



# Kent Academic Repository

**Wassan, Niaz A. and Nagy, Gábor (2013) *A Reactive Tabu Search Meta-Heuristic for the Vehicle Routing Problem with Divisible Deliveries and Pickups*. Working paper. University of Kent**

## Downloaded from

<https://kar.kent.ac.uk/51852/> The University of Kent's Academic Repository KAR

## The version of record is available from

[http://www.kent.ac.uk/kbs/research/documents/working-papers/2013/285\\_2013.pdf](http://www.kent.ac.uk/kbs/research/documents/working-papers/2013/285_2013.pdf)

## This document version

UNSPECIFIED

## DOI for this version

## Licence for this version

UNSPECIFIED

## Additional information

## Versions of research works

### Versions of Record

If this version is the version of record, it is the same as the published version available on the publisher's web site. Cite as the published version.

### Author Accepted Manuscripts

If this document is identified as the Author Accepted Manuscript it is the version after peer review but before type setting, copy editing or publisher branding. Cite as Surname, Initial. (Year) 'Title of article'. To be published in *Title of Journal*, Volume and issue numbers [peer-reviewed accepted version]. Available at: DOI or URL (Accessed: date).

## Enquiries

If you have questions about this document contact [ResearchSupport@kent.ac.uk](mailto:ResearchSupport@kent.ac.uk). Please include the URL of the record in KAR. If you believe that your, or a third party's rights have been compromised through this document please see our [Take Down policy](https://www.kent.ac.uk/guides/kar-the-kent-academic-repository#policies) (available from <https://www.kent.ac.uk/guides/kar-the-kent-academic-repository#policies>).

## ***Working Paper Series***

### **A Reactive Tabu Search Meta-Heuristic for the Vehicle Routing Problem with Divisible Deliveries and Pickups**

**Niaz A. Wassan and Gábor Nagy**  
**Kent Business School**

***Working Paper No. 285***  
***December 2013***

# A Reactive Tabu Search Meta-Heuristic for the Vehicle Routing Problem with Divisible Deliveries and Pickups

Niaz A. WASSAN and Gábor NAGY

Centre for Logistics & Heuristic Optimisation, Kent Business School, University of Kent, Canterbury, UK

M. Grazia SPERANZA and Claudia ARCHETTI

Dipartimento Metodi Quantitativi, Università degli Studi di Brescia, Brescia, Italy

*The vehicle routing problem with divisible deliveries and pickups is a new and interesting model within reverse logistics. Each customer may have a pickup and a delivery demand that have to be served with capacitated vehicles. The pickup and the delivery quantities may be served, if beneficial, in two separate visits. The model is placed in the context of other delivery and pickup problems. In this paper we study the savings that can be achieved by allowing the pickup and delivery quantities to be served separately with respect to the case where the quantities have to be served simultaneously. A reactive tabu search heuristic is proposed and the results analysed in depth for a better understanding of the problem structure and an average estimation of the savings due to the possibility of serving pickup and delivery quantities separately.*

**Keywords:** vehicle routing, divisible deliveries and pickups, metaheuristics, tabu search.

## 1. Introduction

This paper focuses on an extension of the class of vehicle routing problems (VRP) known as the *vehicle routing problems with deliveries and pickups* (VRPDP). The main difference between these problems and the VRP is that customers may *receive* or *send* goods, while in the VRP all customers just receive goods from a depot. In the context of these problems, customers who only receive goods are called *delivery* or *linehaul* customers, while those only sending goods are called *pickup* or *backhaul* customers – in many applications, however, customers will both send and receive goods. Given customer distances and demands (these include both pickup and delivery demands) we must find a set of routes to minimise the total travelling cost while meeting customer demands. The main *constraint* is that the capacity of the vehicle cannot be exceeded; however other constraints such as

maximum distance or time windows may exist. From a practical point of view, the VRPDP fall within the field of *reverse logistics*, a field that is gaining increasing importance due to more people becoming environmentally conscious. From a mathematical point of view, this problem is an NP-hard *combinatorial optimisation* problem.

We assume that all delivery goods come from depots and all pickup goods are taken to depots. This excludes the possibility of goods travelling directly from one customer to another and implies that delivery goods and pickup goods are not substitutable. In the VRPDP the vehicle may often carry a mixture of delivery and pickup goods: it starts from the depot carrying only delivery goods, at some stage a mixture of goods may occur, finally the vehicle returns to the depot carrying pickup goods only. At each customer location the load on the vehicle may increase or decrease, resulting in a *fluctuating load*. Hence, checking feasibility must be carried out for each route leg. Thus, the VRPDP is conceptually a harder problem than the VRP, where checking feasibility needs only be done for the first arc of each route. In fact, one of the main difficulties in solving the VRPDP lies in checking load feasibility.

In this paper, we focus on an interesting, but rarely addressed, model within the VRPDP, called the *VRP with Divisible Deliveries and Pickups* (VRPDDP), comparing and contrasting it to its more common counterpart, the *VRP with Simultaneous Deliveries and Pickups* (VRPSDP). As these are not well-known VRP variants, and often the terminology used in the subject literature is confusing, we devote the next section to properly define these models.

The aims of this paper will become clearer once the problem is properly defined. As a quick summary, we are interested in:

1. What characterises problem instances where the VRPDDP is a more appropriate model than the VRPSDP?
2. What characterises the customers that are treated differently in the VRPDDP as compared to the VRPSDP?
3. What shapes do VRPDDP routes take? (Note that unlike in the classical VRP, routes may take shapes other than the classical “petals”.)

The next section presents a classification of the various VRPDP models and re-states our aims in a more precise manner. This is followed by a detailed literature review. Section 4 describes a heuristic based on reactive tabu search. The computational analysis is given in section 5. Finally, we present some conclusions and future research directions.

## 2. The Vehicle Routing Problem with Divisible Deliveries and Pickups

In the VRPDDP a set of customers is given requesting for a delivery and/or a pickup service. A fleet of homogenous vehicles, located at a single depot, are available to serve these customers. All delivery goods come from the depot and all pickup goods are transported to the depot. Each vehicle can transport both pickup and delivery goods and has a maximum capacity limit. Each pickup or delivery request has to be satisfied by a single visit. However, a customer requiring both a pickup and a delivery service can be served by two different visits. The objective is to find a set of vehicle routes satisfying the demands of all customers, never exceeding the vehicle capacity and such that the total distance travelled is minimized.

In order to better understand the structure of the VRPDDP, we first put it into the context of other pickup and delivery problems. Then, we discuss some research issues and this will enable us to re-state the research aims of this paper more precisely.

### 2.1. Classification of VRPDP

A classification of vehicle routing problems with deliveries and pickups can be given according to the patterns of goods movement, the characteristics of the customers and restrictions on goods transported on vehicles. Unfortunately, names of problem versions in the literature are often confusing – different authors may use the same term to mean different problems. While two recent reviews on the VRPDP both present a clear taxonomy, their simultaneous appearance means that the names adopted for the same problem are often different. To help the reader, we will make reference to both taxonomies. For the sake of conciseness, we shall refer to Berbeglia, Cordeau, Gribkovskaia and Laporte (2007) as *BCGL* and to Parragh, Doerner and Hartl (2008) as *PDH*.

The first classification is according to the *transport pattern* of goods.

1. In some problems, an item needs to be moved from a customer to another customer. The depot here serves as a basis for the vehicles but these leave the depot empty and return empty. This transport pattern is relevant to dial-a-ride and courier problems. It is called “*transportation between customers (VRP with Pickup and Delivery – VRPPD)*” by PDH. BCGL divides this problem into two classes called “*many-to-many*” and “*one-to-one*” (referred to as “unpaired” and “paired” by PDH). This is in fact a quite different problem to the VRPDP as defined in section 1. As here vehicles perform pickups before deliveries we prefer to refer to this problem as the VRP with pickups and deliveries (VRPPD) to distinguish it from the VRPDP. For a review of

literature, we refer to the surveys mentioned above and do not consider this type of problem any further.

2. Our focus is on the transport pattern where all goods must either originate from, or be destined to, a depot. Goods may not be taken directly from one vehicle to another. In these problems depots serve as hubs or sorting centres. This is typical in mail transportation or where there are two distinct types of goods (e.g. bottled drinks coming from a depot and empty bottles returning there). It is called “*one-to-many-to-one*” by BCGL and “*transportation to/from a depot (VRP with Backhauling – VRPB)*” by PDH.

The second basis of our taxonomy is the characteristics of the customers.

1. In some problems, customers may either receive or send goods, but not both. These customers are referred to as *linehauls* and *backhauls*, respectively. BCGL refers to this problem class as “*single demands*”.
2. In other problems, there is at least one customer who wishes to both send and receive goods. BCGL refers to this problem class as “*combined demands*”.

Thirdly, we may have some restrictions on the travel pattern of the vehicles. One such restriction is that a vehicle may not carry delivery and pickup goods on board at the same time. (Otherwise, the physical design of the vehicles may necessitate having to unload some recently picked up goods to access delivery goods that are stuck behind them on board, leading to delays. This is known as the “*load-shuffling problem*”.) The second restriction is that customers may request that when a delivery is made to them the pickup goods are taken away at the same time. (A separate visit for delivery and pickup may be deemed inconvenient.) Clearly, the first restriction is more applicable to the case of single demands and the second to the case of combined demands. This yields the following four classes of the VRPDP.

1. The *Vehicle Routing Problem with Backhauling (VRPB)* arises when all customers are either linehaul or backhaul and delivery and pickup goods cannot be transported together. This implies that each vehicle tour visits linehauls first and then backhauls. PDH calls this the “*VRP with Clustered Backhauls*”.
2. The *Vehicle Routing Problem with Mixed Deliveries and Pickups (VRPMDP)* allows linehauls and backhauls to occur in any order on a vehicle tour. PDH calls this the “*VRP with Mixed Linehauls and Backhauls*”.

