

# A survey on pickup and delivery models

## Part II: Transportation between pickup and delivery locations

Sophie N. Parragh      Karl F. Doerner      Richard F. Hartl

August 30, 2006

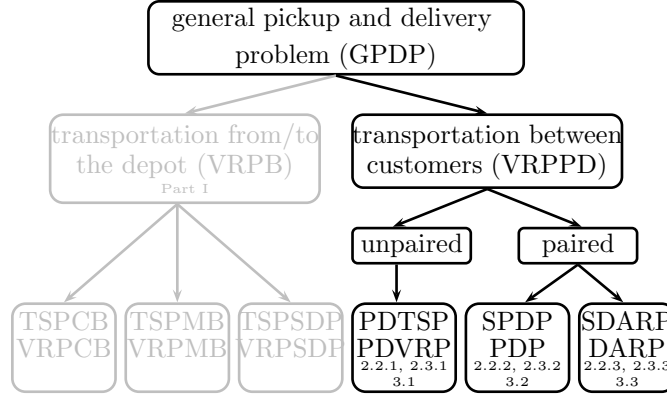
### Abstract

This paper is the second part of a comprehensive survey on pickup and delivery models. Basically, two problem classes can be distinguished. The first part dealt with the transportation of goods from the depot to linehaul customers and from backhaul customers to the depot. In this class three subtypes were considered, namely the Vehicle Routing Problem with Clustered Backhauls (VRPCB - all linehauls before backhauls), the Vehicle Routing Problem with Mixed linehauls and Backhauls (VRPMB - any sequence of linehauls and backhauls permitted), and the Vehicle Routing Problem with Simultaneous Delivery and Pickup (VRPSDP - customers can be linehaul and backhaul customers at the same time). The second part now considers all those problems where goods are transported between pickup and delivery locations, denoted as Vehicle Routing Problems with Pickups and Deliveries (VRPPD). These are the Pickup and Delivery VRP (PDVRP - unpaired pickup and delivery points), the classical Pickup and Delivery Problem (PDP - paired pickup and delivery points), and the Dial-A-Ride Problem (DARP - paired pickup and delivery points and user inconvenience constraints). A single as well as a multi vehicle mathematical problem formulation for all three VRPPD types are given, and the respective exact, heuristic, and metaheuristic solution methods are discussed.

## 1 Basic definitions

The aim of this paper is to present a comprehensive survey on pickup and delivery models and their variants. Part one of this article presented all problem types belonging to the Vehicle Routing Problem with Backhauls (VRPB) class. Furthermore, the motivation for this survey as well the demarcation of the problems discussed here from full-truck-load problems was given. This part covers all those problem types where goods are transported between pickup and delivery locations, referred to as Vehicle Routing Problems with Pickups and Deliveries (VRPPD).

Figure 1: Pickup and delivery models. The numbers indicated refer to the sections covering the respective problems.



The two pickup and delivery problem classes as well as their subclasses are depicted in Figure 1. The gray part was subject to discussion in part one. The numbers indicated in the boxes refer to the sections covering the respective problems. The first two indicators refer to the modeling part while the last refer to the sections on solution methods.

### 1.1 VRPB subclass definitions

The first category can be subdivided into three subclasses. In the first subclass the cluster of delivery customers has to be visited before the first pickup customer can be served. Delivery customers are also denoted as linehaul customers, pickup customers as backhaul customers, respectively. This subclass will be referred to as VRP with all linehauls before backhauls or VRP with Clustered Backhauls (VRPCB). The second subclass does not consider this restriction. Mixed visiting sequences are explicitly allowed, denoted as VRP with Mixed linehaul and Backhaul (VRPMB). The third subclass covers situations where every customer is associated with a linehaul as well as a backhaul quantity. It is imposed that every customer can only be visited exactly once, denoted as VRP with Simultaneous Delivery and Pickup (VRPSDP). For a more detailed description with references as well as for the different denotations used in the literature refer to the first part of this article.

### 1.2 VRPPD subclass definitions

The class we denote VRPPD refers to problems where goods are transported from pickup to delivery points. It can be further divided into two subclasses. The first subclass refers to situations where pickup and delivery points are unpaired. An identical good, such as money, is considered. Each unit picked up

can be used to fulfill the demand of every delivery customer. In the literature mostly the single vehicle case is tackled, denoted as Capacitated Traveling Salesman Problem with Pickups and Deliveries in Anily and Bramel (1999), One-Commodity Pickup and Delivery Traveling Salesman Problem (1-PDTSP) in Hernández-Pérez and Salazar-González (2003), and Traveling Salesman Problem with Pickup and Delivery in Hernández-Pérez and Salazar-González (2004a). Since also a multi vehicle application has been reported in the literature, see Dror et al. (1998), we will denote this problem class as Pickup and Delivery VRP (PDVRP) and Pickup and Delivery TSP (PDTSP), in the multi and in the single vehicle case, respectively.

The second VRPPD subclass comprises the classical Pickup and Delivery Problem (PDP) and the Dial-A-Ride Problem (DARP). Both types consider transportation requests, each associated with an origin and a destination, resulting in paired pickup and delivery points. PDP deal with the transportation of goods while DARP deal with people transportation. This difference is usually expressed in terms of additional constraints or objectives that explicitly refer to user (in)convenience. The single vehicle case of the PDP has also been referred to as Pickup-Delivery Traveling Salesman Problem by Kalantari et al. (1985) and the multi vehicle case as Pickup and Delivery Vehicle Routing Problem in Malca and Semet (2004) and Vehicle Routing Problem with Pickup and Delivery in Derigs and Döhmer (2006). However, a majority of the work published refers to this problem class as Pickup and Delivery Problem (PDP), compare, e.g. Dumas et al. (1991) or Van der Bruggen et al. (1993). Dial-a-ride problems are also mostly referred to as such. However, some authors, such as Toth and Vigo (1996), denote the same problem as the Handicapped persons Transportation Problem. The dynamic case is also referred to as Demand Responsive Transport, compare, e.g., Mageean and Nelson (2003). We denote the single vehicle case of the PDP as SPDP, the single vehicle case of the DARP as SDARP.

The remainder of this paper is organized as follows. First, in order to clearly define the different VRPPD types, a consistent mathematical problem formulation for the single and for the multi vehicle case is given. After that solution methods according to problem class are discussed and an overview over representative work in the respective domain is given in table form. In each of these sections the solution methods presented will be divided into exact, heuristic and metaheuristic approaches. Also related work will be mentioned.

## 2 Mathematical problem formulation

In the following section a consistent mathematical problem formulation will be proposed. First, the notation used throughout the paper is given. After that two basic problem formulations are introduced, one for the single, and one for the multi vehicle case, and extended to the *unpaired pickup and delivery problem*, the classical *pickup and delivery problem* and the *dial-a-ride problem*.

## 2.1 Notation

$P$ ...	set of backhauls or pickup nodes
$D$ ...	set of linehauls or delivery nodes
$n$ ...	number of pickup nodes, indexed $i = 1, \dots, n$
$\tilde{n}$ ...	number of delivery nodes, in case of paired pickups and deliveries $n = \tilde{n}$ , indexed $i = n + 1, \dots, n + \tilde{n}$
$q_i$ ...	load at vertex $i$ ; pickup nodes are associated with a positive value, delivery nodes with a negative value
$e_i$ ...	earliest time to begin service at vertex $i$
$l_i$ ...	latest time to begin service at vertex $i$
$d_i$ ...	service duration at vertex $i$
$L_i$ ...	maximum ride time of user $i$ ( $i = 1, \dots, n$ )
$c_{ij}^k$ ...	cost to traverse arc or edge $(i, j)$ with vehicle $k$
$t_{ij}$ ...	travel time from vertex $i$ to vertex $j$
$K$ ...	set of vehicles
$m$ ...	number of vehicles, indexed $k = 1, \dots, m$
$\bar{Q}^k$ ...	capacity of vehicle $k$
$T^k$ ...	maximum route duration of vehicle / route $k$

Note that this notation is valid for the symmetric as well as for the asymmetric case. In the symmetric case every arc  $(i, j)$  is equal to the arc  $(j, i)$  and could thus be replaced by one edge. Consequently less variables would be needed to formulate the symmetric case. Pickup and delivery problems are modeled on complete graphs  $G = (V, A)$  where  $V$  is the set of all vertices  $V = \{0, n + \tilde{n} + 1\} \cup P \cup D$ , where 0 and  $n + \tilde{n} + 1$  denote the depot, and  $A$  the set of all arcs.

During the optimization process some or all of the following decision variables are determined, depending on the problem considered.

$x_{ij}^k =$	$\begin{cases} 1, & \text{if arc } (i, j) \text{ is traversed by vehicle } k \\ 0, & \text{else} \end{cases}$
$Q_i^k$ ...	load of vehicle $k$ when arriving at vertex $i$
$B_i^k$ ...	beginning of service at vertex $i$

In the single vehicle problem formulation the subscript  $k$  can be omitted, resulting in the parameter notations  $c_{ij}, Q, T$  and the decision variables  $x_{ij}, Q_i, B_i$ .

## 2.2 Single vehicle pickup and delivery problem formulations

The single vehicle model for the different VRPPD is based on an open TSP formulation. Open refers to the fact that the resulting route is not a cycle but a path since the depot is denoted by two different indices.

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij} \quad (1)$$

subject to:

$$\sum_{i \in V} x_{ij} = 1 \quad \forall j \in V \setminus \{0\} \quad (2)$$

$$\sum_{j \in V} x_{ij} = 1 \quad \forall i \in V \setminus \{n + \tilde{n} + 1\} \quad (3)$$

$$x_{i0} = 0 \quad \forall i \in V \quad (4)$$

$$x_{n+\tilde{n}+1,j} = 0 \quad \forall j \in V \quad (5)$$

$$\sum_{i \in S} \sum_{j \notin S} x_{ij} \geq 1 \quad \forall S \subseteq V \setminus \{n + \tilde{n} + 1\}, S \neq \emptyset \quad (6)$$

$$x_{ij} \in \{0, 1\} \quad \forall i \in V, j \in V \quad (7)$$

The objective function (1) minimizes total routing cost. Constraint sets (2) and (3) ensure that each vertex is visited exactly once. That no arcs entering the origin depot 0 and no arcs leaving the destination depot  $n + \tilde{n} + 1$  are routed is guaranteed by (4) and (5), respectively. Constraints (6) present one of several possibilities to ensure route-connectivity. For other options refer to part one of this article.

