets are serviced with different types. Cordeau and Laporte [31] suggest a Tabu search solution for the time windows constrained version of the problem. The vehicle routes also differ for several delivery request sets. The open vehicle routing problem (OVRP) was introduced by Sariklis and Powell [32] with the proposition of two cluster first route second heuristics. In this problem, vehicles deliver goods to customers in a region where they are not required to return depot, only routes starts from single depot. Brandão [33] suggests the Tabu search solution for the problem.

There is another problem, The Dial-a-Ride Problem (DARP), which has been studied for more than 30 years. DARP is designing vehicle routes and schedules for n users who specify pickup and delivery requests between origins and destinations, where the objective is to minimize the distance traveled. The first algorithms are presented by

Wilson et al. [34] in the early 1970s. After that new solution algorithms have been presented for both single/multi vehicle and static/dynamic types of DARP problems. Cordeau [35] offers a branch-and-cut algorithm for DARP. Cordeau and Laporte [36] suggest a Tabu search heuristic for the static multi-vehicle version. Bergvinsdottir [37] offers a genetic algorithm for solution. Coslovich and Pesenti [38] propose a two-phase insertion technique of unexpected customers for a dynamic version of problem. Also there are lots of real life applications of DARP, such as the Borndörfer and Grötschel [39] case study about the transportation of handicapped people that cannot use public services. In fact, the pickup and delivery problem (PDP) is a general form of the DARP problem. The first studies of PDP are Wilson et al. [34] in the early 1970s. In PDP, all network transportation requests collected between nodes. All these requests have to be carried out, where the origin and the destination of each of these requests are locations other than the depot. DARP occurs in PDP type of problems but in applications of transportation of people. PDP also involves the transportation of goods. Cordeau et al. [40] presents a good review for types of PDP and adds both heuristic and exact solution methods. The multi vehicle version of the PDP problem in single depot manner is: Pickup and Delivery Vehicle Routing Problem (PDVRP). Most of the DARP algorithms are also generated for multi vehicle version. In some cases of PDVRP, the fleet of vehicles delivers goods from customers to other customers in a region where vehicle constraints are disregarded because load sizes are insignificant (like letters). Cordeau [41] suggests a branch-and-cut algorithm for the problem. Also the pickup and delivery problem with time windows (PDPTW) involves vehicles that delivers goods from customers to other customers in a region and with single depot as PDP does but customers should be visited in a given time interval. The initial efforts were Nanry and Barnes [42] that involved solving PDPTW using the Tabu search heuristic. Most recently; Ropke and Pisinger [43] generate a solution for the problem by a heuristic called “Adaptive Large Neighborhood Search”.

There are some other VRP types but most common types are presented. VRPSDP with intermediary delivery (VRPSDP-ID) is more closely related with VRPSDP and PDP. It involves delivery and pickup in the more restricted simultaneous environment like in VRPSDP. In addition, some requests may occur between customers like in PDP. The problem definition and the mathematical model are presented in the next chapter.

CHAPTER 3

PROBLEM DEFINITION AND MATHEMATICAL MODEL

3.1. Problem Definition

Since the Vehicle Routing Problem with Simultaneous Delivery and Pickup and Intermediary Delivery (VRPSDP-ID) is very similar to VRPSDP, we first describe VRPSDP. Dethloff [16] defines the VRPSDP as a group of customers that are serviced by a number of vehicles which has limited capacities and which are generally identical. The customers have requests for pickup of goods (linehaul) from a central depot in where the vehicles initially are and have requests for delivery of goods (backhaul) to the same central depot. Vehicles start from the depot and return to the depot at the end of their service. Note that all deliveries originate from the depot and all the pickups are sent to the depot. The VRPSDP is to determine the vehicle routes while satisfying customer requests with the minimum total distance delivered. Since in many practical applications a customer may have both delivery and pickup requests and differentiating the types of requests are not cost effective, they have to be handled together. In addition, these customers may not accept to be serviced separately for handling reasons. Therefore, simultaneous delivery and pick-up can be done and in a way that each customer is serviced with a single stop.

Our problem is similar with VRPSDP in many aspects: there exists a central depot at which the vehicle routes start from and end at; the vehicles are capacitated and identical; every customer is visited for their requests only once; and requests are treated simultaneously. However; there is one critical exception: in a route, all deliveries do **not** originate from the depot and all pickups are **not** sent to the depot. Some goods may be transported between customers directly. If they are on the same route and pickup of a good occurs before the delivery of the same good than the delivery of that good is performed. Otherwise, all the undelivered pickups are collected in the depot. We refer to these undelivered pickups as a new term called *Returns*.

Similar to VRPSDP-ID, in PDP requests of delivery and pickup between customers exist. Cordeau, Laporte and Ropke [40] present the PDP as follows: an effort to design the least cost vehicle routes for requests that occur between customer pairs. The routes start from and end at a central depot and there are precedence constraints between customer pairs. The model is applied to the transportation of goods and transportation of people such as the Dial-a-Ride problem (DARP). There are also time limitations for PDP such as the total time of a vehicle route, the earliest and latest service times for each customer and the maximum amount of time goods can spend in the vehicle. Also, the delivery and pickup requests of customer pairs should be located on the same vehicle route while satisfying vehicle capacity constraints and time limitations.

Similarly in VRPSDP-ID; There exist delivery and pickup requests of customer pairs. The aim is to design the least costly vehicle routes which start and end at a central depot while satisfying **some** of the customer pair requests, but not necessarily all. There are also requests between the depot and the customers to be fully satisfied. The model can be applied to both the transportation of goods and transportation of people. In the transportation of people case however; some requests may be satisfied causing some people to be transported to the depot instead of destination node referred to as *Returns*. In VRPSDP-ID we do not consider any type of these time limitations and leave this for investigation as a future research.

VRPSDP-ID deals with the delivery of the goods from depot to customers and pickup and return of the goods from customers to depot where delivery of the goods picked up from a customer to another is allowed along the same vehicle route. The remaining goods in the vehicle are transported back to the depot. The objective is to minimize the total distance traveled while considering total number of vehicles assigned. The secondary objective is to maximize intra-route deliveries. The intra-route deliveries are maximized while minimizing the remaining goods in the vehicle which are transported back to the depot. The Returns are the goods with destinations to other customers that have not been delivered along the route of the vehicle and transported back to the depot.

We make the following assumptions when addressing the problem:

- There is only one type of commodity.
- There is a single, uncapacitated depot.
- Distances are linear and symmetric.
- The vehicles are identical.
- Demand is deterministic and stationary (single planning period).
- Each vehicle route starts and ends at the depot and there is no route length restriction.
- Every customer is visited only once.
- Each customer has both a delivery and a pickup request to be satisfied simultaneously.

The objective is to minimize the total distance traveled as well as to minimize the total amount of returns to the depot. The minimization of the number of vehicles is not explicitly addressed in this research. However, a solution with a fewer number of vehicles is accepted as a better solution even if the total distance is longer.

The terms distance and returns are explained using the network in Figure 3.1:

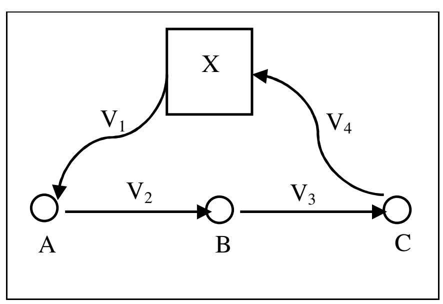

Figure 3.1 Term Explanation Network

Assume that there are 3 customers (A, B, and C), a single depot(X), and a single vehicle. The vehicle departs from the depot X, visits customers A, B, and C and then returns to the depot. The vehicle route consist of the links V_1, V_2, V_3, and V_4. There are delivery and pickup requests between customers and between the depot and customers, as depicted in the supply matrix in Table 3.1.