3. The *Vehicle Routing Problem with Simultaneous Deliveries and Pickups* (VRPSDP) arises when customers wish to both receive and send goods and specify that the pickup must be taken away at the same time when the delivery is made. For this class of problems and the terminology used in this paper is the same as that of PDH.
4. The *Vehicle Routing Problem with Divisible Deliveries and Pickups* (VRPDDP) allows two visits to a customer: one for delivery and another for pickup. Note that we still assume that all of the delivery to a customer is made in a single visit (and same for pickup). Literature does not have any papers with the contrary assumption, but we will briefly look at a problem where deliveries may be split into more than one visit (see section 2.4). We also note that literature often – confusingly – includes this problem class into the previous one. (Even the authors of this paper also referred to this problem as VRPSDP previously. After all, customers here do simultaneously have delivery and pickup needs, even if they may be served separately. The recent PDH review suggested the term “divisible”, an expression we gratefully adopt, since it points to the essential difference between this problem and the VRPSDP.)

Finally, we note that several articles in the literature, especially those seeking theoretical results, restrict themselves to the case of a single vehicle. This is called the Travelling Salesman Problem with Deliveries and Pickups (TSPDP); its subproblems are referred to by the abbreviations TSPB, TSPMDP, TSPSDP and TSPDDP.

## ***2.2. Relationships between various VRPDP versions***

It is important to note that the above problems are not isolated from each other. One particular observation – and a very important one for our research – is that they may sometimes be modelled in terms of another problem.

1. The VRPDDP may be modelled as a VRPMDP by creating two fictitious customers, one purely linehaul and one purely backhaul, co-located at the location of each original customer. Note that this doubles the number of customers which is likely to be detrimental on any solution method (be it exact or heuristic).
2. The VRPSDP cannot be modelled as a VRPMDP as the requirement of simultaneous service may not be satisfied.

3. The VRPMDP may be modelled as a VRPSDP by adding a pickup of zero to each linehaul and a delivery of zero to each backhaul. This does not make the model unduly more complicated. (This also implies that the VRPDDP may be modelled as a VRPSDP.)
4. Although the VRPB and the VRPMDP are totally incompatible with each other, they may both be generalised to a model that is currently gaining recognition, the VRP with Restricted Mixing of Deliveries and Pickups (VRPRMDP), in which there is some restriction on having a mixture of delivery and pickup goods on board.
5. While the “all-deliveries-before-pickups” assumption is somewhat at odds with the VRPDDP and wholly incompatible with the VRPSDP, the idea of restricted mixing could be applied to the VRPDDP. This is likely to force some customers to be served twice.

### **2.3. Research issues in the VRPDDP**

A central research issue in the VRPDDP is the shape of vehicle routes. Our terminology and discussion follows to a large extent the paper of Gribkovskaia *et al.* (2007).

1. A *Hamiltonian* route is where all customers are served simultaneously.
2. A *double-path* route begins with a path from the depot traversing all customers belonging to the route making deliveries only, then follows this path in the opposite direction making pickups only. (In such a route, only one customer is served simultaneously, and no delivery and pickup goods are ever carried together.)
3. A *lasso* route consists of three segments. The first contains deliveries only. In the second segment, both deliveries and pickups are made. The third segment follows the path of the first in the reverse direction, satisfying the pickup needs of these customers.
4. A *figure-of-eight* route is similar to a Hamiltonian one, except that a single customer is served twice.
5. A *general* route is one of no pre-determined shape. Note that all the previous route shapes assume that customers are not split between routes; if they are, the route shapes are deemed to belong to this category.

The following observations were made by researchers on the VRPDDP.

1. Comparing the best route length for different route shapes for the same problem instance, general is better than lasso which is better than Hamiltonian which is better than double-path (Gribkovskaia *et al.*, 2007).



2. Although the optimal general solution is better than the optimal lasso one, the special lasso structure allows for faster heuristics. Thus, in practice better lasso solutions than general ones may be found in the same computing time (Hoff *et al.*, 2009).
3. Relatively few customers are served twice in good quality solutions. Often only one customer is served twice, with a figure-of-eight route shape (Gribkovskaia *et al.*, 2007).
4. Lasso solutions were found to be beneficial in combating the load-shuffling problem, as their structure means that free space is created on the initial deliveries-only route segment (Hoff and Løkketangen, 2006).

#### ***2.4. Relations with the Split Delivery Vehicle Routing Problem***

In the VRPDDP, a customer may be served in more than one visit, but this is restricted to two visits: one for delivery and one for pickup. The delivery and pickup quantities themselves cannot be split into several visits. Relaxing this restriction would yield the VRPPD with Splitting (VRPPDS). This problem has not been investigated. However, research has been done on a special case of this problem, in which there are no pickups – the Split Delivery VRP (SDVRP). Here, the demand of a customer may be served in more than one visit.

Dror and Trudeau (1989) have shown that, if the distances satisfy the triangle inequality, then there always exists an optimal solution of the SDVRP where no two routes have more than one customer in common. This property does not hold for the VRPDDP as shown by the following example. Consider a VRPDDP instance with 4 customers and vehicle capacity equal to 10. Let the customers be located on a straight line at distances 1, 2, 3 and 4 from the depot. Let the delivery and pickup requests of each customer be the following (the first figure is the delivery request while the second is the pickup request): (10,1) for customer 1, (4,5) for customer 2, (5,4) for customer 3 and (1,10) for customer 4. The optimal solution uses only two vehicles and the only way is to build the following two routes: the first route serves customer 1 completely and then the pickup requests of customers 2 and 3. The second route serves the delivery requests of customers 2 and 3 and then serves customer 4 completely. Thus, customers 2 and 3 are visited by both routes.

#### ***2.5. Research aims***

Having defined the problem properly, we can now re-state our research aims more precisely.

1. *What characterises problems where splitting gives significant cost reductions?* This will show in which situations the VRPDDP is applicable. As the VRPDDP is harder to solve than the

VRPSDP, if a problem appears to yield a Hamiltonian solution anyway, it will be easier to solve it straightaway as a VRPSDP. Previous studies on the SDVRP show that cost reductions of up to 50% are possible from the VRP. Would the VRPDDP yield such improvements as compared to the VRPSDP? Would the average demand level be a predictor for cost reductions, as it is in the SDVRP?

2. *What characterises the customers that are being served in more than one visit?* On one hand, the answer to this question will enable the logistics company to focus on these customers and investigate any issues of inconvenience arising out of two visits. On the other hand, it will enable us to design more efficient solution algorithms. As stated before, finding general solutions to the VRPDDP can be time-consuming. If we could identify customers that are unlikely to be served twice in good solutions, we could restrict our problem by not allowing splitting for such customers. As one way of solving the VRPDDP is by converting it into a VRPMDP, if instead of doubling the size of the problem we could just duplicate those customers into fictitious linehauls and backhauls that are likely to be served twice, this would reduce the size of the resulting VRPMDP. We hypothesise that customers' demands are likely to play a part. Do their locations matter? Previous research on the VRPDDP suggests that they do, but studies on the SDVRP suggest that they do not.
3. *What shapes do routes take?* We wish to identify a pattern (if there is any) of split and non-split customers on a route. Again, such analysis will enable us to design more efficient algorithms. If route shapes are restricted to some given patterns, we can create algorithms that are simple modifications of VRPSDP methods, thus faster than general VRPDDP algorithms. Previous studies suggest that lasso and figure-of-eight route shapes often occur in good VRPDDP solutions.

### **3. Literature Review**

According to our focus on the VRPDDP, and its comparison to the VRPSDP, our survey will be more detailed on these topics, and fairly brief on the VRPB, the VRPMDP and the SDVRP. The reader is referred to the two comprehensive surveys of the VRPDP by Berbeglia, Cordeau, Gribkovskaia and Laporte (2007) and by Parragh, Doerner and Hartl (2008). The SDVRP is reviewed by Archetti and Speranza (2008) and Archetti and Speranza (2012). For a comprehensive introduction to vehicle routing, the reader may wish to consult Toth and Vigo (2002).

### ***3.1. The Vehicle Routing Problem with Backhauling (VRPB)***

The VRPB has a long history: it was proposed in 1988 by Casco, Golden and Wasil (1988), who created linehaul routes by a savings method and then inserted backhauls into these. However, the more common heuristic approach later became creating separate linehaul and backhaul routes and then merging these. The standard references on exact results for the VRPB are the works of Toth and Vigo (1997) and Mingozzi, Giorgi and Baldacci (1999). The former used Lagrangean branch-and-bound; the latter branch-and-price. Lately, most researchers applied some sort of a meta-heuristic for the VRPB: Osman and Wassan (2002) used reactive tabu search, Brandão (2006) tabu search, Ropke and Pisinger (2006) large neighbourhood search, Wassan (2007) reactive tabu search and adaptive memory programming, Gajpal and Abad (2009a) multi ant colony system and Zachariadis and Kiranoudis (2012) rich neighbourhoods. Ganesh and Narendran (2007) extended the VRPB to the case where some customers have simultaneous delivery and pickup; all such customers must be served after the pure linehauls but before the pure backhauls. (Due to this strong restriction, we characterise this work as an extension to the VRPB rather than a version of the VRPSDP.)

### ***3.2. The Vehicle Routing Problem with Mixed Deliveries and Pickups (VRPMDP)***

It was pointed out already by Casco, Golden and Wasil (1988), that it is not necessary always to serve all linehauls before commencing servicing the backhauls. Some authors in the 90s applied neighbourhood search methods to solve this problem, see Mosheiov (1998) and Salhi and Nagy (1999). Afterwards meta-heuristics became the method of choice: Wade and Salhi (2003) used ant colony optimisation, Ropke and Pisinger (2006) relied on large neighbourhood search, Bianchessi and Righini (2007) applied tabu search with complex neighbourhoods, Wassan, Nagy and Ahmadi (2008) chose reactive tabu search, while Gajpal and Abad (2009b) opted for ant colony optimisation. However, an exact method, based on a flow formulation and branch-and-cut, was given by Baldacci, Hadjiconstantinou and Mingozzi (2003) for the TSPMDP.