### 2.2.1 The Pickup and Delivery Traveling Salesman Problem (PDTSP)

To model the *unpaired pickup and delivery case* the above problem statement, given in (1) – (7), is required. It is assumed that every unit picked up can be used to satisfy every delivery customer's demand. The same additional vehicle loading constraints as in the Traveling Salesman Problem with Mixed linehauls and Backhauls (TSPMB), described in part one of this article, are needed to accommodate the characteristics of this problem class,

$$Q_j \geq (Q_i + q_i)x_{ij} \quad \forall i \in V, j \in V, \quad (8a)$$

$$\max\{0, q_i\} \leq Q_i \leq \min\{\bar{Q}, \bar{Q} + q_i\} \quad \forall i \in V. \quad (8b)$$

The only difference to the TSPMB consists in the initial load of the vehicle which is free here, see Hernández-Pérez and Salazar-González (2004a).

### 2.2.2 The Single vehicle Pickup and Delivery Problem (SPDP)

The SPDP considers situations where pickup and delivery vertices are paired. This results in  $n + \tilde{n} = 2n$  customer locations. In the literature it is common to refer to such a vertex pair as a *request*, indexed by  $i = 1, \dots, n$  with  $i$  being the origin or pickup point and  $i + n$  the destination or delivery point. To ensure that every destination is only visited after its origin, in addition to (1) – (5),

(7), and (8), precedence constraints are needed which are usually modeled via time variables,

$$B_i \leq B_{i+n} \quad \forall i \in P, \quad (9)$$

$$x_{ij}(B_i + d_i + t_{ij}) \leq B_j \quad \forall i \in V, j \in V. \quad (10)$$

Constraint set (9) states that every origin is to be visited before its destination and (10) ensures that time variables are consistent with travel times. Note that (10) also guarantee that short cycles are avoided and therefore condition (6) is not needed.

### 2.2.3 The Single vehicle Dial-A-Ride Problem (SDARP)

A further extension of the PDP is the *dial-a-ride* problem (DARP). This pickup and delivery problem class deals with the transportation of people. Problems of this kind arise, e.g., in connection with the transportation of handicapped or elderly persons. Another possible application could be, however, the transportation of easily perishable goods, that also require maximum ride time constraints. In addition to the basic model, given in (1) – (5) and (7), and constraints (8) – (10), user inconvenience must be considered which can be handled by another term in the objective function or by additional constraints. Here we choose to formulate a constraint set concerning maximum user ride time,

$$B_{i+n} - (B_i + d_i) \leq L_i \quad \forall i \in P. \quad (11)$$

### 2.2.4 Additional Constraints

Constraints referring to the compliance with time windows,

$$e_i \leq B_i \leq l_i \quad \forall i \in V, \quad (12)$$

can be added to all of the above problems and almost always are in case of SPDP and SDARP.

## 2.3 Multi vehicle pickup and delivery problem formulations

The basic model for multi vehicle pickup and delivery problems is an adapted three index VRP formulation of the one proposed in Cordeau et al. (2001, p. 158f.) for the VRPTW.

$$\min \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}^k \quad (13)$$

subject to:

$$\sum_{k \in K} \sum_{j \in V} x_{ij}^k = 1 \quad \forall i \in V \setminus \{0, n + \tilde{n} + 1\} \quad (14)$$

$$\sum_{j \in V} x_{0j}^k = 1 \quad \forall k \in K \quad (15)$$

$$\sum_{i \in V} x_{i, n + \tilde{n} + 1}^k = 1 \quad \forall k \in K \quad (16)$$

$$\sum_{i \in V} x_{ij}^k - \sum_{i \in V} x_{ji}^k = 0 \quad \forall j \in V \setminus \{0, n + \tilde{n} + 1\}, k \in K \quad (17)$$

$$x_{ij}^k (B_i^k + d_i + t_{ij}) \leq B_j^k \quad \forall i \in V, j \in V, k \in K \quad (18)$$

$$Q_j^k \geq (Q_i^k + q_i) x_{ij}^k \quad \forall i \in V, j \in V, k \in K \quad (19)$$

$$\max \{0, q_i\} \leq Q_i^k \leq \min \{\bar{Q}^k, \bar{Q}^k + q_i\} \quad \forall i \in V, k \in K \quad (20)$$

$$x_{ij}^k \in \{0, 1\} \quad \forall i \in V, j \in V, k \in K \quad (21)$$

Total routing cost are minimized by the objective function, given in (13). Constraints (14) state that every vertex has to be served exactly once. Constraint sets (15) and (16) guarantee that every vehicle starts at the depot and returns to the depot at the end of its route. Flow conservation is ensured by (17). Time variables are introduced in constraint set (18) to ensure that no sub-tours occur and to facilitate the introduction of time related constraints later on. Constraint sets (19) and (20) guarantee that a vehicle's capacity is not violated throughout its tour. It should be noted that a model formulation in the above form requires the introduction of additional decision variables,  $Q_i^k$ , corresponding to the total load of vehicle  $k$  at vertex  $i$ , that is not needed in the basic VRP but essential for its extension to a pickup and delivery problem. However, it is sometimes used in VRP formulations to ensure route connectivity (see (18) in the above formulation).

### 2.3.1 The Vehicle Routing Problem with unpaired Pickups and Deliveries (VRPPD)

The special characteristic of this pickup and delivery problem class refers to the fact that every unit picked up can be used to satisfy every customer's demand. While the above formulation (13) – (21) was one of the possible VRP formulations, if all  $q_i$  have the same sign, it automatically becomes the correct formulation of the VRPPD when both pickups and deliveries can occur and no modifications are needed.

### 2.3.2 The Pickup and Delivery Problem (PDP)

Here every pickup point is associated with a delivery point and therefore  $n = \tilde{n}$ . In addition to (13) – (21), two more sets of constraints are needed. First, both

origin and destination of a request must be served by the same vehicle:

$$\sum_{j \in V} x_{ij}^k - \sum_{j \in V} x_{n+i,j}^k = 0 \quad \forall i \in P, k \in K \quad (22)$$

Furthermore, delivery can only occur after pickup, i.e.

$$B_i^k \leq B_{i+n}^k \quad \forall i \in P, k \in K. \quad (23)$$

### 2.3.3 The Dial-A-Ride Problem (DARP)

To extend the multi vehicle PDP to the multi vehicle DARP, again, as in the single vehicle case, constraints related to the minimization of user inconvenience need to be added. As in the single vehicle case we will restrict this requirement to adding maximum user ride time constraints to formulation (13) – (23):

$$B_{i+n}^k - (B_i^k + d_i) \leq L_i \quad \forall i \in P \quad (24)$$

### 2.3.4 Additional Constraints

Two more sets of constraints can be added to all of the above problem classes. These correspond to time window and maximum route duration restrictions,

$$e_i \leq B_i^k \leq l_i \quad \forall i \in V, k \in K, \quad (25)$$

$$B_{2n+1}^k - B_0^k \leq T^k \quad \forall k \in K. \quad (26)$$

The above formulations are extensively based on the DARP formulation presented in Cordeau (2006). Non linear constraints, given in (8a) and (10) in the single vehicle case, and (18) and (19) in the multi vehicle case, can easily be reformulated as a linear program by means of the usual big  $M$  formulation, compare Cordeau (2006).

## 3 Solution methods for pickup and delivery problems

In the following section solution methods as well as benchmark instances for the PDVRP, the PDP and the DARP are presented. Solution methods are classified according to exact, heuristic and metaheuristic approaches. A description of the benchmark instances used can be found at the very end of this section.

### 3.1 Unpaired pickups and deliveries

The PDVRP, i.e. the problem class where every good can be picked up and transported anywhere, did not receive as much attention in the literature as



the other problem classes. For this reason no table with representative work will be presented. However, the different solution methods proposed will be briefly discussed and the respective problemsizes of the instances solved will also be mentioned. Moreover, most of the literature is restricted to the PDTSP. Therefore, with the exception of Dror et al. (1998), all the solution methods presented are only applicable to the one vehicle case. To the authors' knowledge no metaheuristic approach for the PDTSP has been proposed until today.

### 3.1.1 Exact methods

The only exact method proposed for the problem at hand was introduced in Hernández-Pérez and Salazar-González (2003, 2004a). It is a *branch and cut* algorithm using a cutting plane approach. To speed up the algorithm construction and improvement heuristics are used to generate the initial solution. The construction heuristic applied is an adaptation of the nearest insertion algorithm which is improved by 2-opt and 3-opt exchanges, Lin (1965). The test instances solved are adaptations of the ones used in Mosheiov (1994) and Gendreau et al. (1999), containing up to 75 customers.

### 3.1.2 Heuristics

A special case of the problem considered is tackled in Chalasani and Motwani (1999). In this case the number of goods to be picked up is equal to the goods to be delivered and the demand (supply) at every delivery (pickup) location is equal to one. This problem is an extension of the swapping problem where the vehicle's capacity is also set to one. Chalasani and Motwani propose an *approximation* algorithm with a worst case bound of 9.5. They use Christofides' heuristic to construct a traveling salesman tour, one containing only pickup locations, and one containing only delivery locations. These two tours are then combined by means of decomposition and matching. In Anily and Bramel (1999) a polynomial time iterated tour matching algorithm for the same problem is proposed.

Dror et al. (1998) propose a heuristic algorithm for the application of the PDVRP to the redistribution of self-service cars that is related to Dijkstra's algorithm. Also other solution approaches are briefly discussed.

Two heuristic methods can be found in Hernández-Pérez and Salazar-González (2004b). The first algorithm is of the *construction-improvement* type, using a greedy construction procedure that is improved by 2-opt and 3-opt exchanges. The second heuristic is based on *incomplete optimization*. The branch and cut procedure, described in Hernández-Pérez and Salazar-González (2004a), with restrictions on the search space is applied. Random instances with up to 500 customers were solved.

An *approximation* algorithm on a tree graph was proposed by Lim et al. (2005). It is based on a recurrent construction process and has a worst case bound of 2.

### 3.2 The pickup and delivery problem

Solution methods for the classical pickup and delivery problem (PDP), where every transportation request is associated with a pickup and a delivery point, are presented in this section. Lokin (1978) was the first to discuss the incorporation of precedence constraints into the traditional TSP which are needed to formulate the PDP. The first attempt of generalize the PDP in unified notation was proposed by Savelsbergh and Sol (1995), covering all possible versions of the PDP including the dial-a-ride problem. They also provide a brief survey on existing solution methods until 1995. Mitrovic-Minic (1998) present a survey on the PDP with Time Windows (PDPTW). An early survey on vehicle routing problems already including the PDP was given by Desrochers et al. (1988). Cordeau et al. (2004) provide a survey on demand responsive transport covering PDP and DARP. Dynamic routing is dealt with in Ghiani et al. (2003), Psaraftis (1988). Further surveys on solution methods can be found in Assad (1988), Cordeau et al. (2004), Desaulniers et al. (2001), Desrochers et al. (1988).

In the following paragraphs the various solution techniques proposed for the static as well as the dynamic PDP are summarized according to exact, heuristic and metaheuristic approaches. A representative overview of solution methods for static PDP is given in Table 1, representative solution procedures designed for the dynamic case can be found in Table 2.