	X	A	B	C
X	0	S_{XA}	S_{XB}	S_{XC}
A	S_{AX}	0	S_{AB}	S_{AC}
B	S_{BX}	S_{BA}	0	S_{BC}
C	S_{CX}	S_{CA}	S_{CB}	0

Table 3.1 Supply Matrix (S) (S_{ij} : Amount of goods transported from origin i to destination j)

Also the distances are given with distance matrix in Table 3.2.

	X	A	B	C
X	0	D_{XA}	D_{XB}	D_{XC}
A	D_{AX}	0	D_{AB}	D_{AC}
B	D_{BX}	D_{BA}	0	D_{BC}
C	D_{CX}	D_{CA}	D_{CB}	0

Table 3.2 Distance Matrix (D) (D_{ij} : Distance between origin i and destination j)

From the matrix the length for arc V_1 is D_{XA}, for arc V_2 is D_{AB}, for arc V_3 is D_{BC} and for arc V_4 is D_{CX}. For the route of the vehicle:

At Depot X: Loads S_{XA}, S_{XB}, S_{XC} deliveries from X;

At A: Unloads S_{XA} delivery to A, Loads S_{AX} pickup and S_{AB}, S_{AC} deliveries

At B: Unloads S_{XB}, S_{AB} delivery to B, Loads S_{BX} pickup and S_{BA}, S_{BC} deliveries

At C: Unloads S_{XC}, S_{AC}, S_{BC} delivery to C, Loads S_{CX} pickup and S_{CB}, S_{CA} deliveries

At Depot X: Unloads S_{AX}, S_{BX}, S_{CX} delivery to X, <u>Returns</u> are S_{BA}, S_{CA}, S_{CB} deliveries

Returns are the goods that could not be delivered on a vehicle route and carried back to depot at the end of route.

Total Distance Traveled: $D_{XA} + D_{AB} + D_{BC} + D_{CX}$ (One of the objective)

Also;

Total Returns: $S_{BA} + S_{CA} + S_{CB}$ (Another objective)

The following is a numerical example. We assume the same network structure with a single depot and three customers. The supply table and distance table are given as follows:

S	X	A	B	C
X	0	2	2	2
A	1	0	1	0
B	1	1	0	0
C	1	2	0	0

D	X	A	B	C
X	0	1	1	1
A	1	0	1	2
B	1	1	0	1
C	1	2	2	0

Table 3.3(a) and 3.3(b) Supply (S) and Distance (D) Matrix of Example (in units)

The fleet is homogenous with a vehicle capacity of 6 units. Then, the VRPSDP solution will be as follows as illustrated in figure 3.2:

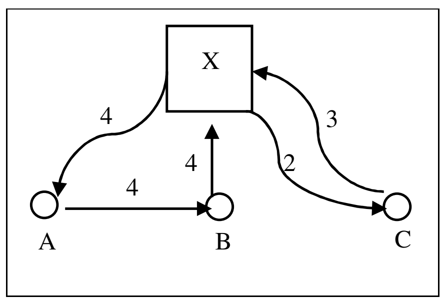

Figure 3.2 VRPSDP Solution for the Example

The first vehicle leaves from the depot, visits A and B respectively, and then returns to the depot. The second vehicle leaves from the depot, visits C, and returns to the depot. The loads on each arc are given in Figure 3.2.

The total distance is: 3 (for the first vehicle) + 2 (for the second vehicle) = 5

The total returns is: 2 (for the first vehicle) + 2 (for the second vehicle) = 4 (there is no intra route delivery)

The no of vehicles assigned: 2

The corresponding PDP solution will be as follows:

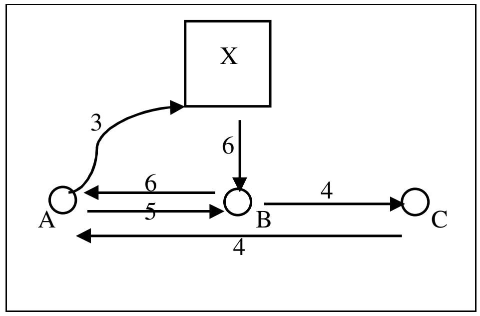

Figure 3.3 PDP Solution for the Example

The first vehicle leaves from depot and visits the customers in the following order: B, A, B, C, and A then returns to the depot. There is no need for a second vehicle. The loads on each arc are given;

The total distance is: 7

The total returns is: 0 (some customers visited more than once)

The no of vehicle assigned: 1

The VRPSDP-ID solution will be as follows:

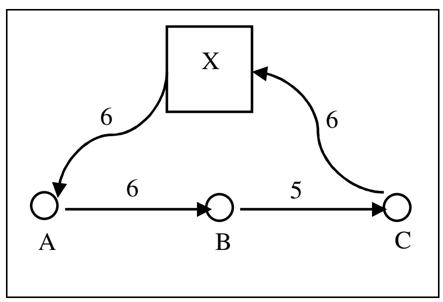

Figure 3.4 VRPSDP-ID Solution for the Example

The first vehicle leaves from depot, visits the customers in the order: A, B, C then returns depot. There is no need for a second vehicle. The loads on each arc are given;

The total distance is: 4

The total returns is: 3 (there is intra route delivery and each customer visited only once)

The no of vehicle assigned: 1

Table 3.4 summarizes the results of the example:

Solutions	Distance	Returns	Number of Vehicle
VRPSDP	5	4	2
PDP	7	0	1
VRPSDP-ID	4	3	1

Table 3.4 Summary of Solutions

We observe that PDP solution has no returns but that solution violates the rule of visiting each customer once. In addition PDP solution has greater distance values. Since VRPSDP has lower distance solution than PDP, its returns value is more than VRPSDP-ID. Moreover VRPSDP-ID has the best distance value. Note that all the example solutions are found by trials.

3.2. The Mathematical Model

Following the mathematical model of VRPSDP [16], VRPSDP-ID can be modeled with some additional constraints as follows:

Notation

Sets

J : set of all customer locations

J_0 : set of all nodes, i.e. customer locations and depot, $J_0 = J_0 \cup \{0\}$

V : set of all vehicles

Parameters

C : vehicle capacity

C_{ij} : distance from node $i \in J_0$ to $j \in J_0, i \neq j; C_{ii} = M\,(i \in J), C_{00} = 0$

S_{ij} : delivery amount between node i to node j $\quad i \in J_0, j \in J_0, i \neq j; S_{ii} = 0$

n : number of nodes, i.e. $n = |J_0|$

M : large number, e.g. $M = \max\left\{ \sum_{j \in J_0} S_{ij}, \sum_{i \in J_0} \sum_{j \in J_0, j \neq i} C_{ij} \right\}$

Decision Variables

$l_v^{'}$: load of vehicle $v \in V$ when leaving depot; can be eliminated from the model

l_j : load of the vehicle after servicing customer $j \in J$

π_j : variable used to prohibit subtours; can be interpreted as position of node $j \in J$ in the route

x_{ijv} : binary variable indicating whether vehicle $v \in V$ travels directly from node $i \in J_0$ to node $j \in J_0$ ($x_{ijv} = 1$) or not ($x_{ijv} = 0$)

y_{ijv} : binary variable indicating whether vehicle $v \in V$ travels (not directly required) from node $i \in J_0$ to node $j \in J_0$ ($y_{ijv} = 1$) or not ($y_{ijv} = 0$)