Casco, Golden and Wasil (1988) also observed that in practice some restrictions may apply to mixing delivery and pickup goods on board (but not necessarily going as far as forbidding mixing them as in the VRPB). This version of the VRPMDP is also known as the VRP with Restricted Mixing of Deliveries and Pickups (VRPRMDP). This issue was however forgotten until Wade and Salhi (2002) revisited it and applied a neighbourhood search method for this problem. Reimann and Ulrich (2006) solved this problem using ant colony optimisation, Tütüncü, Carreto and Baker (2009) used

GRASP with guidance, while Nagy, Wassan and Salhi (2013) opted for reactive tabu search. Hoff and Løkketangen (2006) and Hoff *et al.* (2009) addressed this issue in the context of the TSPDDP/VRPDDP (see Section 3.4).

### ***3.3. The Vehicle Routing Problem with Simultaneous Deliveries and Pickups (VRPSDP)***

The VRPSDP was introduced by Min (1989) who solved a small-scale real-world problem using an *ad hoc* procedure that does not guarantee a feasible solution. Gendreau, Laporte and Vigo (1999) and Dethloff (2001) also tackled this problem using simple heuristics.

Three papers present exact solution procedures for the VRPSDP. Angelelli and Mansini (2002) created a set covering formulation for the VRPSDP with time windows. They applied a variety of pricing and branching strategies and solved some small problems to optimality. Dell'Amico, Righini and Salani (2006) proposed a branch-and-price algorithm. They used a variety of heuristic pricing procedures and an exact one based on bidirectional labelling algorithms. This algorithm can solve medium-sized problems to optimality. Subramanian *et al.* (2011) observe that the VRPSDP requires additional constraints as compared to the VRP or the VRPB, namely feasibility must be checked for each arc rather than once for the whole route. Their branch-and-cut algorithm adds these constraints in a lazy fashion.

Most of the papers on the VRPSDP use some sort of a metaheuristic. For the sake of brevity, these are summarised briefly as follows: Crispim and Brandão (2005) [hybrid tabu search and variable neighbourhood search], Tang and Galvão (2006) [tabu search with short- and long-term memory], Chen and Wu (2006) [tabu search], Bianchessi and Righini (2007) [tabu search with complex and variable neighbourhoods], Wassan, Wassan and Nagy (2008) [reactive tabu search], Zachariadis, Tarantilis and Kiranoudis (2009) [hybrid guided local search and tabu search], Ai and Kachitvichyanukul (2009) [particle swarm optimisation], Gajpal and Abad (2009b) [ant colony system], Çatay (2010) [ant colony optimisation], Subramanian *et al.* (2010) [variable neighbourhood descent and iterated local search], Souza *et al.* (2011) [iterated local search and GENIUS], Zachariadis and Kiranoudis (2011) [static move descriptor and promises concept] and Maquera *et al.* (2011) [scatter search]. Nearly all of these works use the operators Shift (insert) and Swap (exchange), many also use the Crossover (splitting and splicing two routes) and 2-opt moves. The VRPPD-specific operator Reverse, introduced by Nagy and Salhi (2005), is also used by Wassan, Wassan and Nagy (2008), Zachariadis, Tarantilis and Kiranoudis (2009), Subramanian *et al.* (2010)

and Souza *et al.* (2011). We note that Crispim and Brandão (2005) and Bianchessi and Righini (2007) allow the search to traverse infeasible solutions.

Finally, Alshamrani, Mathur and Ballou (2007) tackled a version of the VRPSDP, namely the stochastic, periodic TSPSDP. In this problem delivery figures are only known probabilistically while pickup figures are known – as each day, the pickup from a customer equals to its delivery the previous day. A pickup request can be delayed but this incurs in a penalty. The problem is to establish which pickup point to serve and to construct the pickup and delivery route. First, a feasible travelling salesman tour is constructed. Then, this tour is improved using Or-opt.

### ***3.4. The Vehicle Routing Problem with Divisible Deliveries and Pickups (VRPDDP)***

Compared to the previous problem classes, the VRPDDP has only been investigated by a handful of authors, many of whom restricted themselves to the single vehicle case.

Mosheiov (1994) addressed the TSPDDP by converting it into a TSPMDP creating fictitious co-located linehauls and backhauls. He proved that any tour can be made feasible by reinserting the depot into a different edge on the tour. This suggests a simple solution approach: find the optimal TSP tour and insert the depot to the nearest such arc that results in a feasible TSPMDP tour. Optimality of course is lost: the nearest such arc may be located very far from the depot. An alternative insertion-based heuristic is also given.

Anily (1996) also decomposed customers with both a pickup and a delivery demand into pairs of customers. However, somewhat surprisingly, she also assumed that all deliveries must be made before pickups, yielding a VRPB model. This forces customers with combined demands to be served twice, unless they happen to be the last linehaul and first backhaul. A region-division scheme called circular regional partitioning is proposed. An assignment problem is solved to connect linehaul and backhaul routes.

Salhi and Nagy (1999) and Nagy and Salhi (2005) modelled the VRPDDP directly. The problem is initially solved as a VRPSDP using a route-first cluster-second heuristic. The “divisible” aspect is accounted for by a pair of improvement routines called *Neck* and *Unneck*: the first splits a customer into a linehaul and backhaul entity, the second merges these. Neck inserts the backhaul entity into the best position on the vehicle route, hence creating a figure-of-eight shaped route. (It is noted that disabling these routines solves the VRPSDP.) The improvement heuristic also includes standard VRP routines such as 2-Opt, 3-Opt, Shift, Exchange and Perturb. There are two more VRPDDP-specific routines, *Reverse* and *Reinsert*. Reversing the direction of a route can reduce load levels;

this may enable a subsequent insertion of customers. Reinsert, motivated by the work of Mosheiov (1994), inserts the depot to its best possible position on a route. One variant of the heuristic allows infeasible solutions to occur subject to a penalty proportional to the value of maximum load constraint violation in a strategic oscillation framework. An insertion-based method is also developed for comparison purposes. It models the VRPDDP as a VRPMDP and is based on the concept of inserting more than one backhaul at a time, called cluster insertion. Both methods can also cater for multiple depots.

Halskau, Gribkovskaia and Myklebost (2001) introduced the concept of *lasso* solutions (described previously in section 2.3). A lasso construction heuristic is proposed for the TSPDDP. It builds a TSP route sequentially (using *e. g.* the nearest-neighbour method). Each time a load feasibility violation is encountered, a sufficient number of pickups are removed from the beginning of the route to eliminate the violation. Once all customers are on the route, all removed pickups are served on the return way, in the opposite order of deliveries. This method can be adapted to turn a TSP tour into a TSPDDP tour; one just needs to check the tour arc-by-arc for feasibility violations. If one is encountered, the above idea is used to turn the Hamiltonian tour into a lasso. For the VRPDDP, the authors suggest that a cluster-first route-second approach is best, as for each cluster a TSPDDP can be solved using the above ideas.

Hoff and Løkketangen (2006) investigated the TSPDDP with Restricted Mixing. In their model, a mixture of delivery and pickup goods is only allowed if there is sufficient free space to combat the load shuffling problem. They suggest that lasso solutions are beneficial for this model, as the load level on the vehicle can decrease on the outbound spoke of the lasso until sufficient free space is available for deliveries and pickups to be carried out simultaneously. Initial solutions are found using the algorithm of Mosheiov (1994); these are then improved using a tabu search method based on the 2-opt operator. The authors found that lasso solutions can be an acceptable compromise between the reduction of route length and the complications due to load shuffling.

Gribkovskaia *et al.* (2007) discuss various route shapes that may occur in the TSPDDP (our terminology in section 2.3 is based to a large extent on this paper). Some theoretical properties of these route shapes are presented. An initial TSP solution is found using nearest-neighbour or sweep. This tour is then converted to a number of TSPDDP solutions by removing one of its edges. These solutions follow the TSP tour till the removed edge, then return to the first customer, move to the last customer, then follow the TSP tour backwards till the other side of the removed edge and finally return along the tour to the depot. Such tours have far too many double visits and hence a merging procedure is used to eliminate them; each vertex is scanned in turn and if it can feasibly be served in

just one of the directions (forward or backward) then it will be bypassed in the other direction. A shift operator is applied to improve on the best result found. The authors also present a tabu search metaheuristic. This finds an initial TSPSDP solution using Mosheiov's (1994) reinsertion heuristic. The objective function caters for feasibility violation by means of a penalty term. The neighbourhood structure consists of the operators Neck and Unneck, while a reoptimisation procedure based on Shift is carried out after each improving move or periodically. The results show that the best solutions are often non-Hamiltonian; such solutions for most instances contain just one customer who is served twice, in a "figure-of-eight" shape.

Gribkovskaia, Laporte and Shyshou (2008) tackled the TSPSDP with selective pickups. In this model all deliveries must be served but pickups are optional. Each pickup generates a certain revenue; balancing the revenue from these pickups and any detour needed to serve these pickups forms the objective function of the model. A classical heuristic is given, based on an initial Hamiltonian solution. Each customer may be assigned one of three states: simultaneous delivery and pickup, separate delivery and pickup, delivery only – if a customer's status can be changed from delivery-only to simultaneous without creating a feasibility violation, then this is done. Improvement operators include Shift, Neck, Unneck, and "Shifting Pickups": changing the status of a simultaneous customer to separate, if it helps, turns the status of some other customer from delivery-only to simultaneous. The tabu search metaheuristic of Gribkovskaia *et al.* (2007) is also modified to cater for this model.

Hoff *et al.* (2009) extend the model of Hoff and Løkketangen (2006) to the case of several vehicles. A tabu search metaheuristic creating lasso solutions is proposed based on the operators Shift and Swap, and 2-opt as post-optimiser. Infeasible solutions are allowed and attract a penalty. The authors compare the lasso solutions on the one hand to VRPSDP solutions and on the other hand to general (no predetermined route shape) solutions. The latter are found by converting the VRPDDP to a VRPMDP. This doubles the size of the problem and, the authors observe, slows down the heuristic.