#### 3.2.1 Exact methods

All the methods discussed in the following paragraph belong the class of static PDP.

The first exact solution method applicable to both the single as well as the multi vehicle case dates back to Kalantari et al. (1985). The *branch and bound* algorithm proposed is an extension of the one developed by Little et al. (1963). Instances with up to 18 requests could be solved to optimality.

Desrosiers and Dumas (1988) study the constraint shortest path problem occurring when column generation is used to solve the PDP. A dynamic programming algorithm as well as heuristic procedures to speed up the algorithm are proposed to solve the problem.

A *column generation* approach was proposed by Dumas et al. (1991) to tackle the multi vehicle PDP considering heterogeneous vehicles, time windows as well as multiple depots. Exact solutions to test instances with up to 30 requests are reported.

Ruland and Rodin (1997) propose a *branch and cut* algorithm to solve the single vehicle PDP. It includes cutting plane generation in the bounding phase as well as a greedy route construction procedure to generate the first upper bound. The test instances solved comprise up to 15 requests.

An exact algorithm for the multi vehicle PDPTW is proposed in Sigurd et al. (2004). Via *Dantzig-Wolfe decomposition* a master problem and subproblems (one per vehicle) are generated. The master problem is solved by delayed column generation and the pricing problems, through reformulation as shortest path

Table 1: static PDP

Reference	Type	Obj.	Con.	Algorithm	Problemsize
The single vehicle case					
Sexton and Choi (1986)	-	min. RC, TWV	soft TW	heuristic algorithm using Bender's decomposition	up to 17 requests
Van der Bruggen et al. (1993)	-	min. RC	TW	variable depth-search based algorithm	up to 38 requests
Ruland and Rodin (1997)	-	min. RC	-	branch and cut algorithm, using cutting plane generation, greedy route construction	up to 15 requests
Renaud et al. (2000)	-	min. RC	-	two phase algorithm. (1) double insertion (2) deletion and re-insertion (4-opt*)	up to 220 requests
Renaud et al. (2002)	-	min. RC	-	7 perturbation heuristics (instance, algorithmic and solution perturbation)	up to 220 requests
Carrabs et al. (2005)	-	min. RC	LIFO	variable neighborhood search	adapted TSPLIB instances
The multi vehicle case					
Kalantari et al. (1985)	-	min. RC	-	extension of branch and bound algorithm of Little et al. (1963)	up to 18 requests
Dumas et al. (1991)	HF, MD	min. RC	TW	exact algorithm using column generation, constrained shortest path subproblems	up to 55 requests
Pankratz (2005b)	-	min. NV, RC	TW	grouping genetic algorithm	NB00, LL01
Lu and Dessouky (2006)	-	min. RC	TW	construction heuristic based on distance increase, TW slack reduction, visual attractiveness	LL01
Ropke and Pisinger (2005)	MD, S	min. RC	TW	adaptive large neighborhood search	LL01+
Ropke et al. (2005)	-	min. RC	TW	2 branch and cut algorithms	up to 96 requests
Bent and Van Hentenryck (2006)	-	min. NV, RC	TW	hybrid algorithm. (1) simulated annealing (2) large neighborhood search	LL01+
Cordeau et al. (2006)	-	min. RC	TW, LIFO	branch and bound algorithms	up to 51 requests

Con. = Constraints, Obj. = Objective(s), RC = Routing Cost, TWV = Time Window Violation, NV = Number of Vehicles, TW = Time Windows, MD = Multi depot, HF = Heterogeneous Fleet, S = Service Time; The respective benchmark instances are described in Section 3.4.

Table 2: dynamic PDP

Reference	Type	Obj.	Con.	Algorithm	Problemsize
The multi vehicle case					
Savelsbergh and Sol (1998)	HF, MD	min. RC	TW	DRIVE. branch and price algorithm	simulated real life case study
Gendreau et al. (1998)	-	min. sum of RC, lateness, overtime	TW	parallelized tabu search heuristic, ejection chain, neighborhood, adaptive memory	simulation (7.5 (4) hrs., 20 (10) veh., 23 (33) requests/hr)
Mitrovic-Minic and Laporte (2004)	-	min. RC	TW	2 phase algorithm. (1) cheapest insertion, improved by TS (2) scheduling (different waiting strategies)	up to 1000 requests
Mitrovic-Minic et al. (2004)	MD	min. RC	TW	double horizon based heuristic using advanced dyn. waiting	up to 1000 requests
Fabri and Recht (2006)	-	min. RC	TW	adaptation of Caramia et al. (2002) algorithm, allowing for waiting times	up to 1000 requests (1 per minute)

Con. = Constraints, Obj. = Objective(s), RC = Routing Cost, TW = Time Windows, MD = Multi depot, HF = Heterogeneous Fleet, TS = tabu search, veh. = vehicle

problems, via dynamic programming. Feasibility is achieved using a branch and bound algorithm. Instances with up to 108 requests could be solved optimally.

Lu and Dessouky (2004) propose another exact solution algorithm. Time window as well as capacity constraints are considered to be soft. A *branch and bound* approach is chosen and four classes of valid inequalities are proposed. Optimal solutions to problem instances with up to 25 requests are reported.

Two *branch and cut* algorithms departing from different PDPTW formulations are studied in Ropke et al. (2005). New valid inequalities to strengthen the proposed formulations are discussed. The results reported consider up to 96 customers and 8 vehicles.

A *branch and bound* algorithm for the single vehicle PDPTW with LIFO loading is presented in Cordeau et al. (2006). They propose three mathematical problem formulations and various valid inequalities based on existing inequalities for the basic PDP as well as new problem specific ones. The instances solved contain up to 51 transportation requests.

### 3.2.2 Heuristics

The first heuristic solution procedures for *static PDP* were developed in the 1980s. Cullen et al. (1981) develop an interactive heuristic algorithm for the static PDP that follows the cluster first route second approach. It is based

on a set partitioning formulation solved by means of column generation. The location-allocation subproblem is only solved approximately. Cordeau et al. (2004) classify this approach as an early DARP. However, according to our classification scheme it belongs to the group of PDP, since no user related constraints are considered.

Sexton and Choi (1986) used *Bender's decomposition* procedure to solve the static single vehicle PDP. To save CPU time the initial solutions were constructed using a space-time heuristic. A route improvement phase was also implemented. As soft time windows were considered, the objective function takes into account the total operating time as well as time window violation penalties. Sexton and Choi (1986) solved problem instances with up to 17 transportation requests.

A *construction-improvement* heuristic algorithm for the static single vehicle PDP is discussed in Van der Bruggen et al. (1993). First a feasible initial solution is constructed. Then this solution is improved by variable depth exchange procedures maintaining feasibility at all times. The instances solved contain up to 55 transportation requests. Renaud et al. (2000) also propose a construction-improvement algorithm for the same problem using a double insertion construction heuristic improved by deletion and re-insertion (4-opt\*, Renaud et al. (1996)). The problem instances tackled contain up to 220 requests.

A new local search *improvement* heuristic was proposed by Healy and Moll (1995) for the single vehicle PDP. Healy and Moll (1995) refer to their method as an improvement procedure for the DARP, however, neither time windows nor user inconvenience are considered. For this reason their publication is dealt with in this section. The presented algorithm is based on the classical 2-opt operator and consists of two alternating phases referred to as optimizing and sacrificing. During the optimizing phase the primary objective of minimum tour length is considered while in the sacrificing phase a secondary metric is used, i.e. the size of the feasible neighborhood of the respective solution. Healy and Moll (1995) report results for problem instances with up to 100 instances.

Renaud et al. (2002) present seven different *perturbation* heuristics to generate near optimal solutions for the static single vehicle PDP. In all seven implementations, first, an initial solution is computed which is improved by a 4-opt\*\* heuristic, an adaptation of the 4-opt\* heuristic proposed by Renaud et al. (1996). Then, a perturbation scheme is applied (instance, algorithmic or solution perturbation) followed by a post-optimization phase. The last two steps are repeated until a stopping criterion is met. The same instances as in Renaud et al. (2000) (up to 220 requests) were solved.

Lim et al. (2002) propose a *squeaky wheel optimization* with local search to solve the multi vehicle case. Solutions to the test instances of Li and Lim (2001) are reported.

Xu et al. (2003) propose a *column generation based* heuristic algorithm. They consider the static multi vehicle PDP with several additional constraints, such as multiple time windows at pickup and delivery locations, loading restrictions, compatibility of goods and vehicles as well as driver working hours. By column generation the master problem can be solved using a commercial LP

solver. However, the resulting subproblems have to be solved heuristically. The problem instances tackled contain up to 500 requests.

Lu and Dessouky (2006) propose a *construction* heuristic for the static multi vehicle PDPTW. The algorithm proposed does not only incorporate distance increase into the evaluation criterion but also time window slack reduction as well as visual attractiveness (referred to as crossing length percentage). The instances solved are those of Li and Lim (2001).

A variation of the classical PDP is presented in Shang and Cuff (1996). The problem they tackled does not require a good (here medical records) to be pickup and delivered by the same vehicle. Transfers between vehicles are allowed. The objective considered is the minimization of unsatisfied customers. The proposed heuristic uses *concurrent insertion* and a mini-clustering algorithm. A problem where 167 items had to be transported was tackled. A similar problem is studied in Thangiah and Awan (2006). In contrast to Shang and Cuff (1996) time windows are considered to be hard. The solution heuristic proposed by Shang and Cuff (1996) is adapted and enhanced by an improvement step based on local search. Moreover split deliveries, i.e. two vehicles can serve one transportation request, are explicitly allowed. Split deliveries have also been studied in the context of the traditional VRP, compare Archetti et al. (2006a,b).

The first heuristic procedure determined for the *dynamic PDP* was proposed by Savelsbergh and Sol (1998). Their solution methodology called DRIVE (Dynamic Routing of Independent VEHicles) incorporates a branch and price algorithm based on a set partitioning problem formulation that generates approximate solutions via *incomplete optimization*. The problem tackled consists of a 10 days real life simulation. Up to 354 active requests were considered at the various re-optimization runs.

A stochastic and dynamic SPDP is discussed in Swihart and Papstavrou (1999). The objective minimized consists of the expected time a request remains in the system. They test three routing policies, a sectoring, a nearest neighbor and a stacker crane policy. The stacker crane policy refers to grouping arriving demands into contiguous sets of equal size and serving them according to the first come first served rule. Lower bounds under light and heavy traffic conditions are computed.

An *insertion* based heuristic procedure for the dynamic multi vehicle PDP is proposed in Popken (2006). The heuristic is combined with different types of order circuitry control in order to increase the efficient utilization of the vehicle's capacity. Results for test instances with up to 2500 initial orders are reported.