Mathematical Model

$$\text{Minimize } z = \sum_{i \in J_0} \sum_{j \in J_0} \sum_{v \in V} C_{ij} x_{ijv} \tag{1}$$

Objective 1: (Minimize total travel distance)

$$\text{Minimize } z = \sum_{j \in J} \sum_{v \in V} x_{j0v} l_j - \sum_{j \in J} S_{j0} \tag{2}$$

Objective 2: (Minimize total returns)

subject to

$$\sum_{i \in j_0} \sum_{v \in V} x_{ijv} = 1 \qquad j \in J \tag{3}$$

(Service all customers exactly once)

$$\sum_{i \in J_0} x_{isv} = \sum_{j \in J_0} x_{sjv} \qquad s \in J, v \in V \tag{4}$$

(Arrive at and leave each customer with the same vehicle)

$$l_v^{'} = \sum_{i \in j_0} \sum_{j \in J} S_{ij} x_{ijv} \qquad v \in V \tag{5}$$

(Initial vehicle loads)

$$l_j \geq l_v^{'} - S_{0j} + \sum_{i \in J_0} S_{ij} - M(1 - x_{0jv}) \qquad j \in J, v \in V \tag{6}$$

(Vehicle loads after first customer)

$$l_j \geq l_i - \sum_{k \in J_0} S_{kj} y_{kjv} + \sum_{k \in J_0} S_{jk} - M(1 - \sum_{v \in V} x_{ijv}) \qquad i \in J, j \in J, j \neq i \qquad (7)$$

(Vehicle loads 'en route')

$$l_v^{'} \leq C \qquad v \in V \qquad (8)$$

$$l_j \leq C \qquad j \in J \qquad (9)$$

(Vehicle capacity of initial loads, after first customer and 'en route')

$$\pi_j \geq \pi_i + 1 - n(1 - \sum_{v \in V} x_{ijv}) \qquad i \in J, j \in J, j \neq i \qquad (10)$$

(Subtour breaking constraints)

$$\begin{aligned} & y_{ijv} \geq x_{ijv} \\ & 2y_{ijv} \geq x_{ikv} + x_{kjv} \\ & 3y_{ijv} \geq x_{ikv} + x_{klv} + x_{ljv} \quad i \in J_0, j \in J_0, k \in J_0, l \in J_0, v \in V \qquad (11) \\ & \ldots \\ & ny_{ijv} \geq x_{ikv} + \ldots + x_{ljv} \end{aligned}$$

(Indirect vehicle travels)

$$\pi_j \geq 0 \quad j \in J \qquad (12)$$

$$x_{ijv} \in \{0,1\} \qquad i \in J_0, j \in J_0, v \in V \qquad (13)$$

$$y_{ijv} \in \{0,1\} \qquad i \in J_0, j \in J_0, v \in V \qquad (14)$$

First objective (1) minimizes the total distance travelled by all vehicles. The second objective (2) minimizes total returns to the depot. Returns are calculated by collecting final loads of each vehicle minus depot deliveries (pickups from customers to depot). Constraint (3) ensures that each customer is serviced exactly once. Constraints (4) assure that a vehicle arriving to a customer also leaves the same customer. Constraints (5), (6) and (7) define vehicle loads after leaving the depot, after leaving the first customer and en route respectively. Constraints (8) and (9) are the vehicle capacity constraints. Constraints (10) are subtour elimination constraints. Constraints (11) define indirect travels, i.e. if two customers are on the same vehicle route but not necessarily consecutive then there is an indirect travel. Finally (12), (13), and (14) are the bounds for the decision variables.

Since VRPSDP is NP-Hard and it is a special case of VRPSDP-ID with zero intermediary deliveries, VRPSDP-ID is also NP-Hard. Furthermore, constraint set (11) appears as a significant complicating factor in the problem with a large number of additional constraints. For n customers, the number of constraints of this type for each vehicle will be as follows:

$$n(n-1)+n(n-1)(n-2)+n(n-1)(n-2)(n-3)+\ldots+n(n-1)\ldots(n-(n-1))$$

Since the problem is intractable for even moderately large instance we propose a metaheuristic solution approach in the next chapter.

CHAPTER 4

SOLUTION METHODOLOGY

The aim of this study is to develop a solution methodology to generate a Pareto curve for the dual objective problem. We apply a hybrid metaheuristic approach which consists of an Ant Colony Optimization (ACO) algorithm based construction algorithm and a Tabu search based improvement algorithm. The dual objective problem is solved by tuning the related parameters accordingly in an attempt to obtain Pareto efficient solutions. For each solution case both objective values are recorded separately and dominated solutions are discarded when creating the Pareto curve.

4.1. Construction: Ant Colony Optimization Algorithm

ACO is used in difficult optimization problems for generating approximate solutions [43]. Colorni et al. [44] introduced the algorithm to solve the TSP. ACO has been applied to different kind of combinatorial optimization problems since then. Stützle and Dorigo [45] use it in quadratic assignment problem, Colorni et al. [46] in scheduling problems, Gambardella and Dorigo [47] in sequential ordering problem, and Bullnheimer et al. ([48], [49]); Gambardella et al. [50]; Doerner et al. [51]; Reimann et al. ([52], [53]) in VRP.

Mimicking the behavior of real ant colonies while they were searching for food is the origin of the ACO. Real ants leave pheromone, a kind of scent, while walking on their path. Also they use the density of pheromone while selecting the path which was secreted from previous ants. In any path, the density of pheromone depends on the quality of food source and length of path. As the path leads to a high quality food source, more ants subsequently use the path and the density of pheromone increases, which also increases the selection chance of that path for subsequent ants. In optimization problems, artificial ants are created and the problem is transformed into a weighted graph. The solution is generated as a stochastic construction process by the moves of the artificial ants on weighted graph [43].

Colorni et al. [44] developed the first ACO and named it the Ant System (AS). The ants finish their tours; then, the pheromone update is performed in AS. Colorni et al. use AS to solve the TSP problems but the results were worse than the existing procedures. An elitist strategy (EAS) is proposed by Dorigo et al. [54] by reinforcing the best ant tour with more weight than the others. Dorigo and Gambardella [55] propose that the Ant Colony System (ACS) generates better solutions with AS using the magnitude of the intensification instead of diversification. Magnitude is accomplished by a strong elitist strategy in updating and a pseudo-random proportional rule in selecting the next node. Also, in ACS a pheromone update is used while constructing tours and during local search.

Stützle and Hoos [56] suggest Max-Min Ant System (MMAS) by letting the best-so-far ant or the iteration-best ant in pheromone updates. To avoid getting trapped in inaction, they define maximum and minimum bounds for pheromone levels. If no action or no improvement is achieved after subsequent iterations, pheromone values are refreshed. Bullnheimer et al. [49] rank each ant solution depending on their quality and update the pheromone with all ants weighted with their rank. This method is called the Rank-based Ant System (ASrank). In ASrank, the best-so-far ant is authorized to update the pheromone with the largest weight.

Dorigo and Stützle [57] comprehensively review a variety of Ant Colony Algorithm metaheuristics. In our approach, we adopt an EAS strategy.