### **3.5. The Split Delivery Vehicle Routing Problem (SDVRP)**

The SDVRP was independently introduced by Brenninger-Göthe (1989) and Dror and Trudeau (1989). The former proposed a cluster-first route-second heuristic. The latter proposed a heuristic solution procedure and observed that high demand is a good predictor for splitting and customers which are close to the depot have a higher chance to be split. Archetti, Savelsbergh and Speranza (2006) carried out a worst-case analysis and proved that a SDVRP solution may be up to twice as

good as a VRP solution. Archetti, Hertz and Speranza (2006) solved the SDVRP using tabu search, the results of which were fed into a heuristic-exact hybrid method by Archetti, Savelsbergh and Speranza (2008a). Archetti, Savelsbergh and Speranza (2008b) extended their previous worst-case analysis and showed that the number of SDVRP routes may also be up to 50% fewer than the corresponding VRP routes; in fact they suggested that the route length reduction achievable by splitting is due to the reduction of the number of delivery routes. Their computational experiments also suggest that splitting gives the largest benefit when the average customer demand is between 50% and 75% of the vehicle capacity and the demand variance is small. The experiments do not suggest that customer location is a useful predictor of splitting. For a recent survey on SDVRP the reader is referred to Archetti and Speranza (2012).

Delivery and pickup aspects are rarely addressed in the SDVRP literature. Mosheiov (1998) solved the VRPMDP with split loads. His model created  $d_i$  fictitious co-located customers each with unit demand for each original customer of demand  $d_i$ , resulting in a VRPMDP and a huge increase in problem size.

#### **4. Solution Method**

We modelled the VRPDDP as a VRPMDP by creating a fictitious linehaul and backhaul entities for each customer, a valid approach as discussed in section 2.2., and in line with previous studies, see *e.g.* Mosheiov (1994), Salhi and Nagy (1999) and Hoff *et al.* (2009). The approach has the drawback of having twice as many customers, however our aim here is to analyse split solutions with the view of creating more efficient solution algorithms. Section 2.2 also stated that a VRPMDP can be modelled as a VRPSDP, hence have adopted the VRPSDP algorithm of Wassan, Wassan and Nagy (2008) to solve the VRPDDP.

Initially, we experimented with a modified sweep method for creating initial solutions. However, this proved to be inefficient. Having twice as many customers to cope with, the VRPDDP method struggled to find competitive solutions with the VRPSDP. Hence, we decided to take the solution of the corresponding VRPSDP as the initial solution. This approach guarantees that our algorithm will always yield a non-negative improvement.



#### **4.1. Reactive tabu search**

For complex combinatorial problems heuristics and meta-heuristics are the best way forward for tackling large sized instances. We adapted a reactive tabu search meta-heuristic (RTS) for the VRPDDP. This is based on the RTS heuristic originally developed by Wassan, Wassan and Nagy (2008) for the VRPSDP. *Tabu search*, see Glover and Laguna (1997), is one of the intelligent mechanisms used to avoid being trapped in local optima. *Reactive tabu search*, originally put forward by Battiti and Tecchioli (1994), see also the survey of Battiti, Brunato and Mascia (2008), entails *dynamically* controlling the *tabu list size*, also known sometimes as tabu tenure, which is one of the crucial components of tabu search. The inclusion of the reactive elements (reactive to cycling) into the tabu search components makes this method dynamic that brings a balanced intensification and diversification to the search process. This kind of guiding the search which includes a regular updating of the parameters makes the method self-contained and less sensitive to parameter values.

RTS has been shown to be robust in producing good results for a variety of combinatorial optimisation problems. In particular, it is one of the best methods in the literature for a variety of VRP problems (VRP: Wassan (2006); VRP with time windows: Chiang and Russell (1997); VRP with mix fleet: Wassan and Osman (2002); VRPB: Osman and Wassan (2002), Wassan (2007); VRPSDP: Wassan, Wassan and Nagy (2008); VRPMDP: Wassan, Nagy and Ahmadi (2008)). We may also justify the choice of our heuristic by referring to the experiments of Nagy, Wassan and Salhi (2011), where it was found that on small ( $n \leq 32$ ) instances our RTS heuristic yields an optimality gap of about 1%. Moreover, on the special-structure instances of section 4 our heuristic yielded an optimality gap of about 1%.

#### **4.2. The RTS-VRPDDP algorithm**

The main steps of the RTS-VRPDDP algorithm are shown below.

Step 1: Initialisation phase (reading in the initial solution which is the Hamiltonian solution found by the algorithm of Wassan, Wassan and Nagy (2008) and initialising RTS parameters)

Step 2: Neighbourhood search phase

Step 3: The RTS updating phase

Step 4: Fine tuning phase

Step 5: Stopping phase (stop if stopping criteria are met, otherwise go to Step 2)

Steps 2, 3 and 4 are explained in detail in the following sections.

### ***4.3. The neighbourhood search phase***

The improvement phase of the algorithm (Step 2) is built around two well-known moves, namely *Shift* and *Swap*.

*Shift* entails moving a customer  $i$  from a route to the best possible position on another route. All customers are considered for shifting and the move with the largest decrease in total route length is implemented. The shift process can result in drastic reduction of total distance travelled. It can also be very useful in the sense that it could produce an empty route if a customer is shifted from a single customer route. The computational complexity of this move is  $O(n^2)$ . The main difference from the VRP version of this operator is in checking the feasibility of proposed moves. While in the VRP it is enough to check if the total customer demand is less than the vehicle capacity, for the VRPPD the load can vary from arc to arc. Calculating the arc load for each arc for each possible customer insertion would be excessively time-consuming. We developed a procedure that checks feasibility without increasing the computational complexity of this routine (see Wassan, Wassan and Nagy (2008) for details).

*Swap* involves reallocating two customers, say  $i$  and  $j$ , which are currently on different routes. Customer  $i$  is removed from its route and inserted to a position on the route from which  $j$  is removed. Customer  $j$  is moved to a position on the route formerly containing  $i$ . All pairs of customers are considered for this operation within an iteration. The computational complexity of this move is  $O(n^4)$ . Feasibility check is carried out in a similar fashion to the Shift move.

In Step 2, the entire neighbourhood defined by the moves Shift and Swap is evaluated. If the best move found is not tabu (see section 4.4) or is tabu but surpasses our aspiration criterion (*i.e.*, it yields a better solution than the best one recorded), it is carried out. Otherwise, the best non-tabu move is implemented. Note that the tabu search framework allows for non-improving moves.

### ***4.4. The reactive tabu search updating phase***

We define the tabu status of moves using a tabu list. If the move in Step 2 was a Shift move, where customer  $i$  was removed from route  $r$ , we put  $(i, r)$  onto the tabu list for the next  $tls$  iterations, where  $tls$  is the size of the tabu list (also known as *tabu tenure*). This means that customer  $i$  cannot re-enter route  $r$ , unless an aspiration criterion (see previous section) is met. If the move in Step 2 was a Swap move, the same principle applies.

The remainder of this phase is concerned with dynamically updating the value of  $tls$ . The new solution is checked against previously found solutions. If the search stumbles upon the same solution over and over again, it is appropriate to increase the value of  $tls$  to avoid this. This is known as the *fast reaction* of RTS. In our implementation,  $tls$  is multiplied by a constant ( $>1$ ), if the number of iterations between now and the last time the same solution was found is lower than a threshold. Conversely, if the search progresses well, with no repetitions, we should adjust the  $tls$  to a lower value. This is known as the *slow reaction* of RTS. In our implementation,  $tls$  is multiplied by a constant ( $<1$ ), if its value was constant for more than a set number of iterations. Finally, if the search appears to be totally stuck (there are many frequent repetitions), a *diversification* scheme is implemented – the tabu list is deleted and the search restarted from a randomly chosen solution from the list of previously chosen solutions.

For more details, the reader is referred to Wassan, Wassan and Nagy (2008). The parameter settings used in this research are the same as given in that paper.

#### **4.5. The fine-tuning phase**

This phase contains two operators, reverse and local-shift. These are applied in turn repeatedly to the two routes that were affected in Step 2, until no improvement is found. Note that tabu status of customers is neither checked nor updated during this phase.

*Reverse* is specific to the VRPDP. It does not, in itself, change the solution quality, but was found to be helpful for guiding the search. Operator *Reverse* entails simply reversing the direction of a route, if this results in a decrease in the maximum load on that route. This then makes it easier for customer insertion into the route later on. The computational complexity of this move is  $O(n)$ .

*Local-shift* is an intra-route move that relocates a customer to a different position within the route, if this improves the solution quality. (Non-improving moves are not considered.) The computational complexity of this operator is  $O(n^2)$ . Checking feasibility is done similarly to the *shift* procedure (section 4.3).

## 5. Computational Analysis

We carried out our analysis on a well-known data set, focusing on the three research aims set in section 2.5. The next section introduces our experiments while the three subsequent ones focus on each of the research questions in turn.

### 5.1. Computational experiments

We chose one of the most commonly used set of VRPSDP test instances, namely that proposed by Salhi and Nagy (1999). This set originally contains 28 instances, ranging from 50 to 199 customers. Distances are Euclidean, and, to eliminate any problem associated with computer precision, are rounded to the nearest integer. Note that instances 6 to 10, 13 and 14 have a maximum time constraint, while instances 11 to 14 are clustered. A particular characteristic of this data set is that in some instances there are pairs of customers located at the same coordinates. In instances 4 and 9, customers 80 & 150 and 99 & 104 are co-located. In instances 5 and 10, customer pairs at the same locations are: 3 & 158, 4 & 155, 10 & 189, 58 & 182, 80 & 150, 92 & 151, 99 & 104 and 138 & 154.

Our initial experimentation did not show significant benefits of splitting, hence we devised further instances. Although various sets of instances were tested, for the sake of conciseness and simplicity, we report here only on two in detail. Firstly, we noticed the average demand and pickup values are very small in the Salhi and Nagy (1999) data set, on average 4% of the vehicle capacity and none larger than 22% of the vehicle capacity, leading to a few long routes. Therefore we kept the locations of the Salhi and Nagy (1999) data set, but changed the delivery and pickup values by multiplying all values by four and adding  $0.1C$  to them. This new data set has delivery and pickup values between 10% and 98% of the vehicle capacity, averaging 26%, leading to many short routes. Secondly, noting the example in section 4 that gave a 50% cost improvement, where the difference between delivery and pickup figures was large, we created such a data set. We added  $0.75C$  to the delivery and  $0.2C$  to the pickup of every odd customer, and  $0.2C$  to the delivery and  $0.75C$  to the pickup of every even customer. Coordinates were retained. Thus for every customer either the delivery or the pickup value is between 75% and 97% of the vehicle capacity, while the other value is between 20% and 42%. This means we expect several very short routes.