Fabri and Recht (2006) present an adaptation of the heuristic algorithm initially designed for the dynamic DARP by Caramia et al. (2002). They explicitly allow waiting times. To enhance the procedure an additional local search phase is introduced. This phase is initiated whenever a new request has been inserted and ends when the next request comes in. Fabri and Recht (2006) report solutions to instances with up to 1000 requests, arriving at an average rate of 1 request per minute.

### 3.2.3 Metaheuristics

Early research on *dynamic PDP* was conducted by Shen et al. (1995) and Potvin et al. (1995). Both articles focus on neural networks with learning capabilities to support vehicle dispatchers in real-time. In Potvin et al. (1995) the learning techniques based on neural networks is compared to a linear programming based method. Both articles report results for a real life data set.

The first neighborhood based metaheuristic solution method was proposed by Gendreau et al. (1998). They propose a *tabu search* heuristic for the *dynamic PDPTW* that uses an ejection chain neighborhood and an adaptive memory. A lateness criterion is incorporated into the objective function. To speed up the optimization procedure a parallel implementation was conducted. The resulting program was tested on simulations over 7.5 hours with 20 vehicles and 23 requests per hour, and over 4 hours with 10 vehicles and 33 requests per hour. Another tabu search algorithm for the same problem was proposed by Malca and Semet (2004). The neighborhood used is of the request to vehicle assignment type. In order to speed up the search an elimination matrix that memorizes the compatibility of two requests is used. Thus only promising moves are considered. The proposed procedure was tested on some adapted instances of Li and Lim (2001).

A two-phase solution procedure using a *tabu search* algorithm for the dynamic multi vehicle PDPTW is presented in Mitrovic-Minic and Laporte (2004). In the first phase an initial solution is constructed via cheapest insertion. Then a tabu search algorithm is run to improve on the initial solution. In the second phase different waiting strategies are used to schedule the requests. The waiting strategies tried are referred to as drive first, wait first, dynamic waiting and advanced dynamic waiting. They differ regarding the vehicle's location when waiting occurs. When applying the drive first strategy, the vehicle leaves every vertex as early as possible. If it arrives too early at the subsequent stop it waits there until service is possible. When applying the wait first strategy, the vehicle leaves every vertex as late as possible w.r.t. time windows of subsequent vertices. Dynamic waiting refers to a strategy where customers are clustered according to time windows. The vehicle waits as long as possible before moving on to the first customer of the next cluster. Advanced dynamic waiting refers to a strategy where waiting time before visiting the first cluster depends on the latest possible time to begin service at the last cluster, without violating time windows at intermediate clusters. Mitrovic-Minic and Laporte (2004) report solutions to problem instances with up to 1000 requests. The advanced dynamic waiting strategy is also used in Mitrovic-Minic et al. (2004). They propose a double horizon based heuristic. The routing part is solved by means of a construction heuristic improved by tabu search. Scheduling is conducted according to advanced dynamic waiting. Routes are segmented. The first segment corresponds to the short term horizon, the remainder of the route to the long term horizon. Also the case of several depots is considered. Again instances with up to 1000 requests were solved.

A population based metaheuristic approach for the dynamic PDP was pro-

posed by Jih and Hsu (1999). They use a hybrid *genetic algorithm* to solve the dynamic single vehicle PDP. Hybrid, as the genetic algorithm was combined with a dynamic programming algorithm. The test data sets consisted of up to 50 requests. In Pankratz (2005a) a grouping genetic algorithm for the static PDP is embedded in a rolling horizon framework in order to solve the dynamic PDP with time windows. It is tested on data sets with different degrees of dynamism.

Gutenschwager et al. (2004) compare a steepest descent, a reactive tabu search and a simulated annealing algorithm to solve the dynamic PDP on an electric monorail system.

A reactive *tabu search* for the *static multi vehicle PDP* is presented in Nanry and Barnes (2000). Three neighborhoods were defined, a single pair insertion, a swapping pairs and an intra-route insertion neighborhood. Modified Solomon (1987) instances with up to 100 requests are solved. Lau and Liang (2001) also use a tabu search algorithm to solve the static multi vehicle PDP. The initial solution is constructed via partitioned insertion. Then the tabu search algorithm is applied using adapted versions of the neighborhoods proposed in Nanry and Barnes (2000). Again, adaptations of the Solomon (1987) instances are solved.

A *tabu search* as well as a probabilistic tabu search algorithm for the SPDP were proposed in Landrieu et al. (2001). In the probabilistic version at every iteration the  $N$  best non tabu moves are recorded and one of them is chosen via a probability assignment process. The test instances tackled consist of up to 40 requests.

Li and Lim (2001) develop a *tabu embedded simulated annealing* approach to solve the static PDP. Pickup and delivery pair swap neighborhoods are defined. These are based on a shift, an exchange and a rearrange operator. The first two serve as the neighborhoods searched by the metaheuristic, the third is used for postoptimization purposes. They test their algorithm on newly constructed benchmark instances based on the VRP instances of Solomon (1987), and on the test instances proposed by Nanry and Barnes (2000).

Caricato et al. (2003) propose a parallel *tabu search* algorithm to solve a variation of the static PDP. They consider the multi vehicle PDP under track contention minimizing longest route duration. Instances with up to 50 requests were solved.

Another extension of the static PDP is studied by Ambrosini et al. (2004). The problem tackled includes a further constraint regarding rear loading, i.e. items can only be delivered in a LIFO matter. Ambrosini et al. (2004) propose a *GRASP* to solve this problem. A *variable neighborhood search* for the same problem is presented in Carrabs et al. (2005). Eight different construction techniques are used to generate an initial solution. The neighborhoods are defined by couple exchange, block exchange, relocate-block, 2-opt-L and multi-relocate operators. The proposed metaheuristic was tried on adapted instances of the TSPLIB.

Pisinger and Ropke (2005) present an adaptive *large neighborhood search* algorithm for the static PDPTW. Multiple depots as well as the existence of



service times can be handled by the approach at hand. In Pisinger and Ropke (2005) the proposed method is used to solve VRP instances by transforming VRP into rich PDPTW. In Ropke and Pisinger (2005) their heuristic algorithm is used to solve PDPTW instances. Solutions for the extended test instances of Li and Lim (2001), containing up to 500 transportation requests, are reported. The same metaheuristic was also used to solve VRPB instances, Ropke and Pisinger (2006). Therefore we refer to the section on VRPCB for a more detailed description.

A *genetic algorithm* is used to solve the static PDP by Jung and Haghani (2000). The instances tackled contain up to 30 requests. They also report exact solutions to instance with up to 6 requests. A variation of the PDP is tackled in Schönberger et al. (2003) with a hybrid genetic algorithm. The variation refers to the possibility to reject some of the requests if they do not contribute to the goal of profit maximization. The instances solved are adaptations of those of Nanry and Barnes (2000). A genetic algorithm combined with local search to solve the static multi vehicle PDPTW is presented in Creput et al. (2004). The algorithm is tried on the problem instances proposed by Li and Lim (2001).

Pankratz (2005b) propose a *grouping genetic algorithm* for the PDP. The grouping genetic algorithm differs from traditional genetic algorithms in that a group-oriented genetic encoding is used. The encoding used by Pankratz (2005b) corresponds to the cluster of requests forming a route. The routing aspect not comprised in the encoding is added while decoding the chromosome. Pankratz (2005b) report solutions to the test instances of Nanry and Barnes (2000) and Li and Lim (2001).

Yet another solution procedure was developed by Derigs and Döhmer (2006). They present a competitive *indirect search* method for the multi vehicle case that is tested on the instances of Li and Lim (2001).

A *two-stage hybrid algorithm* for the static PDPTW has recently been presented by Bent and Van Hentenryck (2006). The first phase uses simulated annealing to decrease the number of vehicles needed. The second phase consists of a large neighborhood search algorithm in order to reduce total travel cost. Solutions to the benchmark instances of Li and Lim (2001) are reported.

### 3.2.4 Related work

Pickup and delivery problems do not only arise in the context of vehicle routing but also in ocean borne transportation. Christiansen and Nygreen (1998a) propose a combined PDPTW and multi inventory model arising in the context of ship routing. It is solved by means of a branch and bound embedded iterated solution procedure based on Dantzig-Wolfe decomposition. Similar solution approaches are presented in Christiansen and Nygreen (1998b), Christiansen (1999). A column generation approach is presented in Christiansen and Nygreen (2005), assuming uncertainties in sailing times and considering inventory constraints to be soft. An optimal solution method for the traditional PDPTW in the context of ship routing is reported in Christiansen and Fagerholt (2002). Brønmo et al. (2005) propose a multi-start local search heuristic. A relaxed

version of the multi-ship PDPTW, considering soft time windows, is tackled in Fagerholt (2001). Fagerholt and Christiansen (2000a) also study a combined ship scheduling and allocation problem. Optimal solutions are computed for several real life cases. The subproblem, a TSPTW with allocation and precedence constraints, of the combined problem is studied by Fagerholt and Christiansen (2000b). For an extensive survey on ship routing problems refer to Christiansen et al. (2004).

An interesting extension of the PDPTW was proposed by Recker (1995), namely the household activity pattern problem. It involves ridesharing as well as vehicle-switching options. Its objective refers to the minimization of household travel disutility. Recker solved the problem defined by means of a genetic algorithm.

Research dedicated to polyhedral analysis is presented in Dumitrescu (2005) w.r.t. the SPDP. They also develop and discuss new valid inequalities that are important in the context of exact solution procedures.

Gambardella and Dorigo (2000) discuss a problem that is related to the PDP, namely the sequential ordering problem. Its objective is to determine a minimum weight Hamiltonian path on a directed graph, with weights on arcs and vertices, respecting precedence constraints between vertices. In contrast to the PDPTW one vertex can have multiple predecessors. Gambardella and Dorigo (2000) propose an ant colony optimization based approach to solve the problem. Solution methods for the same problem are also discussed, e.g., in Escudero (1988), Ascheuer et al. (1993).

### 3.3 The dial-a-ride problem

The dial-a-ride problem class has received considerable attention in the literature. The first publications in this area date back to the late 1960s and 1970s, compare Rebibo (1974), Wilson and Weissberg (1967), Wilson et al. (1971), Wilson and Colvin (1977). Surveys on solution methods can be found in Cordeau and Laporte (2003a), Cordeau et al. (2004), Gendreau and Potvin (1998).

A representative overview of solution methods proposed for the static and for the dynamic DARP can be found in Tables 3 and 4 respectively. In the following paragraphs the different solution methods for both problem types are presented. They are classified according to exact, heuristic and metaheuristic approaches.