4.1.1. Initialization

Pheromone value of each arc is initialized as τ_0. This amount is defined in the literature as $\tau_0 = 1/nL_0$ where n is the number of customers and L_0 is the length of an initial feasible route. We also adopt this initialization of pheromone levels in the construction phase of our hybrid metaheuristic: the Ant Colony Algorithm. There is one difference: since we have a second objective function of total returns, the initial total returns value is integrated to the initial amount of pheromone as:

$$\tau_0 = 1/nL_0R_0 \tag{1}$$

where R_0 is the total amount of returns of an initial feasible route where the initial route is constructed using a nearest-neighbor heuristic (NNH) starts at the depot and then selects the not yet visited closest feasible customer as the next customer to be visited regardless of whether it is a linehaul or backhaul customer. A customer is infeasible if s/he violates the vehicle capacity. If no feasible customer is available, then the route is terminated at the depot and a new route is initiated.

In the literature, there is single visibility definition, named L_0, which is a value between a pair of customers calculated by the inverse of their distance. Here the distance value is calculated from the NNH and serves as the first objective (total distance travelled). However, in addition to the total distance travelled, the total returns value is another objective. The returns are goods undelivered at the end of a route. These uncompleted deliveries occur because there is no indirect vehicle travels between pickup customer and delivery customer. This situation may happen if two customers are not on the same route or a delivery customer occurs before the pickup customer on the same route. The amount of load between these customers is returned back to depot. So the loads between customers serve the second objective. Thus a new additional visibility value between a pair of customers is equal to the supply among them:

$$\eta_{ij}^{d} = 1/d_{ij} \qquad \text{for the distance} \qquad (2)$$

$$\eta_{ij}^{r} = \min^{+}(S) + S_{ij} \qquad \text{for the load} \qquad (3)$$

where d_{ij} denotes the distance between customers i and j as usual in ant colony algorithms. Here η_{ij}^{d} denotes visibility for distance and η_{ij}^{r} denotes visibility for returns. Since a high value of η_{ij}^{d} indicates that visiting customer j after customer i is a desired choice, the tour length is expected to be shorter if the probability of moving from customer i to customer j increases with η_{ij}^{d}. Also η_{ij}^{r} is another visibility value that depends on the amount of supply between customers. S_{ij} denotes the supply between customers i and j. Some of the S_{ij} values may be 0. To avoid trapping in 0 in visibility calculations; $\min^{+}(S)$ terms added, that means minimum positive load among all customers and the depot. A high value of η_{ij}^{r} indicates that visiting customer j after customer i is a desired choice; the amount of returns for that tour is expected to be less if the probability of moving from customer i to customer j increases with η_{ij}^{r}.

In our approach, the two visibilities are defined for creating various instances for various levels of the first objective. Each instance differs by various weights of visibilities. The first visibility value attempts to shorten the total distance travelled by assigning higher probabilities to closer customer pairs. On the other hand, the second visibility value attempts to decrease the total returns by assigning higher probabilities to customer pairs which have larger supply between them. If a customer pair has a greater supply then the route passes through both customers and the returns at the end of the route decrease. In this manner, various priorities for these visibilities create various instances for the solution set which in turn creates the opportunity to investigate the trade-off between the two objective functions.

4.1.2. Route Construction

ACS is used in the route construction process with the pseudo-random proportionality rule. Feasible customers set N_i^k defined for each ant are positioned at each customer. Each ant consequently constructs its own route by selecting a customer from its set. N_i^k is listed by taking not yet visited customers minus the customers which violate the vehicle capacity, for each ant k at each customer i. The selection process of customers is defined with an attractiveness value, defined by the combination of pheromone trails and the visibility:

$$\varphi_{ij} = [\tau_{ij}]^{\alpha}[\eta_{ij}^{\ d}]^{\beta}[\eta_{ij}^{\ r}]^{\gamma} \tag{4}$$

where α, β and γ are parameters for the pheromone τ_{ij}, visibility of first objective η_{ij}^{d} and visibility of second objective η_{ij}^{r} to control the weights of these. The different values of β and γ create the instances for a solution set. An ant k located at customer i randomly selects a customer using the following rule:

$$j^k = \begin{cases} \arg\max\limits_{j \in N_i^k} \varphi_{ij} & , \quad \text{if } q \le q_0 \\ J^k & , \quad \text{otherwise} \end{cases} \tag{5}$$

where $q_0(0 \le q_0 \le 1)$ is a parameter to control intensification versus diversification, q is a random variable drawn from a uniform distribution $U(0,1)$ and J^k is selected according to the following probability distribution:

$$p_{ij}^k = \begin{cases} \dfrac{\varphi_{ij}}{\sum\limits_{l \in N_i^k} \varphi_{il}} & , \quad \text{if } j \in N_i^k \\ 0 & , \quad \text{otherwise} \end{cases} \tag{6}$$

4.1.3. Pheromone Update

In AS, which is the first Ant Colony Algorithm proposed for TSP, pheromone update is done by first reducing all pheromone values by a constant rate and then pheromones are increased by the ants' routes which are used in the previous solution. Reducing the process is called evaporation, and increasing is called reinforcement. The evaporation prevents from trapping the local optimums by reducing the previous pheromone values which are created repeatedly by selection of similar ant routes. The reinforcement provides selecting ant routes whose solutions increase the probability of selecting the short tours. The evaporation process is implemented by:

$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij} \tag{7}$$

where ρ $(0 < \rho \le 1)$ is the evaporation parameter. The reinforcement process is implemented by:

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^{K} \Delta\tau_{ij}^k \tag{8}$$

where L^k is the length and R^k is the returns of the best ant's tour and $\Delta\tau_{ij}^k = 1/L^k R^k$. Note, K indicates the number of best-ant used for pheromone reinforcement.

In the elitist strategy, which is used in EAS, the best tour that was achieved since the initiation of the algorithm is used increasing the pheromone levels. On the other hand, a rank-based elitist strategy is proposed in AS_{rank} to avoid ambushing in a local minimum by electing w best-ranked ants' routes instead of just the best ant in pheromone update. We proposed to update the pheromone by selecting the best tour which achieved at iteration. The steps of the proposed algorithm are summarized as:

```
generate initial solution
compute visibilities
initialize pheromone
while (max number of iterations is not reached)
   for each ant
     while (not all customers are visited)
       select a customer to visit from feasible a customer list
       update vehicle capacity, tour length, and a feasible
        customer list
       if (no feasible customer exists)
         return to depot
         start new vehicle and update feasible customer list
       end if
     end while
   end for
   perform pheromone reinforcement
   save the solution to solution list
   save the best solution
end while
```

Figure 4.1 Algorithm Construction Phase

4.2. Improvement: Tabu search Algorithm

After an ant has constructed its route, Tabu search is performed in an attempt to further reduce the route length and returns. Tabu search is a procedure that uses an initial solution as a starting basis for seeking improved solutions by searching different neighborhoods [20]. The heuristic was first introduced by Glover [58] in 1986 and then used in many applications, especially in types of VRP like Gendreau et al. [59]. Most recently and Righini [19] and Montané and Galvão [20] define Tabu search algorithms for VRPSDP, Brandão [7] suggests new Tabu search algorithm for VRPB, and Cordeau and Laporte [36] define the Tabu search algorithm for the static multi-vehicle DARP.