The RTS algorithm was implemented in Fortran 90 and the experiments executed on a Sun-Fire-V440 with UltraSPARC-IIIi processor, CPU speed 1062 MHz, running Solaris 9. The total number of iterations was set to 1500 for all instances.

*Table 2* compares the simultaneous (Hamiltonian) and divisible (general) solutions for each instance. The former were found using the RTS algorithm of Wassan, Wassan and Nagy (2008), with exactly the same parameters as above. As these were taken as the initial solution to our RTS algorithm, the VRPDDP solution will never be worse than the corresponding VRPSDP solution – instances with a positive improvement are highlighted. *Table 3* then presents detailed VRPDDP solutions for the latter instances. These will be analysed in section 5.4 with regard to route shapes. It shows that even in solutions where splitting gives an improvement, only a minority of the customers are served twice. All these customers are tabulated in *Table 4* and analysed (see section 5.3) with the aim of finding out why they were split.

## ***7.2. What characterises problems where splitting gives significant cost reductions?***

Comparing VRPDDP solutions to VRPSDP solutions (see Table 1), on the original instances more than one third (10 out of 28, 36%) of the instances experienced some cost reduction, although the average reduction was only 0.16% (maximum 1.32%, on CMT2X). The number of vehicles was never reduced. We think this is explained by the delivery and pickup figures being too small, thus reducing the need for splitting.

Looking at the results of the data set where delivery and pickup figures are much larger (up to nearly the vehicle capacity), the situation improves. Route length on average is reduced by 1.93% (maximum 6.16%) and the number of vehicles by 3.12% (maximum 8.57%). For every instance, the route length was reduced; for 18 out of 28 instances, the number of vehicles was also reduced. This already shows that the savings achievable by splitting are significant. (We note that in this data set all delivery and pickup values are  $\geq 0.1C$ . On a very similar data set, not reported here in detail for the sake of brevity, where the range for deliveries and pickups was between 0% and 88% (rather than 10%–98%), the average saving by splitting was only 0.60%. This shows that the absence of very small deliveries and pickups is a significant factor for splitting to be useful.)

The best results were achieved when customers had a large difference between their delivery and pickup, *i.e.* half of the customers had large deliveries and small pickups, while the other had large pickups and small deliveries. On this dataset, an average route length reduction of 11.69% and an average vehicle number reduction of 16.41% were achieved. (Maximum values were 16.07% and 22.22%, respectively.) This is a very significant saving, especially when compared to the theoretical limit of 50%. Even larger savings could be achieved with such type of instances, see *e.g.* the 32%

achieved in section 4. However, it is unlikely that such instances occur in realistic situations. Already in this instance set, most (78%) VRPSDP routes contain only a single customer.

It does not appear that the presence of a maximum time constraint is a predictor of splitting. The reduction in the number of vehicles was the same for constrained and non-constrained instances on all three datasets. The difference in average reduction of route length was insignificant. However, one should expect that if there are very tight maximum time constraints applied, then splitting is unlikely to be beneficial, as vehicles will not be filled to capacity anyway.

There is some evidence that splitting gives more benefit to clustered instances. On the second dataset, where there is a large variation in delivery and pickup figures, and there are many short routes, the reduction in the number of routes is 4.43% for the clustered instances (as opposed to only 2.60% for the non-clustered instances). On the third dataset, where customers have a large imbalance between their delivery and pickup, the reduction in route length is 14.75% for the clustered instances (as opposed to 10.47% for the non-clustered instances). This is sensible, as in clustered instances the inter-customer distances, and hence the detour lengths required to serve a customer twice, are small.

### ***5.3. What characterises the customers that are being served in more than one visit?***

We hypothesised that the customers who are served separately for delivery and pickup may have one or more of the following characteristics: being near the depot, having a large demand or pickup, or being located in a densely populated area. (Our analysis here is based only on the original Salhi and Nagy (1999) instances, since in the additional instances too many customers were split for a meaningful analysis.) Table 4 lists for each of the 61 split customers these characteristics. For each instance, all customers were ranked according to increasing distance from the depot, decreasing demand and decreasing pickup. Customers in the top quarter of each list are classified as “near-depot”, “large-demand” and “large-pickup”. The final column classifies the customer as part of a cluster. A customer is considered to be in a cluster if it has at least five neighbours, where a neighbour is defined as a customer that is within a distance of 10% of the average depot-to-customer distance for that instance.

We found that being *near the depot* is the most important characteristic: about four fifths (48 out of 61, 79%) of split customers exhibit this characteristic. This was expected as it is easy to insert a near-depot delivery to the beginning of a route or a near-depot pickup at the end of a route without greatly increasing the total distance travelled.

Having a *high demand or pickup* is also important: about half (31 out of 61, 51%) of split customers have a high demand or pickup. As load feasibility is the major constraint in our problem, such large customers are the most difficult to place on a route. Hence, splitting them gives additional flexibility and thus leads to better solutions.

Being *located in a densely populated area* has also proved to be a predictor for splitting: about half (29/61, 48%) of split customers have at least five other customers nearby. This makes sense as in dense clusters making a detour to serve a split customer yields only a small increase in route length.

Our hypothesis explained the occurrence of splitting for most (55 out of 61, 90%) split customers. We then looked at the remaining six to see if any other factor existed contributing to their splitting. For five of them we found that the reason they are served twice is that they are co-located with another customer. For a pair of co-located customers it makes sense to first deliver to them both then carry out the two pickups, resulting in one or both of them being split. For easier visualisation, such co-located customers are highlighted in italics in Table 2. This is a particular characteristic of the data set, but if in practice such co-located customers exist then they are certainly good candidates for splitting.

Only one split customer (60 in CMT5X) is not explained by any of the above reasons. Therefore, a promising avenue for further research would be to consider splitting only for customers that exhibit one of the above characteristics.

#### **5.4. What shapes do routes take?**

On the instances where splitting gives significant benefits, the routes contain too few customers for a meaningful analysis. Hence, in this section again we focus on the more realistic original Salhi and Nagy (1999) instances. Table 2 presents all the 119 routes on the instances where splitting occurred. The second column shows the number of split customers on a route, while the third describes the shape of the route. We note that slightly more than half the routes (68 out of 119, 57%) contain one or more split customers. From now on, we look at these routes only.

In line with expectations, nearly two thirds (44 out of 68, 65%) of routes contain just one split customer, with very few (4/68, 6%) containing more than three.

Most (56/68, 82%) routes are in the shape of a simple *cycle*, denoted by “C” in Table 3. (To avoid confusion with the terminology of Gribkovskaia *et al.* (2007), we do not refer to such routes as Hamiltonian; this term is reserved for routes where every customer receives a simultaneous service.)

Having taken a closer look at the remaining 12 routes, we saw that the issue of co-located customers, a characteristic of instances 4, 5, 9 and 10, plays a part here. For example, on a cursory look at the first route in CMT5X, it appears that the route zig-zags between customers 10 and 189. A closer look reveals that these customers are located at the same coordinates. Hence, if we represent both with a single vertex, this route actually will have a cyclical shape. To highlight this issue, co-located customers are marked in italics in Table 2. Routes that become cyclical once visiting such customer pairs is considered as a single stop are denoted by “Y”. Including such routes, all but two routes can be described as cyclical-shaped. This is in marked difference to studies on the TSPDDP, where lasso and figure-of-eight solutions are common. Of course, in the TSPDDP, a customer cannot be split between two routes; while in our experiments, if we disregard co-located customers, only 3 customers are served by the same vehicle for delivery and pickup while the remaining 46 are split between routes.

One of the remaining routes (the first route in CMT10X) is *lasso-shaped*, with one split customer (28) that is served for delivery at the very beginning of the route and for pickup at the very end. Between these stops, there is one delivery-only customer and nine non-split customers. The other route (the first route in CMT10Y, 0–166–199P–125D–45–125P–199D–18–0, length 58) has a more surprising shape and even has a pickup before a delivery. On closer observation, we notice that all its customers are placed nearly on a straight line. Due to using integer distances, this tour has the same length as the optimal (Hamiltonian) TSP tour 0–166–199–125–45–18–0. The total of delivery and pickup demand is much less than the vehicle capacity, thus the order of deliveries and pickups does not matter.

Split customers tend to occur at the beginning or the end of the routes – which makes sense as they also tend to be near the depot. However, for about a quarter of the routes (19/68, 28%) they occur mid-route.

Future research on an improved solution algorithm can benefit from the above observations. As customers tend to be split between routes rather than within a route, we should develop an operator that can achieve this. For example, “Splitshift” would duplicate a simultaneous customer and insert either its linehaul or its backhaul into another route. (In this case customers would not be duplicated at the start but by this operator.) Such an operator may work best in an environment where infeasible solutions are allowed, as it could help to achieve/restore feasibility. (However, if the instance contains a few co-located customers these should be modelled as separate linehaul and backhaul entities straightaway to facilitate serving their delivery needs before their pickup needs.) Finally, we



must allow split customers to occur freely – allowing them to be placed only at the beginning or the end of a route would be too restrictive.

## 6. Conclusions and Suggestions

We investigated the Vehicle Routing Problem with Divisible Deliveries and Pickups (VRPDDP), a rarely-addressed extension of the VRP. We placed the VRPDDP in the context of other VRP extensions and presented a reactive tabu search metaheuristic. Our computational experiments led us to the following three main conclusions.