#### 3.3.1 Exact methods

An early exact *dynamic programming* algorithm for the single vehicle DARP was proposed by Psaraftis (1980). Instances with up to 9 customers could be solved. Its adaptation to the dynamic case including an additional constraint regarding the maximum position shift of the customers is also discussed. Transportation requests are ordered according to time of call. Up to 10 customer requests can be handled by the dynamic solution method. In Psaraftis (1983b) a modified

Table 3: static DARP

Reference	Type	Obj.	Con.	Algorithm	Problemsize
The single vehicle case					
Psaraftis (1980)	-	min. RC, CI	service a.s.a.p., PoC	exact algorithm based on dynamic programming	up to 9 customers
Psaraftis (1983b)	-	min. RC, CI	TW	modification of algorithm in Psaraftis (1980). backward instead of forward recursion	up to 9 customers
Sexton and Bodin (1985a,b)	-	min. CI	DDT	routing and scheduling algorithm based on Bender's decomposition	up to 20 customers
Desrosiers et al. (1986)	-	min. RC	TW	forward dynamic programming	up to 40 customers
The multi vehicle case					
Jaw et al. (1986)	-	min. RC, CI	TW, RT	sequential feasible insertion algorithm	up to 2617 customers
Ioachim et al. (1995)	HF, HP, MD, S	min. NV, RC	TW	mini-clustering algorithm using column generation	up to 2545 customers
Toth and Vigo (1997)	HF, HP, MD	min. RC	TW, RT	parallel insertion algorithm, tabu thresholding	up to 312 customers
Borndörfer et al. (1997)	HF, MD, S	min. RC	TW, (RT), driver shift length	cluster first route second. set partitioning model solved (approximately) by branch and bound	up to 1771 customers
Cordeau and Laporte (2003b)	-	min. RC	TW, RT, TL	tabu search algorithm	CL03, real life with up to 295 customers
Diana and Dessouky (2004)	-	min. RC, excess RT, idle times	TW, RT	parallel regret insertion heuristic	up to 1000 customers
Cordeau (2006)	-	min. RC	TW, RT, TL	branch and bound algorithm	Cor06
Ropke et al. (2005)	-	min. RC	TW, RT, TL	2 branch and bound algorithms	Cor06
Bergvinsdottir et al. (2006)	MD	min. RC, CI	TW, RT, TL	genetic algorithm	CL03

Con. = Constraints, Obj. = Objective(s), RC = Routing Cost, CI = Customer Inconvenience, NV = Number of Vehicles, TL = Tour Length, TW = Time Windows, RT = Ride Time, MD = Multi depot, HF = Heterogeneous Fleet, HP = Heterogeneous Passengers, S = Service time, DDT = Desired Delivery Time, PoC = Position of Customer; The respective benchmark instances are described in Section 3.4.

Table 4: dynamic DARP

Reference	Type	Obj.	Con.	Algorithm	Problemsize
The single vehicle case					
Psaraftis (1980)	-	min. RC, CI	service a.s.a.p., PoC	exact algorithm keeping track of position shifts (dynamic case)	up to 10 customers (dynamic case)
The multi vehicle case					
Stein (1978b)	transfers	min. PFT	-	cluster first route second extended to dyn. case	-
Madsen et al. (1995)	HF, HP, S	min. RC, NV, TWT, deviation from promised service	TW, NV, RD, RT	REBUS. insertion based algorithm	up to 300 customers
Attanasio et al. (2004)	-	max. nbr of cus- tomers served	TW, RT, RD	parallelized version of sequential tabu search algorithm of Cordeau and Laporte (2003b)	up to 144 customers
Coslovich et al. (2005)	-	max. nbr of cus- tomers served	TW, RT	2-phase insertion algorithm	up to 50 customers

Con. = Constraints, Obj. = Objective(s), RC = Routing Cost, PFT = Passenger Flow Time, CI = Customer Inconvenience, TWT = Total Waiting Time, NV = Number of Vehicles, TW = Time Windows, RT = Ride Time, RD = Route Duration, HF = Heterogeneous Fleet, HP = Heterogeneous Passengers, S = Service time, PoC = Position of Customer

version of the above algorithm is presented. Instead of backward recursion forward recursion is used and also time windows are considered.

Kikuchi (1984) develop a balanced LP transportation problem for the static DARP minimizing empty vehicle travel as well as idle times and thus fleet size. In a preprocessing step the service area is divided into zones and the time horizon into several time periods. Every request is classified according to origin and destination zone as well as departure and arrival time period. An example with 4 zones is presented.

A forward *dynamic programming* algorithm for the static SDARP was introduced by Desrosiers et al. (1986). Possible states are reduced by eliminating those that are incompatible w.r.t. vehicle capacity, precedence and time window restrictions. User inconvenience w.r.t. ride times is incorporated into time window construction, resulting in tight time windows on both origin and destination of the transportation request. The test instances solved contained up to 40 customer transportation requests.

Cordeau (2006) proposes a *branch and cut* algorithm for the static DARP. The algorithm is based on a new mixed-integer problem formulation and uses

new valid inequalities as well as previously developed ones for the PDP and the VRP. Problem instances with up to 32 customers could be solved to optimality. Another two branch and cut algorithms are presented in Ropke et al. (2005). The same instances as in Cordeau (2006) were solved.

### 3.3.2 Heuristics

An early heuristic method for the *static multi vehicle DARP* with transfer possibilities in the context of bus routing is presented in Stein (1978a,b). The first step consists in clustering the relevant region into as many subregions as vehicles and assigning each subregion to a vehicle. In a second step traveling salesman tours are constructed within each subregion. This algorithm is also applied to the dynamic case.

Psaraftis (1983a) propose a *minimum spanning tree* heuristic for the SDARP. The first customer is randomly chosen. Feasible routes are improved by local interchanges. Instances with up to 50 customers were solved. Adaptations of the 2-opt and 3-opt *improvement* heuristics in combination with a breadth-first and a depth-first search are discussed in Psaraftis (1983c). Solutions to instances with up to 30 customers are reported.

A heuristic routing and scheduling algorithm using *Bender's decomposition* is proposed in Sexton and Bodin (1985a,b). The scheduling problem can be solved optimally while the routing problem is solved by means of a heuristic algorithm. Problem instances with up to 20 customers were solved. Bodin and Sexton (1986) discuss another heuristic algorithm that iterates between routing and scheduling phases. Solutions to instances with about 85 customers are reported.

Jaw et al. (1986) propose a *sequential insertion* procedure for the static DARP. The heuristic method suggested is applied to instances with up to 2617 customers. An adapted version of this algorithm is discussed in Alfa (1986) considering soft ride time constraints and focusing on the maximization of the user respond rate. Moreover, vehicles with heterogeneous capacities are considered. The instances solved consist of up to 49 customer requests. Psaraftis (1986) also study the algorithm of Jaw et al. (1986) and compare it with a grouping-clustering-routing algorithm developed by the same authors. Heuristic algorithms for the single vehicle case are also discussed in Kubo and Kasugai (1990).

Kikuchi and Rhee (1989) develop another *insertion* procedure for the static DARP maximizing the number of trips served per vehicle, considering one vehicle at a time. After feasibility of insertion is checked for each trip, the maximum number of trips, using a tree structured search technique, is determined. The procedure is applied to instances with up to 200 requests.

A *parallel insertion* algorithm for the multi vehicle DARP is developed and tested by Roy et al. (1985b,a). Toth and Vigo (1996) also propose a parallel insertion algorithm for the static DARP improved by inter as well as intra-route improvement procedures such as trip insertion, exchange, double insertion and moves. Up to 312 customers per problem instance are considered. An adapted

parallel insertion procedure is presented in Fu (2002a) and applied to instances with up to 2800 transportation requests. The same authors propose a simulation environment to test future solution methods for the dynamic DARP in Fu (2002b). Wong and Bell (2006) propose another parallel insertion algorithm for the DARP with heterogeneous passengers as well as vehicles. Trip insertion, as proposed in Toth and Vigo (1996), is applied to improve on the constructed solution. The algorithm was applied to a data set containing 150 transportation requests.

A *regret insertion* algorithm for the static DARP was proposed by Diana and Dessouky (2004). First all requests are ranked according to ascending pickup times allowing some swaps in this order, giving preference to requests that might be difficult to insert later on w.r.t. their spatial location. The first  $m$  requests are used as seed customers. All the remaining requests are inserted following a regret insertion strategy, as described in Potvin and Rousseau (1993). Instances with up to 1000 transportation requests were solved. The regret insertion based process is also subject to analysis in a study by Diana (2004) to determine why the performance of this heuristic is superior to that of other insertion rules.

Xiang et al. (2006) present a *construction-improvement* algorithm for the static DARP that consists of a pre-processing, a construction, an improvement and an intensification phase. Heterogeneous vehicles, drivers as well as customers are considered. Xiang et al. (2006) report solutions to instances with up to 2000 requests.

Desrosiers et al. (1988), Dumas et al. (1989) propose a *cluster first route second* algorithm for the static multi vehicle DARP. In a first step mini-clusters of nearby customers are constructed. These segments are then optimally combined and routed using a column generation algorithm. Solutions for instances with up to 200 customers are reported. To solve larger instances, with up to 880 customers, the day was divided into time slices. Desrosiers et al. (1991) exclusively focused on improving on the mini-clustering algorithm used in Desrosiers et al. (1988). They propose a sequential insertion heuristic. When used together with the routing algorithm of Desrosiers et al. (1988) instances with up to 2411 requests could be solved. Another optimization based mini-clustering algorithm was presented by Ioachim et al. (1995). It uses column generation to obtain mini-clusters and an enhanced initialization procedure to decrease processing times. As in Desrosiers et al. (1988) also the case of multiple depots is considered. A real life instance with 2545 customers could be solved.

A classical *cluster first route second* algorithm is discussed in Borndörfer et al. (1997). The clustering as well as the routing problem are modeled as set partitioning problems. The clustering problem can be solved optimally while the routing subproblems are solved heuristically by a branch and bound algorithm. Customer satisfaction is taken care of in terms of punctual service and customer ride times are implicitly considered by the time windows. The real life data sets contained up to 1771 customer requests.

Wolfler Calvo and Colorni (2002) propose another *cluster first route second* algorithm using an assignment heuristic to solve the clustering problem, subsequently, applying vertex reinsertions. Instances with up to 180 customers are

solved.

Early heuristic algorithms to solve the *dynamic DARP* are discussed in Daganzo (1978). Daganzo analyzes three different *insertion* algorithms. The first algorithm consists in visiting the closest stop next, the second heuristic consists in visiting the closest origin or the closest destination in alternating order and the third algorithm only allows the insertion of delivery locations after a fixed number of passengers have been picked up.

The dynamic case of the multi vehicle DARP is also studied in Dial (1995). New transportation requests are assigned to clusters according to *least cost insertion*. Routes are then optimized using dynamic programming. Results for a real life problem instance are reported.