Tabu search is started from an initial solution. In our hybrid metaheuristic the initial solution for each instance is generated by the Ant Colony Algorithm. In tabu search, generating neighborhoods of the current solution is accomplished by different transformations. Neighborhood generation is defined in more detail in the following parts. After neighborhoods are generated, the best solution is selected as the new current solution and the new iterations start for further improvements. There are also iteration limitations for reducing computational time.

The Tabu search is different from classical local search heuristics in the following manner: Tabu search does not terminate when no improvement possible, which may be a local optimum. In Tabu search, the best valued neighborhood is always selected, even if it is worse than the previous best solution. By that way, a larger portion of search space is explored, and being trapped at local optimums is avoided. A problem called cycling occurs because of letting selection of neighborhoods that do not improve the solution. So the most recently visited neighborhoods are forbidden by a data structure called the tabu list to avoid cycling. Tabu list stores recent search neighborhoods but typically not completely, only the transformation of previous solution, because of the ease of comparison. For example these transformations add an element to the current solution, and then the tabu list stores only an added node and forbids the deletion of that node in very close iterations. Note that, a neighborhood which was created by tabu move can be applied, if it results in an overall best solution. In Tabu search algorithms, tabu lists are called short-term memories, because they handle consecutive movements. To obtain more intensive search in a good area of search space, other mechanisms such as medium and long-term are memories implemented. These memories store some parts of solutions which provide that good area of search space. On the other hand, to drive the search into new areas of the search space, diversification techniques also designed. For example the elements which are not used or less used are forced to enter the solution by storing them in memory data structures [28].

Since diversification enlarges the search area, the number of neighborhoods solutions and the computational time increase. Some limitations generated for intensification such as fixing elements are associated with good solutions. Another limitation is the aspiration level which is a threshold level defined for the objective value. It is a controlling mechanism for the acceptance of a neighborhood. The neighborhood might be accepted, if it results in a better value than the aspiration level; otherwise, rejected.

The theory and many applications of TS may be found in Glover and Laguna [60].

In the following, we first describe the methods for neighborhood generation. Then the evaluation of the neighborhoods for each instance is explained. Finally the tabu list and usage of aspiration criteria are defined.

4.2.1. Neighborhood Generation

Tabu search algorithms start from the initial solution. In our algorithm, the initial solution is generated using ACO. The Neighborhood is a new solution that is generated by making some transformations to the current solution. The transformations are called routines and these are called as SWAP, INS, REI and REV. Routines are originated from the research of Nagy and Salhi [15] which are also largely used in local search and Tabu search algorithms.

The routines are not graded as intra route or inter route levels. Each routine is applied in both levels. Intra route routines focus on the transformations of elements in route of the single vehicle; on the other hand, inter route routines are focused on transformations of elements between the routes of several vehicles.

The routine SWAP is the exchange of 2 customers in a route or between routes. The routine INS, which is “insert” in long form, is the insertion of a customer after another customer, both in a route or between routes. The routine REI, which is “reinsert” in long form, is insertion of depot after a customer. And finally REV, which is “reverse” in long form, is reversing routes.

Our strategy is following the routines in the order SWAP, INS, REI and REV. We have observed that REI and REV routines are useless and time consuming after SWAP and INS routines for most of problem instances, so we have decided not to use them. In each routine the neighborhoods are selected such that maximum reduction in the objective function value is achieved. This procedure is repeated until a local optimum is obtained, i.e. until no further improvement is possible.

4.2.2. Evaluation

In our problem two objectives are defined. One is the common objective of minimizing the total distance travelled and the other is minimizing the total number of returns. To create the Pareto curve with respect to these two objectives, each instance is run for varying parameters in an attempt to obtain pareto efficient objective function values. In the construction, different values of β and γ parameters are considered. In the improvement, these parameters are combined into one term as follows:

$$v_i = d^{\beta} R^{\gamma} \tag{9}$$

Here v_i defines the combined solution value of neighborhood i, that is the composition of distance and returns values with weight parameters β and γ. Greater β value increases the importance of the first objective so tabu search generates solutions more dependent on the first objective, the total distance travelled. Greater γ value increases the importance of second objective so the Tabu search generates solutions more dependent on the second objective, total returns.

4.2.3. Tabu List and Aspiration Criteria

In an improvement phase the tabu list is a data structure that stores the transformations of elements in routines. One change which cannot be done in θ iterations is called the tabu tenure. Tabu tenure is the size of the tabu list in our hybrid metaheuristic. θ values are chosen as constant values as parameter in improvement phase of our solution method. Although diversification is formed by routines, the need for limitation criteria occurs, because of the large search space, especially in cases with a large number of customers. Since the aim of using Tabu search is improving the solution from the construction phase, then the solution generated by ant colony algorithm is defined as aspiration criteria for Tabu search procedure. So the aspiration criteria value is $t_{ACO} = d_{ACO}{}^{\beta} R_{ACO}{}^{\gamma}$ where d_{ACO} is distance value and R_{ACO} is returns value generated at construction phase. Proposed improvement algorithm is;

```
initialize the feasible solution using ACO solution
compute aspiration criteria
while (max number of iterations is not reached)
   clear iteration best solution
   for each routine
     for each transformation
       if (transformation is not in tabu list)
         do transformation
         if (solution is not greater than aspiration criteria)
           save the solution to solution list
```

```
          if (solution is less than iteration best solution)
            add transformation to tabu list
            save the iteration best solution
          end if
        end if
      else if (solution is in the tabu list and solution is
       less than the best solution)
      save the best solution
      end if
    end for
  end for
  if (iteration best is less than best solution)
    save the best solution
  end if
end while
```

Figure 4.2 Algorithm Improvement Phase

4.3. General Framework

The two phases, construction and improvement are explained in previous parts. The following summarizes the steps of the proposed algorithm:

```
while (all instances not tested)
  achieve construction: ant colony algorithm
  achieve improvement: Tabu search algorithm
end while
clear solution list
```

Figure 4.3 Description of the Proposed Algorithm

Note that in line 1 the instances are described by different values of parameters β and γ. Also all of the solutions in all iterations of construction and improvement phases are collected in a data structure called the solution list. A bad solution of an instance may be a good solution for another instance, which is defined by different parameters. Finally the solution list is cleared from recessive solutions, which are not better than other solutions in distance solution, in returns solution or both.

CHAPTER 5

COMPUTATIONAL STUDY

The algorithm is coded using C++ and executed on an Intel Celeron M 1.5 GHz processor with 512 MB RAM. After evaluating certain parameter values we decided to use the following set of parameter values in the computational experiments: q_0 = 0.5; α= 1; β= (0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1); γ= (0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 1); ρ= 0.1; θ= 5. Note that different values of β and γ are used to create the Pareto curve. The number of iterations is 100 for the construction phase and 100 for the improvement phase.

The problem is tested on the well-known VRPSDP benchmark problem sets from the literature. Min [14] defined the first real-life problem instance of VRPSDP with 22 customers, a depot, 10500 vehicle capacity, 20300 total delivery and 19950 total pickup amount. Dethloff [16] created problem sets with 50 customers based on two different geographical scenarios. In the first scenario, called SCA, the coordinates of customers are distributed uniformly over the interval [0,100]. The delivery loads of customers (d_i) are distributed uniformly over the interval [0,100] and the pickup loads of customers (p_i) are determined by a function: $p_i = (0,5 + r_i)d_i$ where r_i distributed uniformly over the interval [0,100]. Also the vehicle capacities (C) are determined by a function: $C = \sum_{s\in I} D_s / \mu$ where μ is defined as minimal number of vehicles and chosen to be 3 or 8. On the other hand, second scenario, called CON, half of the coordinates of customers are distributed uniformly over the interval [0,100] while the coordinates of the other half are distributed uniformly over the interval [100/3,200/3] and delivery, pickup and vehicle capacities are determined in the same way as in SCA.