1. Serving customers twice can often reduce costs and – perhaps even more importantly – the number of vehicles required. It appears that the presence of very small deliveries and pickups is not conducive to splitting. Route length and the number of vehicles are reduced considerably when the delivery and pickup figures vary within a wide range. The benefits of splitting are shown to be even more significant for instances where there is a large difference between delivery and pickup values. Splitting seems more beneficial for clustered instances, however, the presence of a maximum time constraint does not appear to be a predictor for splitting.
2. Three important characteristics of customers who are served twice were observed: they are near the depot, they have a high delivery or pickup demand, or they are located in a dense cluster of customers, with the first factor being especially significant. These observations lead us to believe that good solutions could be achieved if we consider splitting only for customers with such characteristics.
3. Almost all routes take the shape of a cycle, with customers being split across (rather than within) routes. Split customers very often, but not always, occur at the beginning or the end of a route.

We plan to take forward this research as follows.

1. Incorporate the findings of our analysis to an improved solution algorithm. Such a method may only allow splitting customers with the characteristics described in the previous section. This way, we have a good chance of still having the optimum solution within the feasible set, but by not doubling the number of customers the method should work faster and have a better chance of finding good solutions.
2. Extend the scope of our analysis to the VRPDDPS, allowing customers' delivery and pickup requests to be served in several visits.

3. Merging our lines of research in this paper and in Nagy, Wassan and Salhi (2013), we wish to investigate the VRPDDP with Restricted Mixing. This model, introduced by Hoff and Løkketangen (2006), forces customers to be served separately to avoid situations where there is a mixture of delivery and pickup goods on board but not enough space to have access to both kinds of goods.

## Acknowledgements

The authors wish to express their gratitude to the Economic and Social Research Council (ESRC) of the United Kingdom, grant RES-000-22-3451, for their generous financial assistance towards this project.

## References

- T. J. Ai and V. Kachitvichyanukul (2009), A Particle Swarm Optimization for the Vehicle Routing Problem with Simultaneous Pickup and Delivery, *Computers & Operations Research* **36**, 1693–1702.
- A. Alshamrani, K. Mathur and R. H. Ballou (2007), Reverse Logistics: Simultaneous Design of Delivery Routes and Returns Strategies, *Computers & Operations Research* **34**, 595–619.
- E. Angelelli and R. Mansini (2002), The Vehicle Routing Problem with Time Windows and Simultaneous Pickup and Delivery. In: M. G. Speranza, A. Klose and L. N. Van Wassenhove (eds), *Quantitative Approaches to Distribution Logistics and Supply Chain Management*. Springer, Berlin, 249–267.
- S. Anily (1996), The Vehicle Routing Problem with Delivery and Back-haul Options, *Naval Research Logistics* **43**, 415–434.
- C. Archetti, A. Hertz and M. G. Speranza (2006), A Tabu Search Algorithm for the Split Delivery Vehicle Routing Problem, *Transportation Science* **40**, 64–73.
- C. Archetti, M. W. P. Savelsbergh and M. G. Speranza (2006), Worst-case Analysis for Split Delivery Vehicle Routing Problems, *Transportation Science* **40**, 226–234.
- C. Archetti, M. W. P. Savelsbergh and M. G. Speranza (2008a), An Optimization-based Heuristic for the Split Delivery Vehicle Routing Problem, *Transportation Science* **42**, 22–31.
- C. Archetti, M. W. P. Savelsbergh and M. G. Speranza (2008b), To Split or Not to Split: That is the Question, *Transportation Research E* **44**, 114–123.
- C. Archetti and M. G. Speranza (2008), The Split Delivery Vehicle Routing Problem: A Survey. In: B. Golden, S. Raghavan and E. Wasil (eds), *The Vehicle Routing Problem: Latest Advances and New Challenges*. Springer, New York, 103–122.
- C. Archetti and M. G. Speranza (2012), Vehicle routing problems with split deliveries, *International Transactions in Operational Research* **39**, 3–22.
- R. Baldacci, E. Hadjiconstantinou and A. Mingozzi (2003), An Exact Algorithm for the Traveling Salesman Problem with Deliveries and Collections, *Networks* **42**, 26–41.
- R. Battiti and G. Tecchioli (1994), Reactive Tabu Search, *ORSA Journal on Computing* **6**, 126–140.
- R. Battiti, M. Brunato and F. Mascia (2008), *Reactive Search and Intelligent Optimization*, Springer, Berlin.
- G. Berbeglia, J-F. Cordeau, I. Gribkovskaia and G. Laporte (2007), Static Pickup and Delivery Problems: A Classification Scheme and Survey, *TOP* **15**, 1–31.
- N. Bianchessi and G. Righini (2007), Heuristic Algorithms for the Vehicle Routing Problem with Simultaneous Pick-up and Delivery, *Computers & Operations Research* **34**, 578–594.

- J. Brandão (2006), A New Tabu Search Algorithm for the Vehicle Routing Problem with Backhauls, *European Journal of Operational Research* **173**, 540-555.
- M. Brenninger-Göthe (1989), *Two Vehicle Routing Problems: Mathematical Programming Approaches*, Dissertation, Department of Mathematics, Linköping University.
- D. O. Casco, B. L. Golden and E. A. Wasil (1988), Vehicle Routing with Backhauls: Models, Algorithms and Case Studies. In: B. L. Golden and A. A. Assad (eds): *Vehicle Routing: Methods and Studies*, Elsevier, 127-147.
- B. Çatay (2010), A New Saving-based Ant Algorithm for the Vehicle Routing Problem with Simultaneous Pickup and Delivery, *Expert Systems with Applications* **37**, 6809–6817.
- W. C. Chiang and R. A. Russell (1997), A Reactive Tabu Search Metaheuristic for the Vehicle Routing Problem with Time Windows, *INFORMS Journal on Computing* **9**, 417-430.
- M. Dell'Amico, G. Righini and M. Salani (2006), A Branch-and-Price Approach to the Vehicle Routing Problem with Simultaneous Distribution and Collection, *Transportation Science* **40**, 235–247.
- M. Dror and P. Trudeau (1989), Savings by Split Delivery Routing, *Transportation Science* **23**, 141-145.
- Y. Gajpal and P. L. Abad (2009a), Multi-Ant Colony System (MACS) for a Vehicle Routing Problem with Backhauls, *European Journal of Operational Research* **196**, 102-117.
- Y. Gajpal and P. L. Abad (2009b), An Ant Colony System (ACS) for Vehicle Routing Problem with Simultaneous Delivery and Pickup, *Computers & Operations Research* **36**, 3215-3223.
- K. Ganesh and T. T. Narendran (2007), CLOVES: A Cluster-and-Search Heuristic to Solve the Vehicle Routing Problem with Delivery and Pick-up, *European Journal of Operational Research* **178**, 699–717.
- F. Glover and M. Laguna (1997), *Tabu Search*, Kluwer, New York.
- I. Gribkovskaia, Ø. Halskau, G. Laporte and M. Vlček (2007), General Solutions to the Single Vehicle Routing Problem with Pickups and Deliveries, *European Journal of Operational Research* **180**, 568–584.
- I. Gribkovskaia, G. Laporte and A. Shyshou (2008), The Single Vehicle Routing Problem with Deliveries and Selective Pickups, *Computers & Operations Research* **35**, 2908–2924.
- Ø. Halskau, I. Gribkovskaia and K. N. B. Myklebost (2001), Models for Pick-up and Deliveries from Depots with Lasso Solutions. In: G. Stefnansson and B. Tilanus (eds), *Proceedings of the 13th Annual Conference on Logistics Research NOFOMA 2001*. Chalmers University of Technology, Göteborg, 279–293.
- A. Hoff and A. Løkketangen (2006), Creating Lasso-solutions for the Traveling Salesman Problem with Pickup and Delivery by Tabu Search, *Central European Journal of Operations Research* **14**, 125-140.
- A. Hoff, G. Laporte, A. Løkketangen and I. Gribkovskaia (2009), Lasso Solution Strategies for the Vehicle Routing Problem with Pickups and Deliveries, *European Journal of Operational Research* **192**, 755-766.
- G. Maquera, M. Laguna, D. A. Gandelman and A. P. Sant'Anna (2011), Scatter Search Applied to the Vehicle Routing Problem with Simultaneous Delivery and Pickup, *International Journal of Applied Metaheuristic Computing* **2**, 1-20.
- A. Mingozzi, S. Giorgi and R. Baldacci (1999), An Exact Method for the Vehicle Routing Problem with Backhauls, *Transportation Science* **33**, 315–329.
- G. Mosheiov (1994), The Travelling Salesman Problem with Pick-up and Delivery, *European Journal of Operational Research* **79**, 299–310.
- G. Mosheiov (1998), Vehicle Routing with Pickup and Delivery: Tour-partitioning Heuristics, *Computers and Industrial Engineering* **34**, 669-684.
- G. Nagy and S. Salhi (2005), Heuristic Algorithms for Single and Multiple Depot Vehicle Routing Problems with Pickups and Deliveries, *European Journal of Operational Research* **162**, 126-141.
- G. Nagy, N. A. Wassan and S. Salhi (2013), The Vehicle Routing Problem with Restricted Mixing of Deliveries and Pickups, *Journal of Scheduling* **16**, 199–213.
- I. H. Osman and N. A. Wassan (2002), A Reactive Tabu Search Meta-heuristic for the Vehicle Routing Problem with Backhauls, *Journal of Scheduling* **5**, 263-285.
- S. N. Parragh, K. F. Doerner and R. F. Hartl (2008), A Survey on Pickup and Delivery Problems. Part I: Transportation between Customers and Depot, *Journal für Betriebswirtschaft* **58**, 21-51.
- M. Reimann and H. Ulrich (2006), Comparing backhauling strategies in vehicle routing using ant colony optimization, *Central European Journal of Operations Research* **14**, 105–123.
- S. Ropke and D. Pisinger (2006), A Unified Heuristic for a Large Class of Vehicle Routing Problems with Backhauls, *European Journal of Operational Research* **171**, 750-775.
- S. Salhi and G. Nagy (1999), A Cluster Insertion Heuristic for Single and Multiple Depot Vehicle Routing Problems with Backhauling, *Journal of the Operational Research Society* **50**, 1034-1042.
- S. Salhi (2006), Heuristic Search in Action. In: S. Salhi (ed), *OR48 Keynote Papers*, Government Operational Research Service, London.