A multi-objective approach to the dynamic DARP is followed by Madsen et al. (1995). They propose an *insertion* based algorithm called REBUS. The objectives considered are the total driving time, the number of vehicles, the total waiting time, the deviation from promised service times as well as cost. It was applied to real-life instances with up to 300 customers.

Coslovich et al. (2005) propose a *two-phase insertion* heuristic for the dynamic DARP. A simple insertion procedure allows for quick answers with respect to inclusion or rejection of a new customer. The subsequent step consists in improving on the initial solution by means of a local search operator, i.e. a 2-opt arc swap. Instances with up to 50 customers were solved.

Caramia et al. (2002) use a *dynamic programming* algorithm to iteratively solve the single vehicle subproblems to optimality. Results for instances with up to 50 clients per hour are reported.

Teodorovic and Radivojevic (2000) propose a fuzzy logic *cluster first route second* algorithm. Thus clusters as well as routes are formed by use of approximate reasoning. The problem tackled is the dynamic DARP. Results for up to 900 customer instances are reported.

Another *cluster first route second* algorithm for the dynamic problem is proposed by Colorni and Righini (2001). Considering only the most urgent requests the routing subproblems can be solved to optimality using a branch and bound algorithm. In contrast to many other approaches instead of exact routes only the scheduling of the different pickup and drop off locations is transmitted to the drivers in order to increase driver responsibility.

Horn (2002a) provide a software environment for fleet scheduling and dispatching of demand responsive services. The system can handle advance as well as immediate requests. New incoming requests are inserted into existing routes according to *least cost insertion*. A steepest descend improvement phase is run periodically. Also automated vehicle dispatching procedures to achieve a good combination of efficient vehicle deployment as well as customer service are included. The system was tested using the modeling framework LITRES-2, Horn (2002b), on a real life data set covering a 24 hour time period of taxi operations and 4282 customer requests.

### 3.3.3 Metaheuristics

One of the first metaheuristic approaches for the *static DARP* was proposed by Potvin and Rousseau (1992). It is a *constraint directed search* algorithm based on the notion of beam search using Jaw et al. (1986) insertion technique and a post-optimization phase. Up to 90 customer instances were solved.

A *simulated annealing* algorithm was presented by Colomi et al. (1996). The only objective considered refers to the maximization of service quality. Baugh et al. (1998) propose another simulated annealing approach that incorporates a cluster first route second strategy. Moreover, multiple objectives are considered. These are the number of vehicles, the total distance traveled and customer inconvenience measured in terms of time window violations. The algorithm was tested on real life data sets of a company that serves up to 300 customers per day.

Toth and Vigo (1997) propose another local search based algorithm, i.e. a *tabu thresholding* algorithm that is compared to the heuristic algorithm discussed in Toth and Vigo (1996). The tabu thresholding algorithm uses parallel insertion to obtain an initial solution. Real life data sets are used to test the algorithms.

Cordeau and Laporte (2003b) propose a *tabu search* algorithm for the static multi vehicle DARP. Time windows are considered at either origin or destination depending on the type of request (inbound or outbound). The neighborhood considered in the tabu search is defined by moving one request to another route. The best possible move serves to generate a new incumbent solution. Reverse moves are declared tabu. However, an aspiration criterion is defined, such that tabu moves that provide a better solution, w.r.t. all other solutions already constructed by the same move, can constitute a new incumbent. The initial solution is constructed randomly only respecting precedence constraints. Solutions to instances with up to 295 requests are reported. The above version of the proposed tabu search is adapted to the *dynamic DARP* by means of parallelization in Attanasio et al. (2004). Different parallelization strategies are tested on instances with up to 144 customers.

Aldaihani and Dessouky (2003) propose another *tabu search* algorithm using an insertion-improvement procedure to obtain the initial solution. The problem type considered is a hybrid system between static dial-a-ride and fixed-route-transit systems. Instances with up to 155 requests were solved.

Ho and Haugland (2004) develop and compare a *tabu search* and a hybrid GRASP-tabu search algorithm for the *probabilistic DARP*, where each user requires service with a certain probability. Solutions to the instances of Cordeau and Laporte (2003b), adapted to the probabilistic case, are reported.

Melachrinoudis et al. (2005) formulate a static real life DARP considering multiple depots and a heterogeneous fleet. Total transportation cost and user inconvenience are lexicographically minimized. The solution method used is a *tabu search* algorithm. To generate initial routes customers are ordered according to a combination of their distance from the depot and the time until the beginning of service at their origins. Results for real life instances with up to 4



requests per day were reported.

A population based algorithm applied to the static DARP, namely a *genetic algorithm*, was developed by Uchimura et al. (1999). Only recently another genetic algorithm was proposed by Bergvinsdottir et al. (2006). The solution encoding used splits the clustering and the routing part. Thus the customer clustering is solved by means of a genetic algorithm while the routing part is solved by a modified space-time nearest neighbor heuristic. The instances solved are those of Cordeau and Laporte (2003b). An application of a grouping genetic algorithm to solve der static DARP is reported in Rekiek et al. (2006). Again, a chromosome consists of clusters of customers, each cluster corresponding to one vehicle. Crossover and mutation are carried out on this level. Non-served clients due to these operations are inserted using a best-fit heuristic. Local improvement procedures minimizing ride time are used to improve on the service plan. Results for real-life data sets with up to 164 clients are reported.

### 3.3.4 Related work

Dealing with the transportation of people, especially handicapped or elderly, research has also been dedicated to the comparison of dial-a-ride systems with other modes of transportation, e.g. public bus systems or mixed systems. Early studies of dial-a-ride transportation systems are discussed in Carlson (1976), Teixeira and Karash (1975). Elmberg (1978) already tested a robot dispatcher dial-a-ride system in Sweden. Daganzo (1984) compares fixed route transit systems with checkpoint dial-a-ride and door-to-door dial-a-ride systems. He concludes that most of the time either fixed route systems or door-to-door transportation is the appropriate choice, while the intermediate solution of checkpoint dial-a-ride systems hardly ever performs well. Belisle et al. (1986) investigate the impact of different operating scenarios on the quality of transportation systems for the handicapped. More recent studies comparing dial-a-ride and traditional bus systems by means of simulation were conducted by Noda et al. (2003), Noda (2005). In Noda et al. (2003) the heuristic used to schedule the customers in the dial-a-ride system was a successive best insertion algorithm, for the fixed route-system a genetic algorithm was used. Another study by means of simulation w.r.t. the usability of dial-a-ride systems in urban areas was presented by Shinoda et al. (2003). Mageean and Nelson (2003) study and evaluate telematics based demand responsive transport services in Europe. Palmer et al. (2004) study the impact of management practices as well as advanced technologies in the context of demand responsive transport systems. The impact of information flows is investigated by Diana (2006). In Diana et al. (2006) the optimal fleet size w.r.t. predetermined service quality is studied.

Research has also been dedicated to possible ways of computation time reduction. Hunsaker and Savelsbergh (2002), e.g., propose a fast feasibility check for DARP. The proposed procedure can deal with waiting times, ride times as well as time window restrictions. Castelli et al. (2002) discuss three algorithms granting 2-opt-improvement feasibility.

A problem class referred to as *online DARP* in the literature deals with

the real time scheduling of server moves, see Ascheuer et al. (2000), Feuerstein and Stougie (2001), Hauptmeier et al. (2000), Krumke et al. (2005). The transportation requests consist of objects not people. However, since the objective minimized is the completion time, the proposed approximation algorithms might also be applicable in the context of people transportation. Lipmann et al. (2004) study the influence of restricted information on the online DARP, i.e. the destination of a request is only revealed after the object has been picked up. Its extension to the time window case is tackled in Yi and Tian (2005). The static version on a caterpillar graph is studied in Coja-Oghlan et al. (2005). A heuristic algorithm for an extended static version was proposed by Hauptmeier et al. (2001), considering a single vehicle with unit capacity traveling between the different origins and destinations. At the origins more than one object waits to be transported. These objects are ordered according to the FIFO rule.

Another problem class related to the DARP is the *car pooling* problem. It consists in finding subsets of employees that share a car and in determining the path the driver should follow and possibly also who should be the driver. In contrast to the DARP either origin or destination are the same for all users depending on whether the trip is from home to the office or back. Two variants can be investigated, either one car pool for both ways or differing to-work and from-work problems. Baldacci et al. (2004) propose an exact as well as a heuristic procedure to solve the car pooling problem. A real life application was reported by Wolfler Calvo et al. (2004) and Maniezzo et al. (2004) propose an ant colony optimization algorithm for the long-term problem.

### 3.4 Benchmark instances for pickup and delivery problems

Nanry and Barnes (2000) provide a set of benchmark instances for the static multi vehicle PDPTW based on the VRPTW instances of Solomon (1987) with 25, 50 and 100 customers. Optimal solution schedules for the VRPTW instances were computed by means of the solution procedure developed by Carlton (1995). Given these schedules, customers were randomly paired, adding dummy customers whenever a route contained an odd number of customers. For each problem size, data sets with randomly distributed, clustered and mixed customer locations were available. These instances are referred to as NB00 in Table 1.

Li and Lim (2001) generate a set of 56 problem instances for the PDPTW based on those of Solomon (1987) for the VRP. In contrast to Nanry and Barnes (2000) not the optimal schedules were used to obtain paired pickup and delivery locations. Customers, appearing on the same route in a solution of the VRPTW by using the solution procedure proposed by Li et al. (2001), were randomly paired. All problems consist of 100 customer nodes and several additional dummy nodes if required by the pairing procedure. Thus 6 problem classes were generated, two with randomly distributed customers, two with clustered customers and two with partially randomly distributed and partially clustered customers. This set is denoted as LL01 in Table 1. The extended set, containing instances with up to 500 customer requests, based on the extended Solomon

instances, is denoted as LL01+.

Cordeau and Laporte (2003b) develop a data set of 20 instances for the static multi vehicle DARP, containing 24 – 144 requests. Half of the requests are assumed to be inbound (tight time window at origin) and half to be outbound (tight time window at destination). Pickup and delivery locations were generated around several seed points in the square  $[-10, 10]$ . The first 10 instances are associated with narrow time windows, the second 10 instances with wider time windows. This set of instances is denoted as CL03 in Table 3.

Cordeau (2006) present two sets of instances for the DARP with up to 32 requests, containing 15 instances each. Pickup and delivery locations are randomly distributed in the square  $[-10, 10] \times [-10, 10]$ . Inbound requests have tight time windows on the origin, outbound requests on the destination. The two data sets differ w.r.t. vehicle capacities. In the first set every vehicle is associated with a capacity  $Q = 3$ , unit user demand, and a maximum user ride time of 30 minutes, in the second set the capacity is set to  $Q = 6$  and every user is associated with a value of  $q_i$  randomly chosen in  $\{1, \dots, Q\}$ , and a maximum ride time of 60 minutes. These instances are denoted as Cor06 in Table 3.