The Min's problem's data is given as distance matrix between customers and depot, delivery and pickup amounts of customers and vehicle capacity. Dethloff's data sets are given in the same manner: vehicle capacity, number of customers, distance matrix and delivery and pickup amounts of customers. Since these problems do not include the requests of delivery of goods between customers, we modified them as follows: Half of the pickups from a customer to the depot were allocated as deliveries from the customer to other customers. Among all customers 30 percent are selected as the potential destination customers. All customer to customer deliveries defined in this subset and pickup values for these deliveries were assigned randomly. The modified data structure for VRPSDP-ID is as follows: number of customers plus a depot, vehicle capacity, distance matrix and load matrix which includes pickup and delivery between the customers and the depot. Note that the original problems are modified such that benchmarks are not truly unbiased. However, the below reported best known solutions for the original VRPSDP instances are only for information purposes and do not constitute any base for comparison.

Table 5.1 shows the best-known results in the literature for VRPSDP as well as our best distance and best returns values for VRPSDP for all problem instances. Note that 88 is the optimal solution for Min's problem obtained by Halse [61]. The Best Distance column shows the best distance and returns value when only distance objective is considered by parameter values β= 1 and γ= 0. Note that the distance values are close to best-known distance values because of disregarding of returns objective. On the other hand the Best Returns column presents the best returns value by considering the returns objective and disregarding the distance objective. The Best Returns column is generated by parameter values β= 0 and γ= 1. We run the algorithm for 100 iterations for each case and problem. We observe that the average computation time is quite larger than the Ropke's [62] computational time, because of calculation effort of second objective.

	VRPSDP		VRPSDP-ID					
	Best Known		Best Distance‡			Best Returns‡		
Problem	Ref†	Dist	Dist	Ret	CTime	Dist	Ret	CTime
Min	H	88	88	2072	16.5	211	992	29.1
SCA3-0	R	636.1	700	316	631.4	2626	82	617.7
SCA3-1	R	697.8	753	270	624	2834	74	636
SCA3-2	R	659.3	721	346	633.4	2479	123	619.8
SCA3-3	R	680.6	738	337	613	2812	97	625.1
SCA3-4	R	690.5	749	422	621.3	2885	189	625.4
SCA3-5	R	659.9	808	277	609.9	2718	86	613.5
SCA3-6	R	651.1	720	266	627.9	2670	78	631.9
SCA3-7	MG	659.2	755	362	623.5	2870	120	623.4
SCA3-8	R	719.5	781	393	621.7	2844	211	623.8
SCA3-9	R	681	731	279	583.7	2897	96	622.7
SCA8-0	R	975.1	1075	336	733	2724	259	347.1
SCA8-1	R	1052.4	1133	308	747	2868	220	741.3
SCA8-2	R	1044.5	1127	374	752.4	2961	289	752.2
SCA8-3	R	999.1	1127	356	747.9	2771	266	747
SCA8-4	R	1065.5	1264	471	772.9	3039	374	745.4
SCA8-5	R	1027.1	1273	330	724.8	2949	235	731.7
SCA8-6	R	972.5	1125	296	728.9	2649	220	733.7
SCA8-7	R	1061	1213	372	756	2816	289	763.3
SCA8-8	R	1071.2	1212	450	733.1	2893	356	338.5
SCA8-9	R	1060.5	1154	299	758.9	2751	232	763.5
CON3-0	R	616.5	675	404	628.9	1868	219	628.8
CON3-1	R	554.5	597	402	610.5	1903	192	618.1
CON3-2	R	521.4	544	418	622.5	1474	259	622.7
CON3-3	R	591.2	628	459	631.6	1906	251	620.6
CON3-4	R	588.8	669	403	623.8	1755	234	622
CON3-5	R	563.7	659	242	616.7	2049	63	660.9
CON3-6	R	500.8	539	203	619.5	1641	44	639.6
CON3-7	R	576.5	643	308	618.9	2142	105	617.8
CON3-8	R	523.1	569	354	630.6	1848	171	616.2
CON3-9	R	578.2	637	312	625.8	1813	102	629.9
CON8-0	R	857.2	970	433	425.6	2097	361	743.5
CON8-1	R	740.9	829	439	750.5	2005	365	748.7
CON8-2	R	716	783	484	738.3	1588	399	741.2
CON8-3	R	811.1	921	507	766.9	2300	419	771.2
CON8-4	R	772.3	892	446	750.5	1941	374	352.4
CON8-5	R	755.7	912	259	746.5	2069	192	364.7
CON8-6	MG	678.9	748	227	743.8	1688	144	731
CON8-7	MG	814.5	967	351	744.2	2466	271	737.9
CON8-8	R	774	882	408	761.6	1752	321	756.6
CON8-9	MG	809	909	337	748.4	1809	259	758.9
Average		744.3*	834.6	398.2	661.9 s	2298.9	216,0	633.5 s

Table 5.1 Solutions of Min and Dethloff Problems

† H: Halse [61], R: Ropke [62], MG: Montané and Galvão [20]

Since the values of the two objectives, total distance travelled and total returns are given above; the aim of the study is to generate a Pareto curve for this dual objective problem for different levels of the first objective to investigate the trade-off. So the Min's problem, SCA 3-2, SCA 8-4, CON 3-6 and CON 8-8 instances of Dethloff's problem are selected for this aim. The graphs below present the trade of between two objectives for selected problems. The computational times are presented in parenthesis.

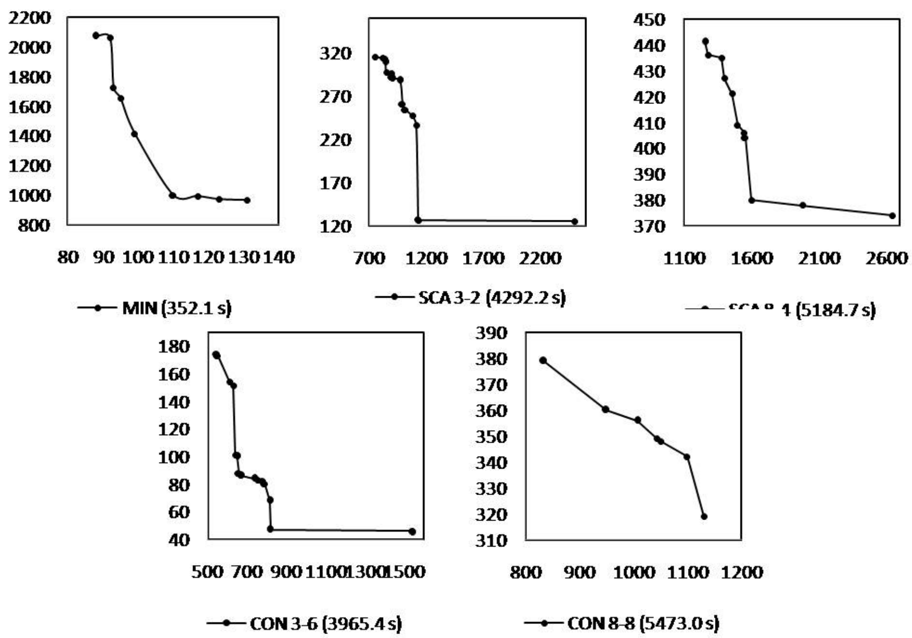

Figure 5.1.a, 5.1.b, 5.1.c, 5.1.d and 5.1.e Pareto Curves for the MIN, SCA 3-2, SCA 8-4, CON 3-6 and CON 8-8 problems