- M. J. F. Souza, M. T. Mine, M. S. A. Silva, L. S. Ochi and A. Subramanian (2011), A Hybrid Heuristic, Based on Iterated Local Search and GENIUS, for the Vehicle Routing Problem with Simultaneous Pickup and Delivery, *International Journal of Logistics Systems and Management* **10**, 142-157.
- A. Subramanian, L. M. A. Drummond, C. Bentes, L. S. Ochi and R. Farias (2010), A Parallel Heuristic for the Vehicle Routing Problem with Simultaneous Pickup and Delivery, *Computers & Operations Research* **37**, 1899–1911.
- A. Subramanian, E. Uchoa, A. A. Pessoa and L. S. Ochi (2011), Branch-and-cut with Lazy Separation for the Vehicle Routing Problem with Simultaneous Pickup and Delivery, *Operations Research Letters* **39**, 338–341.
- P. Toth and D. Vigo (1997), An Exact Algorithm for the Vehicle Routing Problem with Backhauls, *Transportation Science* **31**, 372-385.
- P. Toth and D. Vigo (2002), *The Vehicle Routing Problem*, Society for Industrial and Applied Mathematics, Philadelphia.
- A. C. Wade and S. Salhi (2002), An Investigation into a New Class of Vehicle Routing Problem with Backhauls, *Omega* **30**, 479-487.
- A. C. Wade and S. Salhi (2003), An Ant System Algorithm for the Mixed Vehicle Routing Problem with Backhauls. In: M. G. Resende and J. P. de Sousa (eds.), *Metaheuristics: Computer Decision-Making*, Kluwer, New York, 699-719.
- N. A. Wassan (2006), A Reactive Tabu Search for the Vehicle Routing Problem, *Journal of the Operational Research Society* **57**, 111-116.
- N. A. Wassan (2007), Reactive Tabu Adaptive Memory Programming Search for the Vehicle Routing Problem with Backhauls, *Journal of the Operational Research Society* **58**, 1630-1641.
- N. A. Wassan, G. Nagy and S. Ahmadi (2008), A Heuristic Method for the Vehicle Routing Problem with Mixed Deliveries and Pickups, *Journal of Scheduling* **11**, 149-161.
- N. A. Wassan and I. H. Osman (2002), Tabu Search Variants for the Mix Fleet Vehicle Routing Problem, *Journal of the Operational Research Society* **53**, 768–782.
- N. A. Wassan, A. H. Wassan and G. Nagy (2008), A Reactive Tabu Search Algorithm for the Vehicle Routing Problem with Simultaneous Pickups and Deliveries, *Journal of Combinatorial Optimization* **15**, 368–386.
- G. Y. Tütüncü, C. A. C. Carreto and B. M. Baker (2009), A Visual Interactive Approach to Classical and Mixed Vehicle Routing Problems with Backhauls, *Omega* **37**, 138-154.
- E. E. Zachariadis, C. D. Tarantilis and C. T. Kiranoudis (2009), A Hybrid Metaheuristic Algorithm for the Vehicle Routing Problem with Simultaneous Delivery and Pick-up Service, *Expert Systems with Applications* **36**, 1070–1081.
- E. E. Zachariadis and C. T. Kiranoudis (2011), A Local Search Metaheuristic Algorithm for the Vehicle Routing Problem with Simultaneous Pick-ups and Deliveries, *Expert Systems with Applications* **38**, 2717–2726.
- E. E. Zachariadis and C. T. Kiranoudis (2012), An Effective Local Search Approach for the Vehicle Routing Problem with Backhauls, *Expert Systems with Applications* **39**, 3174–3184.

**Table 2. Comparison of VRPSDP and VRPDDP results**

instance	size	original						large variation in deliveries and pickups						large difference between delivery and pickup					
		VRPSDP		VRPDDP		improvement		VRPSDP		VRPDDP		improvement		VRPSDP		VRPDDP		improvement	
		z	k	z	k	z	k	z	k	z	k	z	k	z	k	z	k	z	k
CMT1X	50	478	3	478	3	0.00%	0.00%	1185	19	1169	19	1.35%	0.00%	2315	45	1943	35	16.07%	22.22%
CMT1Y	50	476	3	476	3	0.00%	0.00%	1101	18	1093	18	0.73%	0.00%	2245	45	1968	35	12.34%	22.22%
CMT2X	75	713	7	712	7	0.14%	0.00%	2074	38	1978	36	4.63%	5.26%	3586	74	3217	60	10.29%	18.92%
CMT2Y	75	694	7	694	7	0.00%	0.00%	2028	35	1903	32	6.16%	8.57%	3586	74	3292	63	8.20%	14.86%
CMT3X	100	727	5	726	5	0.14%	0.00%	2055	32	2009	30	2.24%	6.25%	4210	80	3805	69	9.62%	13.75%
CMT3Y	100	723	5	723	5	0.00%	0.00%	1884	28	1860	28	1.27%	0.00%	4146	80	3728	64	10.08%	20.00%
CMT4X	150	901	8	901	8	0.00%	0.00%	2884	47	2806	45	2.70%	4.26%	6263	121	5738	105	8.38%	13.22%
CMT4Y	150	859	7	859	7	0.00%	0.00%	2612	42	2605	42	0.27%	0.00%	6017	116	5328	97	11.45%	16.38%
CMT5X	199	1090	11	1083	11	0.65%	0.00%	3865	68	3843	68	0.57%	0.00%	8366	167	7554	147	9.71%	11.98%
CMT5Y	199	1053	10	1052	10	0.10%	0.00%	3403	59	3371	58	0.94%	1.69%	8080	163	7391	142	8.53%	12.88%
CMT6X	50	555	6	555	6	0.00%	0.00%	1185	19	1169	19	1.35%	0.00%	2315	45	1943	35	16.07%	22.22%
CMT6Y	50	556	6	556	6	0.00%	0.00%	1101	18	1093	18	0.73%	0.00%	2245	45	1968	35	12.34%	22.22%
CMT7X	75	899	11	899	11	0.00%	0.00%	2083	38	2027	36	2.69%	5.26%	3586	74	3217	60	10.29%	18.92%
CMT7Y	75	902	11	902	11	0.00%	0.00%	2028	35	1903	32	6.16%	8.57%	3586	74	3292	63	8.20%	14.86%
CMT8X	100	874	9	874	9	0.00%	0.00%	2055	32	2009	30	2.24%	6.25%	4210	80	3805	69	9.62%	13.75%
CMT8Y	100	867	9	867	9	0.00%	0.00%	1884	28	1860	28	1.27%	0.00%	4146	80	3728	64	10.08%	20.00%
CMT9X	150	1200	15	1193	15	0.59%	0.00%	2884	47	2806	45	2.70%	4.26%	6263	121	5738	105	8.38%	13.22%
CMT9Y	150	1215	15	1215	15	0.00%	0.00%	2612	42	2605	42	0.27%	0.00%	6017	116	5328	97	11.45%	16.38%
CMT10X	199	1439	19	1438	19	0.07%	0.00%	3865	68	3843	68	0.57%	0.00%	8366	167	7554	147	9.71%	11.98%
CMT10Y	199	1467	19	1452	19	1.03%	0.00%	3403	59	3371	58	0.94%	1.69%	8080	163	7391	142	8.53%	12.88%
CMT11X	120	1009	5	1009	5	0.00%	0.00%	3941	32	3894	31	1.19%	3.13%	10024	90	8608	77	14.13%	14.44%
CMT11Y	120	905	4	905	4	0.00%	0.00%	3333	29	3309	28	0.72%	3.45%	9727	89	8372	75	13.93%	15.73%
CMT12X	100	680	6	680	6	0.00%	0.00%	2609	37	2535	34	2.84%	8.11%	5328	83	4523	68	15.11%	18.07%
CMT12Y	100	632	5	632	5	0.00%	0.00%	2289	33	2233	32	2.45%	3.03%	5114	80	4304	68	15.84%	15.00%
CMT13X	120	1647	11	1644	11	0.18%	0.00%	3941	32	3894	31	1.19%	3.13%	10024	90	8608	77	14.13%	14.44%
CMT13Y	120	1710	12	1708	12	0.12%	0.00%	3333	29	3309	28	0.72%	3.45%	9727	89	8372	75	13.93%	15.73%
CMT14X	100	842	10	831	10	1.32%	0.00%	2609	37	2535	34	2.84%	8.11%	5328	83	4523	68	15.11%	18.07%
CMT14Y	100	854	11	854	11	0.00%	0.00%	2289	33	2233	32	2.45%	3.03%	5114	80	4304	68	15.84%	15.00%
<b>average</b>						<b>0.16%</b>	<b>0.00%</b>					<b>1.93%</b>	<b>3.12%</b>					<b>11.69%</b>	<b>16.41%</b>

z: solution value, k: number of vehicles, improvement:  $\frac{z(VRPSDP) - z(VRPDDP)}{z(VRPSDP)}$  or  $\frac{k(VRPSDP) - k(VRPDDP)}{k(VRPSDP)}$