## Acknowledgements

This work was supported by the Special Research Program Translational Research under grant #L286-N04 from the Fonds zur Förderung der wissenschaftlichen Forschung (FWF). We thank the transportation research community for their various contributions. Special thanks go to Michel Gendreau and Jacques Desrosiers for their kind support.

## References

- Aldaihani M, Dessouky MM (2003) Hybrid scheduling methods for paratransit operations. *Comput Ind Eng* 45:75–96
- Alfa AS (1986) Scheduling of vehicles for transportation of elderly. *Transport Plann Tech* 11:203–212
- Ambrosini M, Caruso T, Foresti S, Righini G (2004) A GRASP for the pickup and delivery problem with rear loading. Tech. Rep. Note del Polo - Ricerca n.65, DTI, University of Milan
- Anily S, Bramel J (1999) Approximation algorithms for the capacitated traveling salesman problem with pickups and deliveries. *Naval Res Log* 46:654–670
- Archetti C, Hertz A, Speranza MG (2006a) A tabu search algorithm for the split delivery vehicle routing problem. *Transport Sci* 40:64–73
- Archetti C, Savelsbergh M, Speranza MG (2006b) Worst-case analysis for split delivery vehicle routing problems. *Transport Sci* 40:226–234

- Ascheuer N, Escudero LF, Grotschel M, Stoer M (1993) A cutting plane approach to the sequential ordering problem (with applications to job scheduling in manufacturing). *SIAM J Optim* 3:25–42
- Ascheuer N, Krumke SO, Rambau J (2000) Online dial-a-ride problems: Minimizing the completion time. In: *STACS 2000: 17th Annual Symposium on Theoretical Aspects of Computer Science*, Lille, France, February 2000. Proceedings. Springer-Verlag GmbH, Berlin-Heidelberg, Lecture Notes in Computer Science, 639–650
- Assad AA (1988) Modeling and implementation issues in vehicle routing. In: *Vehicle Routing: Methods and Studies*. North-Holland, Amsterdam, Studies in management science and systems, vol. 16, 7–45
- Attanasio A, Cordeau JF, Ghiani G, Laporte G (2004) Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem. *Parallel Comput* 30:377–387
- Baldacci R, Maniezzo V, Mingozzi A (2004) An exact method for the car pooling problem based on Lagrangean column generation. *Oper Res* 52:422–439
- Baugh JW, Krishna G, Kakivaya R, Stone JR (1998) Intractability of the dial-a-ride problem and a multiobjective solution using simulated annealing. *Eng Optim* 30:91–123
- Belisle JP, Desrosiers J, Dumas Y, Rousseau JM, Roy S, Soumis F (1986) The impact on vehicle routing of various operational rules of a transportation system for handicapped. In: *Third International Conference on Mobility and Transportation of Handicapped Persons*, Office of the Secretary of Transportation, Washington, DC 20590, 6.47–6.50
- Bent R, Van Hentenryck P (2006) A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows. *Comput Oper Res* 33:875–893
- Bergvinsdottir KB, Larsen J, Jørgensen RM (2006) Solving the dial-a-ride problem using genetic algorithms. Tech. rep., Technical University of Denmark. Kgs. Lyngby, Denmark. To appear in the *J Oper Res Soc*
- Bodin L, Sexton T (1986) The multi-vehicle subscriber dial-a-ride problem. *TIMS Studies in Management Science* 22:73–86
- Borndörfer R, Grötschel M, Klostermeier F, Küttner C (1997) Telebus Berlin: Vehicle scheduling in a dial-a-ride system. Tech. Rep. SC 97-23, Konrad-Zuse-Zentrum für Informationstechnik Berlin
- Brønmo G, Christiansen M, Fagerholt K, Nygreen B (2005) A multi-start local search heuristic for ship scheduling – a computational study. *Comput Oper Res* To appear
- Caramia M, Italiano GF, Oriolo G, Pacifici A, Perugia A (2002) Routing a fleet of vehicles for dynamic combined pickup and deliveries services. In: Chamoni P, Leisten R, Martin A, Minnemann J, Stadtler H (eds.) *Operations Research Proceedings 2001*. Springer, Berlin, Germany, 3–8
- Caricato P, Ghiani G, Grieco A, Guerriero E (2003) Parallel tabu search for a pickup and delivery problem under track contention. *Parallel Computing* 29:631–639

- Carlson RC (1976) Anatomy of a systems failure: Dial-a-ride in Santa Clara County, California. *Transportation* 5:3–16
- Carlton WB (1995) A tabu search approach to the general vehicle routing problem. Ph.D. thesis, University of Texas at Austin
- Carrabs F, Cordeau JF, Laporte G (2005) Variable neighborhood search for the pickup and delivery traveling salesman problem with LIFO loading. *Tech. Rep. CRT-2005-12*
- Castelli L, Coslovich L, Pesenti R, Ukovich W (2002) Improving techniques for an on-line dial-a-ride problem with time windows and capacity constraints. In: *Proceedings of the 13th Mini-EURO Conference and 9th Meeting of the EURO Working Group on Transportation*, Bari, Italy - June 10-13 2002
- Chalasani P, Motwani R (1999) Approximating capacitated routing and delivery problems. *SIAM J Comput* 28:2133–2149
- Christiansen M (1999) Decomposition of a combined inventory and time constrained ship routing problem. *Transport Sci* 33:3–16
- Christiansen M, Fagerholt K (2002) Robust ship scheduling with multiple time windows. *Naval Res Log* 49:611–625
- Christiansen M, Fagerholt K, Ronen D (2004) Ship routing and scheduling: status and perspectives. *Transport Sci* 38:1–18
- Christiansen M, Nygreen B (1998a) A method for solving ship routing problems with inventory constraints. *Ann Oper Res* 81:357–387
- Christiansen M, Nygreen B (1998b) Modeling path flows for a combined ship routing and inventory management problem. *Ann Oper Res* 82:391–412
- Christiansen M, Nygreen B (2005) Robust inventory ship routing by column generation. In: Desaulniers G, Desrosiers J, Solomon MM (eds.) *Column generation*. Springer, New York
- Coja-Oghlan A, Krumke SO, Nierhoff T (2005) A hard dial-a-ride problem that is easy on average. *J Sched* 8:197–210
- Colomi A, Dorigo M, Maffioli F, Maniezzo V, Righini G, Trubian M (1996) Heuristics from nature for hard combinatorial optimization problems. *Int Trans Oper Res* 3:1–21
- Colomi A, Righini G (2001) Modeling and optimizing dynamic dial-a-ride problems. *Int Trans Oper Res* 8:155–166
- Cordeau JF (2006) A branch-and-cut algorithm for the dial-a-ride problem. *Oper Res* 54:573–586
- Cordeau JF, Desaulniers G, Desrosiers J, Solomon MM, Soumis F (2001) VRP with time windows. In: Toth P, Vigo D (eds.) *The Vehicle Routing Problem*. SIAM, Philadelphia, PA, vol. 9, 175–193

- Cordeau JF, Iori M, Laporte G, Salazar-Gonzalez JJ (2006) Branch-and-cut for the pickup and delivery traveling salesman problem with LIFO loading. Tech. Rep. OR-06-01, DEIS, University of Bologna
- Cordeau JF, Laporte G (2003a) The dial-a-ride problem (DARP): Variants modeling issues and algorithms. *4OR* 1:89–101
- Cordeau JF, Laporte G (2003b) A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transport Res B-Meth* 37:579–594
- Cordeau JF, Laporte G, Potvin JY, Savelsbergh MWP (2004) Transportation on demand. In: *Handbooks in Operations Research and Management Science*, Elsevier, North-Holland, Amsterdam, vol. (to appear)
- Coslovich L, Pesenti R, Ukovich W (2005) A two-phase insertion technique of unexpected customers for a dynamic dial-a-ride problem. *Eur J Oper Res* To appear, available online
- Creput JC, Koukam A, Kozlak J, Lukasik J (2004) An evolutionary approach to pickup and delivery problem with time windows. In: Bubak M, van Albada GD, Sloot PMA, Dongarra JJ (eds.) *Computational Science - ICCS 2004: 4th International Conference*, Krak, Poland, June 6-9, 2004, Proceedings. Springer, 1102–1108
- Cullen F, Jarvis J, Ratliff D (1981) Set partitioning based heuristics for interactive routing. *Networks* 11:125–144
- Daganzo CF (1978) An approximate analytic model of many-to-many demand responsive transportation systems. *Transport Res* 12:325–333
- Daganzo CF (1984) Checkpoint dial-a-ride systems. *Transport Res B-Meth* 18:315–327
- Derigs U, Döhmer T (2006) Indirect search for the vehicle routing problem with pickup and delivery and time windows. Working Paper
- Desaulniers G, Desrosiers J, Erdmann A, Solomon MM, Soumis F (2001) VRP with pickup and delivery. In: Toth P, Vigo D (eds.) *The Vehicle Routing Problem*. SIAM, Philadelphia, PA, vol. 9, 225–242
- Desrochers M, Lenstra JK, Savelsbergh MWP, Soumis F (1988) Vehicle routing with time windows: Optimization and approximation. In: Golden BL, Assad AA (eds.) *Vehicle Routing: Methods and Studies*. Elsevier (North-Holland), Amsterdam, 65–84
- Desrosiers J, Dumas Y (1988) The shortest path for the construction of vehicle routes with pick-up, delivery and time constraints. In: Eiselt H, Pederzoli G (eds.) *Advances in Optimization and Control*. Springer-Verlag, Heidelberg Berlin, Lecture Notes in Economics and Mathematical Systems, 144–157
- Desrosiers J, Dumas Y, Soumis F (1986) A dynamic programming solution of the large-scale single-vehicle dial-a-ride problem with time windows. *Am J Math Manag Sci* 6:301–325
- Desrosiers J, Dumas Y, Soumis F (1988) The multiple vehicle dial-a-ride-problem. In: Daduna J, Wren A (eds.) *Computer-Aided Transit Scheduling*. Lecture Notes in Economics and Mathematical Systems. Springer, Berlin, vol. 308, 15–27