CHAPTER 6

CONCLUSION AND FUTURE WORK

In this thesis, we address a new variant of VRP called as VRPSDP-ID. VRPSDP-ID involves the properties of both VRPSPD and PDP problems. The complexity of VRPSPD and PDP problem has motivated researchers to develop good heuristic-based algorithms. In this paper, we present a two-phase hybrid metaheuristic which consists of an ACO algorithm followed by TS. The performance of the approach is tested using well-known VRPSDP instances from the literature. These instances are modified to include customer-to-customer deliveries. To ensure foresight about solution values original best known solutions for VRPSDP are also mentioned. But benchmarks are not truly unbiased because original problems are modified. The experimental analysis is based on the Pareto efficient values of the dual objectives and the comparison of the travel distances to those in the literature. Although we have been able to find the optimal distance for Min's problem the performance of our approach are inferior with respect to minimizing the total distance objective compared to best distances in the literature. Since we do not have the route information for the test problems we do not have any means for the comparison of minimizing the total returns objective.

A fair comparison of computational effort cannot be done because of the use of different processors. We noticed that our computation times are quite large compared to other heuristics presented in the literature. This is largely due to dual objective structure of the problem. Decreasing the number of iterations, the instances created by different parameters for Pareto curve, and the number of routines in Tabu search may lead to a reduction in the computational effort. However the solution quality may decline due to a decrease in number of points in the Pareto curve or the best solutions for both objectives. The usage of various routines in various sequences in Tabu search or several ACO procedures may result a significant reduction on the computation times which is our focus hereafter.

Finally, researchers may focus on the new defined objective using different methods. In addition, new methodologies can be generated for dual objective based ant colony algorithms.

REFERENCES

[1] Clarke, G., and Wright, W., "Scheduling of vehicles from a central depot to a number of delivery points," Operations Research, Vol: 12, pp. 568-581, 1964.

[2] Toth, P., and Vigo, D., "Models, relaxations and exact approaches for the capacitated vehicle routing problem," Discrete Applied Mathematics, Vol: 123, pp. 487-512, 2002.

[3] Bodin, L.D., Golden, B.L., Assad, A.A., and Ball, M.O., "Routing and scheduling of vehicles and crews, the state of the art," Computers and Operations Research, Special Issue on the Routing and Scheduling of Vehicles and Crews, Vol: 10, No 2, pp. 63-211, 1983.

[4] Toth, P., and Vigo, D., "An exact algorithm for the vehicle routing problem with backhauls," Transportation Science, Vol: 31, pp. 372–385, 1997.

[5] Mingozzi, A., Giorgi, S., and Baldacci, R., "An Exact Method for the Vehicle Routing Problem with Backhauls," Transportation Science, Vol: 33, No 3, pp. 315-329, 1999.

[6] Salhi, S., and Nagy, G., "A cluster insertion heuristic for single and multiple depot vehicle routing problems with backhauling," Journal of the Operational Research Society, Vol: 50, pp. 1034-1042, 1999.

[7] Brandão, J., "A new Tabu search algorithm for the vehicle routing problem with backhauls," European Journal of Operational Research, Vol: 173, pp. 540–555, 2006.

[8] Gendreau, M., Guertin, F., Potvin, J., and Se´guin, R., "Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries," Transportation Research Part C, Vol: 14, pp. 157–174, 2006.

[9] Ropke, S., and Pisinger, D., "A unified heuristic for vehicle routing problems with backhauls," European Journal of Operational Research, Vol: 171, pp. 750-775, 2006.

[10] Goetschalckx, M., and Jacobs-Blecha, C., "The vehicle routing problem with backhauls," European Journal of Operational Research, Vol: 42, pp. 39–51, 1989.

[11] Ganesh, K., and Narendran, T.T., "CLOVES: A cluster-and-search heuristic to solve the vehicle routing problem with delivery and pick-up," European Journal of Operational Research, Vol: 178, Issue 3, pp.699-717, 2007.

[12] Golden, B.L., Baker, E.K., Alfaro, J.L., and Schaffer, J.R., "The vehicle routing problem with backhauling: Two approaches," In R. Hammesfahr, editor, Proceedings of the XXI Annual Meeting of S.F. TIMS, Myrtle Beach, SC, pp. 90-92, 1985.

[13] Wade, A., and Salhi, S., "An investigation into a new class of vehicle routing problem with backhauls," Omega, Vol: 30, pp. 479-487, 2002.

[14] Min, H., "The multiple vehicle routing problem with simultaneous delivery and pick-up points," Transportation Research, Vol: 23A, pp. 377-386, 1989.

[15] Nagy, G., and Salhi, S., "Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries," European Journal of Operational Research, Vol: 162, pp. 126-141, 2005.

[16] Dethloff, J., "Vehicle routing and reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up," OR Spektrum, Vol: 23, pp. 79-96, 2001.

[17] Dethloff, J., "Relation between vehicle routing problems: an insertion heuristic for the vehicle routing problem with simultaneous delivery and pick-up applied to the vehicle routing problem with backhauls," Journal of the Operational Research Society Vol: 53, pp. 115–118., 2002.

[18] Crispim, J., and Brandão, J., "Metaheuristics applied to mixed and simultaneous extensions of vehicle routing problems with backhauls," Journal of the Operational Research Society, Vol: 56, pp. 1296–1302, 2005.

[19] Bianchessi, N., and Righini, G., "Heuristic algorithms for the vehicle routing problem with simultaneous pick-up and delivery," Computers & Operations Research, Vol: 34, pp. 578–594, 2007.

[20] Montané, F.A.T, and Galvão, R.D., "A Tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service," Computers & Operations Research, Vol: 33, pp. 595-619, 2006.

[21] Tillman, F.A., and Hering, R.W., "A study of look-ahead procedure for solving the multiterminal delivery problem," Trans. Res., Vol: 5, pp. 225–229, 1971.

[22] Cordeau, J., Gendrau, M., and Laporte, G., A "Tabu search Heuristic for Periodic and Multi-Depot Vehicle Routing Problems," Vol: 30, Issue 2, pp. 105-119, 1997.

[23] Pisinger, D., and Ropke, S., "A general heuristic for vehicle routing problems," Computers & Operations Research, Vol: 34, Issue 8, pp.2403-2435, 2007.

[24] Solomon, M.M., "Algorithms for the vehicle routing and scheduling problems with time window constraints," Operations Research, Vol: 35, pp. 254-265, 1987.

[25] Rousseau, L., and Gendrau, M., "Using Constraint-Based Operators to Solve the Vehicle Routing Problem with Time Windows," Journal of Heuristics, Vol: 8, pp. 43-58, 2002.

[26] Azi, N., Gendrau, M., and Potvin, J., "An exact algorithm for a single-vehicle routing problem with time windows and multiple routes," European Journal of Operational Research, Vol: 178, 755-766, 2007.

[27] Gelinas, S., Desrochers, M., Desrosiers, J., and Solomon, M., "A new branching strategy for time constrained routing problems with application to backhauling," Annals of Operations Research, Vol: 61, pp. 91-109, 1995.