**Table 3. Characteristics of split customers**

instance	customer	near depot?	large demand?	large pickup?	cluster?	
CMT2X	67	YES	YES	NO	NO	
	75	YES	YES	NO	NO	
CMT3X	30	NO	YES	NO	NO	
CMT5X	10	NO	NO	NO	NO	
	27	YES	YES	NO	NO	
	28	YES	YES	NO	NO	
	53	YES	NO	NO	NO	
	60	NO	NO	NO	NO	
	80	NO	NO	NO	NO	
	111	YES	YES	NO	NO	
	112	YES	YES	NO	NO	
	138	YES	NO	NO	NO	
	146	YES	NO	NO	NO	
	150	NO	NO	NO	NO	
	154	YES	YES	NO	NO	
	156	YES	YES	NO	NO	
	167	YES	YES	NO	NO	
	189	NO	NO	NO	NO	
	196	YES	YES	NO	YES	
CMT5Y	10	NO	NO	NO	NO	
	28	YES	NO	YES	NO	
	68	NO	YES	YES	NO	
	156	YES	NO	YES	NO	
	190	YES	YES	YES	NO	
CMT9X	1	YES	NO	NO	NO	
	28	YES	YES	NO	YES	
	53	YES	NO	NO	YES	
	96	YES	NO	NO	YES	
	104	YES	NO	NO	YES	
	111	YES	YES	NO	YES	
	138	YES	NO	NO	YES	
	146	YES	NO	NO	NO	
CMT10X	28	YES	YES	NO	NO	
	53	YES	NO	NO	YES	
	80	NO	NO	NO	YES	
	92	NO	NO	NO	YES	
	111	YES	YES	NO	NO	
	152	YES	YES	NO	YES	
	156	YES	YES	NO	YES	
	196	YES	YES	NO	YES	
	CMT10Y	28	YES	NO	YES	NO
		53	YES	YES	NO	YES
69		YES	NO	NO	YES	
96		YES	NO	NO	YES	
104		YES	NO	NO	YES	
111		YES	YES	NO	NO	
125		NO	NO	YES	YES	
138		YES	NO	NO	NO	
154		YES	NO	YES	NO	
199		NO	NO	NO	YES	
CMT13X	89	YES	NO	NO	YES	
	99	YES	NO	NO	YES	
CMT13Y	39	NO	NO	NO	YES	
	87	YES	YES	YES	YES	
	90	YES	NO	YES	YES	
	91	YES	NO	NO	YES	
	92	YES	YES	NO	YES	
	94	YES	NO	NO	YES	
	97	YES	YES	NO	YES	
105	YES	NO	NO	YES		
CMT14X	43	YES	NO	NO	YES	

**Table 4. Detailed configurations for routes with split solutions**

instance	t	S	Route
CMT 2X	1	C	0- <b>75P</b> -68-2-62-73-1-43-41-42-64-22-61-28-74-0
	1	C	0- <b>75D</b> -30-48-21-47-36-69-71-60-70-20-37-5-0
	1	C	0- <b>67P</b> -34-52-27-13-54-57-15-29-45-4-0
	1	C	0- <b>67D</b> -35-14-59-19-8-46-0
CMT 3X	1	C	0-27-69-1-50-33-81-9-51- <b>30D</b> -32-90-63-10-62-19-11-64-49-36-47-46-45-17-84-5-60-89-0
	1	C	0-28-76-77-3-79-78-34-35-71-65-66-20- <b>30P</b> -70-31-88-7-48-82-8-83-18-52-0
CMT 5X	4	Y	0-132-69-162-101-70-30-20-188-66-128-160-131-32-181-63-126-90-108- <b>189D-10D-189P-10P</b> -31- <b>167P-146D</b> -0
	1	C	0- <b>111P</b> -50-102-157-9-135-35-136-65-71-161-103-51-122-1-0
	3	C	0- <b>27P</b> -176-33-81-120-164-34-78-169-129-79-185- <b>196P</b> -184- <b>28P</b> -0
	5	Y	0- <b>111D</b> -76- <b>196D</b> -116-77-3-158-121-29-24-134-163-68- <b>80D-150D-80P-150P</b> -177-109-12- <b>154P</b> -0
	1	C	0- <b>53D</b> -105-180-198-110-155-4-139-187-39-67-170-25-55-165-130-54-179-149-26-0
	4	Y	0- <b>28D-138P-154D-138D</b> -195-21-72-197-56-186-23-75-133-22-41-145-171-74-73-40- <b>53P</b> -0
	1	C	0- <b>112D</b> -183-94-95-117-97-87-172-43-15-57-178-115-2-58-152-0
	1	C	0-137-144-42-142-14-38-140-44-119-192-91-61-85-93-59-104-99-96-6- <b>112P</b> -0
	2	C	0- <b>156P</b> -13-151-92-37-98-100-193-191-141-16-86-113-17-173-84-5-118- <b>60P</b> -166-89-0
	3	C	0- <b>156D</b> -147- <b>60D</b> -83-199-125-45-174-46-36-143-49-64-107-123-182-7-194-106-153-52- <b>146P</b> -0
2	C	0- <b>27D-167D</b> -127-190-88-148-62-159-11-175-19-168-47-124-48-82-8-114-18-0	
CMT 5Y	1	C	0-27-176-1-122-51-103-161-71-65-136-35-135-9-120-185-77-196-76- <b>28P</b> -0
	1	C	0-167-127- <b>190D</b> -88-148-62-159-90-126-63-181-32-131-160-128-66-188-20-30-70-101-162-69-132-0
	2	Y	0-153-106-194-7-82-48-47-36-143-49-64-107-175-11-108- <b>10D-189-10P</b> -31- <b>190P</b> -146-0
	1	C	0-52-182-123-19-168-124-46-174-8-114-125-45-199-83-18-166-89- <b>156D</b> -0
	1	C	0- <b>156P</b> -13-117-97-87-42-43-15-57-178-2-115-145-41-22-133-75-74-171-73-152-58-0
	2	C	0-53-105-180-198-110-155-25-55-165-130-54-134-163-24-29-121- <b>68P</b> -116-184- <b>28D</b> -0
CMT 9X	1	C	0-111-50-102-157-33-81-164-34-78-169-129-79-3-158- <b>68D</b> -150-80-177-109-12-138-154-0
	1	C	0-27-127-31-10-108-90-63-126-107-19-123- <b>146P</b> -0
	1	C	0-69-101-70-30-32-131-128-66-20-122- <b>1P</b> -0
	1	C	0-9-13-35-136-65-71-103-51- <b>1D</b> -132-0
	2	C	0- <b>111P</b> -50-102-33-81-120-34-78-129-79-3-77- <b>28D</b> -0
3	C	0- <b>111D</b> -76-116-68-121-29-24-134-150-80-12- <b>138P-28P</b> -0	

instance	t	S	Route
CMT 9X	1	C	0- <b>138D</b> -109-54-130-55-25-149-26-0
	1	C	0- <b>53D</b> -110-4-139-39-67-23-56-75-72-21-0
	1	C	0-105-40-73-74-133-22-41-145-115-2-58- <b>53P</b> -0
	2	Y	0- <b>96D-104D</b> -99- <b>104P</b> -6-0
CMT 10X	1	C	0- <b>96P</b> -59-93-85-61-17-45-125-83-60-118-89-0
	1	C	0- <b>146D</b> -52-106-7-82-48-124-46-8-114-18-0
	2	L	0- <b>28D</b> -76- <b>196D</b> -77-3-158-29-121-68-116-184- <b>28P</b> -0
	1	C	0- <b>156D</b> -112-183-96-99-104-93-85-61-173-5-147-0
	1	Y	0-94- <b>92D</b> -151- <b>92P</b> -98-91-16-86-141-191-193-59-6-0
	1	C	0- <b>156P</b> -13-87-172-42-142-43-15-57-144-137-0
	1	Y	0- <b>152D</b> -58- <b>152P</b> -0
	1	C	0- <b>53D</b> -180-198-110-4-155-25-55-165-130-54-179-149-0
	2	Y	0-138-154-12- <b>80P-150-80D</b> -134-24-163-177-109-195-26-105- <b>53P</b> -0
	2	C	0- <b>111P</b> -50-102-157-33-81-164-34-78-169-129-79- <b>196P</b> -0
CMT 10Y	1	C	0- <b>111D</b> -9-161-71-65-136-35-135-120-185-0
	2	O	0-166- <b>199P-125D</b> -45- <b>125P-199D</b> -18-0
	2	Y	0-147-5-84-173-17-113-61-93- <b>104D-99-104P-96D</b> -6-0
	1	C	0-183- <b>96P</b> -59-85-16-86-141-191-193-91-98-92-151-0
	1	C	0- <b>53P</b> -2-115-178-57-15-43-142-42-172-144-137-0
	1	C	0- <b>53D</b> -180-21-73-72-171-74-75-133-22-41-145-40-0
	1	C	0-105-149-179-110-4-155-25-55-165-130-54-109- <b>154D</b> -0
	2	Y	0-26-195-134-24-163-150-80-177-12- <b>138D-154P-138P</b> -0
	1	C	0-196-77-3-158-79-129-169-29-121-68-116-184- <b>28D</b> -0
	2	C	0- <b>111D</b> -50-102-157-33-81-120-164-34-78-185-76- <b>28P</b> -0
	1	C	0- <b>111P</b> -9-135-35-136-65-71-161-103-51-0
	1	C	0-27-176-1-122-128-66-188-20-30- <b>69D</b> -132-0
	1	C	0-167-108-126-63-90-32-131-160-70-101- <b>69P</b> -0
	1	C	0- <b>99D</b> -98-59-65-61-57-54-52-110-97-95-0
CMT 13X	1	C	0- <b>99P</b> -40-43-45-48-51-50-49-44-41-37-0
	1	C	0-109-26-32-35-36-34-31-28-23-20- <b>89D</b> -0
CMT 13Y	1	C	0- <b>89P</b> -114-17-22-24-27-33-30-25-19-16-0
	1	C	0- <b>105D</b> -106-103-73-76-77-78-80-79-68-98-99-0
	2	C	0- <b>94D-97D</b> -115-40-43-45-59-57-54-52-53-0
	3	C	0- <b>94P</b> -41-44-46-49-47-50-51-48-42- <b>39P-97P</b> -0
	5	C	0- <b>105P</b> -107-104-116-110- <b>39D</b> -38-37-109-114- <b>90D-91D-87P</b> -0
	1	C	0- <b>92D</b> -26-28-31-30-33-34-36-35-32-29-0
CMT 14X	1	C	0- <b>87D</b> -17-16-19-25-22-24-27-23-20-21-0
	3	C	0-86-85-84-5-4-3-6-118-18- <b>90P-91P-92P</b> -0
	1	C	0- <b>43D</b> -42-44-45-46-48-51-50-52-49-47-0
1	C	0-67-65-66-62-74-72-61-64-68-41- <b>43P</b> -0	

$t$ : number of split customers on the route, S: route shape, C: cycle, Y: cycle with co-located customer pairs, L: lasso, O: other, D: delivery service only, P: pickup service only.



**University of Kent**

<http://www.kent.ac.uk/kbs/research-information/index.htm>