[28] Duhamel, C., Potvin, J., and Rousseau, J., "A Tabu search Heuristic for the Vehicle Routing Problem with Backhauls and Time Windows," Transportation Science, Vol: 31, No 1, pp. 49-59, 1997.

[29] Cheung, R., and Hang, D., "Multi-attribute label matching algorithms for vehicle routing problems with time windows and backhauls," IIE Transactions, Vol: 35, pp. 191-205, 2003.

[30] Nag, B., Golden, B.L., and Assad, A., "Vehicle routing with site dependencies," In: Golden B, Assad A, editors, Vehicle routing: methods and studies, Amsterdam: Elsevier, pp. 149-159, 1988.

[31] Cordeau, J.F., and Laporte, G., "A Tabu search algorithm for the site dependent vehicle routing problem with time windows," INFOR, Vol: 39, pp.292–298, 2001.

[32] Sariklis, D., and Powell, S., "A heuristic method for the open vehicle routing problem," Journal of the Operational Research Society, Vol: 51, pp. 564–573, 2000.

[33] Brandão, J., "A Tabu search algorithm for the open vehicle routing problem," European Journal of Operational Research 157, Vol: 3, pp. 552–564, 2004.

[34] Wilson, N.H.M., Sussman, J., Wong, H., and Higonnet, B., "Scheduling Algorithms for dial-a-ride systems," Technical Report USL TR-70-13, Urban Systems Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 1971.

[35] Cordeau, J., "A Branch-and-Cut Algorithm for the Dial-a-Ride Problem," Operations Research, Vol: 54, pp. 573–586, 2006.

[36] Cordeau, J., and Laporte, G., "A Tabu search heuristic for the static multi-vehicle dial-a-ride problem," Transportation Research Part B, Vol: 37, pp. 579–594, 2003.

[37] Bergvinsdottir, K., "The GA for solving DARP," Masters Thesis, Informatics and Mathematical Modelling, Technical University of Denmark, 2004.

[38] Coslovich, L., and Pesenti, R., "A two-phase insertion technique of unexpected customers for a dynamic dial-a-ride problem," European Journal of Operational Research, Vol: 175, pp. 1605–1615, 2006.

[39] Borndörfer, R., and Grötschel, M., "Telebus Berlin: Vehicle Scheduling in a Dial-a-Ride System," Technical Report SC 97-23, Konrad-Zuse-Zentrum fur Informationstechnik Berlin, 1997.

[40] Laporte, G., and Ropke, S., "Recent Models and Algorithms for One-to-One Pickup and Delivery Problems," Technical report, G-2006-68, Gerad, Montreal, Canada, 2006.

[41] Cordeau, J.F., "A branch-and-cut algorithm for the dial-a-ride problem" Operations Research, Vol: 54, pp. 573–586, 2006.

[42] Nanry, W.P., and Barnes, J.W., "Solving the pickup and delivery problem with time windows using reactive Tabu search," Transportation Research Part B, Vol: 34, pp. 107-121, 2000.

[43] Dorigo, M., “Ant colony optimization,” Scholarpedia (http://www.scholarpedia.org /article/Ant_Colony_Optimization, last accessed in May 2007), 2007.

[44] Colorni, A., Dorigo, M., and Maniezzo, V., “Distributed optimization by ant colonies,” In: Varela F, Bourgine P (eds), Proceedings of the First European Conference on Artificial Life, Elsevier Publishing, Paris, pp. 134–142, 1991.

[45] Stützle, T., and Dorigo, M., “ACO algorithms for the quadratic assignment problem,” In: Corne D, Dorigo M, Glover F (eds) New ideas in optimization, McGraw-Hill, London, pp. 33-50, 1999.

[46] Colorni, A., Dorigo, M., Maniezzo, V., and Trubian, M., “Ant system for job-shop scheduling,” Belgian Journal of Operations Research, Statistics and Computer Science, Vol: 34, pp. 39-45, 1994.

[47] Gambardella, L.M., and Dorigo, M., “Ant colony system hybridized with a new local search for the sequential ordering problem,” INFORMS Journal on Computing, Vol: 12, pp.237-255, 2000.

[48] Bullnheimer, B., Hartl, R.F., and Strauss, C., “Applying the ant system to the vehicle routing problem,” In: Voss S, Martello S, Osman IH, Roucairol C (eds) Meta-heuristics: advances and trends in local search paradigms for optimization, Kluwer Academic Publishers, Boston, pp. 285-296, 1998.

[49] Bullnheimer, B., Hartl, R.F., and Strauss, C., “An improved ant system algorithm for the vehicle routing problem,” Annals of Operations Research, Vol: 89, pp. 319-328, 1999.

[50] Gambardella, L.M., Taillard, E., and Agazzi, G., “MACS-VRPTW: a multiple ant colony system for vehicle routing problems with time windows,” In: Corne D, Dorigo M, Glover F (eds) New ideas in optimization, McGraw-Hill, London, pp. 63-76, 1999.

[51] Doerner, K.F., Gronalt, M., Hartl, R.F., Reimann, M., Strauss, C., and Stummer, M., “Savings Ants for the vehicle routing problem,” In: Cagnoni S et al. (eds), Proceedings of Evo Workshops 2002, Lecture Notes in Computer Science 2279, pp. 11-20, 2002.

[52] Reimann, M., Doerner, K., and Hartl, R.F., “Insertion based ants for vehicle routing problems with backhauls and time windows” ANTS, Lecture Notes in Computer Sciences 2463, pp. 136-148, 2002.

[53] Reimann, M., Doerner, K., and Hartl, R.F., “Analyzing a unified ant system for the VRP and some of its variants,” In: Cagnoni S et al. (eds), Proceedings of Evo Workshops 2003, Lecture Notes in Computer Science 2611, pp. 300-310, 2003.

[54] Dorigo, M., Maniezzo, V., and Colorni, A., “The ant system: optimization by a colony of cooperating agents,” IEEE Transactions on Systems, Man, and Cybernetics-Part B, Vol: 26, pp. 29-41, 1996.

[55] Dorigo, M., and Gambardella, L.M., “Ant colony system: a cooperative learning approach to the traveling salesman problem,” IEEE Transactions on Evolutionary Computation, Vol: 1, pp. 53-66, 1997.

[56] Stützle, T., and Hoos, H.H., "The MAX-MIN ant system and local search for the traveling salesman problem," In: Bäck T, Michalewicz Z, Yao X (eds), Proceedings of the 1997 IEEE International Conference on Evolutionary Computation (ICEC'97), IEEE Press, Piscataway, NJ, pp. 309–314, 1997.

[57] Dorigo, B., and Stützle, T., "Ant colony optimization," MIT Press, Cambridge, Massachusetts, 2004.

[58] Glover, F., "Future Paths for Integer Programming and Links to Artificial Intelligence," Computers and Operations Research, Vol: 5, pp. 533-549, 1986.

[59] Gendreau, M., Hertz, A., and Laporte, G., "A Tabu search Heuristic for the Vehicle Routing Problem," Management Science, Vol: 40, pp. 1276-1290, 1994.

[60] Glover, F., and Laguna, M., "Tabu search," Kluwer Academic Publishers, Dordrecht, 1997.

[61] Halse, K., "Modeling and solving complex vehicle routing problems," Ph.D. thesis, Institute of Mathematical Modeling, Technical University of Denmark, Lyngby, Denmark, 1992.

[62] Ropke, S., "Heuristic and exact algorithms for vehicle routing problems," Ph.D Thesis, Computer Science Department, University of Copenhagen, 2005.