- Desrosiers J, Dumas Y, Soumis F, Taillefer S, Villeneuve D (1991) An algorithm for mini-clustering in handicapped transport. Technical Report Cahiers du GERAD G-91-02, Ecole des Hautes Etudes Commerciales, Montreal, Canada
- Dial R (1995) Autonomous dial-a-ride transit introductory overview. *Transport Res C-Emer* 3:261–275
- Diana M (2004) Innovative systems for the transportation disadvantaged: towards more efficient and operationally usable planning tools. *Transport Plan Tech* 27:315–331
- Diana M (2006) The importance of information flows temporal attributes for the efficient scheduling of dynamic demand responsive transport services. *J Adv Transport* 40:23–46
- Diana M, Dessouky M, Xia N (2006) A model for the fleet sizing of demand responsive transportation services with time windows. *Transport Res B-Meth* 40:651–666
- Diana M, Dessouky MM (2004) A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows. *Transport Res B-Meth* 38:539–557
- Dror M, Fortin D, Roucairol C (1998) Redistribution of self-service electric cars: A case of pickup and delivery. Tech. Rep. W.P. 3543, INRIA-Rocquencourt, Rocquencourt, France
- Dumas Y, Desrosiers J, Soumis F (1989) Large scale multi-vehicle dial-a-ride problems. Tech. Rep. G-89-30, Ecole des Hautes Etudes Commerciales, Montreal, Canada
- Dumas Y, Desrosiers J, Soumis F (1991) The pickup and delivery problem with time windows. *Eur J Oper Res* 54:7–22
- Dumitrescu I (2005) Polyhedral results for the pickup and delivery travelling salesman problem. Tech. Rep. CRT-2005-27, Centre de recherche sur les transports, Universite Montreal, Canada
- Elmberg CM (1978) Dial-a-ride with customer operated dispatching. *Transportation* 7:35–43
- Escudero LF (1988) An inexact algorithm for the sequential ordering problem. *Eur J Oper Res* 37:232–253
- Fabri A, Recht P (2006) On dynamic pickup and delivery vehicle routing with several time windows and waiting times. *Transport Res B-Meth* 40:335–350
- Fagerholt K (2001) Ship scheduling with soft time windows: An optimisation based approach. *Eur J Oper Res* 131:559–571
- Fagerholt K, Christiansen M (2000a) A combined ship scheduling and allocation problem. *J Oper Res Soc* 51:834–842
- Fagerholt K, Christiansen M (2000b) A travelling salesman problem with allocation, time window and precedence constraints – an application to ship scheduling. *Int Trans Oper Res* 7:231–244

- Feuerstein E, Stougie L (2001) On-line single-server dial-a-ride problems. *Theor Comput Sci* 268:91–105
- Fu L (2002a) Scheduling dial-a-ride paratransit under time varying, stochastic congestion. *Transport Res B-Meth* 36:485–06
- Fu L (2002b) A simulation model for advanced evaluating dial-a-ride paratransit systems. *Transport Res A-Pol* 36:291–307
- Gambardella LM, Dorigo M (2000) Ant colony system hybridized with a new local search for the sequential ordering problem. *INFORMS J Comput* 12:237–255
- Gendreau M, Guertin F, Potvin J, Seguin R (1998) Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries. Tech. Rep. CRT-98-10, Centre de recherche sur les transports, Université Montreal, Canada
- Gendreau M, Laporte G, Vigo D (1999) Heuristics for the traveling salesman problem with pickup and delivery. *Comput Oper Res* 26:699–714
- Gendreau M, Potvin JY (1998) Dynamic vehicle routing and dispatching. In: Crainic T, Laporte G (eds.) *Fleet Management and Logistics*. Kluwer, New York, 115–126
- Ghiani G, Guerriero F, Laporte G, Musmanno R (2003) Real-time vehicle routing: Solution concepts, algorithms and parallel computing strategies. *Eur J Oper Res* 151:1–11
- Gutenschwager K, Niklaus C, Voss S (2004) Dispatching of an electronic monorail system: Applying metaheuristics to an online pickup and delivery problem. *Transport Sci* 38:434–446
- Hauptmeier D, Krumke SO, Rambau J (2000) The online dial-a-ride problem under reasonable load. In: *Algorithms and Complexity: 4th Italian Conference, CIAC 2000, Rome, Italy, March 2000. Proceedings*. Springer-Verlag GmbH, Lecture Notes in Computer Science, 125–136
- Hauptmeier D, Krumke SO, Rambau J, Wirth HC (2001) Euler is standing in line dial-a-ride problems with precedence-constraints. *Discrete Appl Math* 113:87–107
- Healy P, Moll R (1995) A new extension of local search applied to the dial-a-ride problem. *Eur J Oper Res* 83:83–104
- Hernández-Pérez H, Salazar-González JJ (2003) The one-commodity pickup-and-delivery travelling salesman problem. In: Jünger M, Reinelt G, Rinaldi G (eds.) *Combinatorial Optimization - Eureka, You Shrink!: Papers Dedicated to Jack Edmonds. 5th International Workshop Aussois, France, March 5-9, 2001*. Springer, Lecture Notes In Computer Science, 89–104
- Hernández-Pérez H, Salazar-González JJ (2004a) A branch-and cut algorithm for the traveling salesman problem with pickup and delivery. *Discrete Appl Math* 145:126–139
- Hernández-Pérez H, Salazar-González JJ (2004b) Heuristics for the one-commodity pickup-and-delivery traveling salesman problem. *Transport Sci* 38:245–255



- Ho SC, Haugland D (2004) Local search heuristics for the probabilistic dial-a-ride problem. Tech. Rep. 286, University of Bergen
- Horn MET (2002a) Fleet scheduling and dispatching for demand-responsive passenger services. *Transport Res C-Emer* 10:35–63
- Horn MET (2002b) Multi-modal and demand-responsive passenger transport systems: a modelling framework with embedded control systems. *Transport Res A-Pol* 36:167–188
- Hunsaker B, Savelsbergh MWP (2002) Efficient feasibility testing for dial-a-ride problems. *Oper Res Lett* 30:169–173
- Ioachim I, Desrosiers J, Dumas Y, Solomon MM, Villeneuve D (1995) A request clustering algorithm for door-to-door handicapped transportation. *Transport Sci* 29:63–78
- Jaw J, Odoni AR, Psaraftis HN, Wilson NHM (1986) A heuristic algorithm for the multi-vehicle advance-request dial-a-ride problem with time windows. *Transport Res B-Meth* 20:243–257
- Jih WR, Hsu YJ (1999) Dynamic vehicle routing using hybrid genetic algorithms. In: *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*. IEEE Computer Society, Los Alamitos, CA, 453–458
- Jung S, Haghani A (2000) A genetic algorithm for a pick-up and delivery problem with time windows. *Transport Res Rec* 1733:1–7
- Kalantari B, Hill AV, Arora SR (1985) An algorithm for the traveling salesman problem with pickup and delivery customers. *Eur J Oper Res* 22:377–386
- Kikuchi S (1984) Scheduling of demand-responsive transit vehicles. *J Transp Eng* 110:511–520
- Kikuchi S, Rhee J (1989) Scheduling algorithms for demand-responsive transportation system. *J Transp Eng* 115:630–645
- Krumke SO, de Paepe WE, Poensgen D, Lipmann M, Marchetti-Spaccamela A, Stougie L (2005) On minimizing the maximum flow time in the online dial-a-ride problem. In: *Approximation and Online Algorithms: Third International Workshop, WAOA 2005, Palma de Mallorca, Spain, October 6-7, 2005, Revised Selected Papers*. Springer-Verlag GmbH, Lecture Notes in Computer Science, 258–269
- Kubo M, Kasugai H (1990) Heuristic algorithms for the single vehicle dial-a-ride problem. *J Oper Res Soc Japan* 33:354–364
- Landrieu A, Mati Y, Binder Z (2001) A tabu search heuristic for the single vehicle pickup and delivery problem with time windows. *J Intell Manuf* 12:497–508
- Lau HC, Liang Z (2001) Pickup and delivery with time windows : Algorithms and test case generation. In: *IEEE Computer Society, eds, 13th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'01)*. 333–340
- Li H, Lim A (2001) A metaheuristic for the pickup and delivery problem with time windows. In: *13th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'01)*. IEEE Computer Society, Los Alamitos, CA, 333–340

- Li H, Lim A, Huang J (2001) Local search with annealing-like restarts to solve the VRPTW. Tech. rep., Department of Computer Science, National University of Singapore
- Lim A, Wang F, Xu Z (2005) The capacitated traveling salesman problem with pickups and deliveries on a tree. In: Deng X, Du D (eds.) *Algorithms and Computation: 16th International Symposium, ISAAC 2005, Sanya, Hainan, China, December 19-21, 2005. Proceedings.* 1061–1070
- Lim H, Lim A, Rodrigues B (2002) Solving the pickup and delivery problem with time windows using squeaky wheel optimization with local search. In: *American Conference on Information Systems, AMICS 2002, Dallas, USA*
- Lin S (1965) Computer solutions of the traveling salesman problem. *AT&T Tech J* 44:2245–2269
- Lipmann M, Lu X, de Paepe WE, Sitters RA, Stougie L (2004) On-line dial-a-ride problems under a restricted information model. *Algorithmica* 40:319–329
- Little J, Murty K, Sweeney D, Karel C (1963) An algorithm for the traveling salesman problem. *Oper Res* 11:972–989
- Lokin FCJ (1978) Procedures for traveling salesman problems with additional constraints. *Eur J Oper Res* 3:135–141
- Lu Q, Dessouky MM (2004) An exact algorithm for the multiple vehicle pickup and delivery problem. *Transport Sci* 38:503–514
- Lu Q, Dessouky MM (2006) A new insertion-based construction heuristic for solving the pickup and delivery problem with time windows. *European Journal of Operational Research* 175:672–687
- Madsen OBG, Ravn HF, Rygaard JM (1995) A heuristic algorithm for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives. *Ann Oper Res* 60:193–208
- Mageean J, Nelson JD (2003) The evaluation of demand responsive transport services in europe. *J Transport Geogr* 11:255–270
- Malca F, Semet F (2004) A tabu search algorithm for a dynamic pickup and delivery vehicle routing problem. In: *Triennial Symposium on Transportation Analysis, Le Gosier, Guadeloupe, France, juin.*
- Maniezzo V, Carbonaro A, Hildmann H (2004) An ANTS heuristic for the long-term car pooling problem. In: Onwubolu G, Babu B (eds.) *New Optimization Techniques in Engineering.* Springer, Heidelberg Berlin, 411–430
- Melachrinoudis E, Ilhan AB, Min H (2005) A dial-a-ride problem for client transportation in a health-care organization. *Comput Oper Res* To appear, available online
- Mitrovic-Minic S (1998) Pickup and delivery problem with time windows: A survey. Tech. Rep. SFU CMPT TR 1998-12, School of Computing Science, Simon Fraser University, Burnaby, BC, Canada

- Mitrovic-Minic S, Krishnamurti R, Laporte G (2004) Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows. *Transport Res B-Meth* 38:669–685
- Mitrovic-Minic S, Laporte G (2004) Waiting strategies for the dynamic pickup and delivery problem with time windows. *Transport Res B-Meth* 38:635–655
- Mosheiov G (1994) The traveling salesman problem with pickup and delivery. *Eur J Oper Res* 79:299–310
- Nanry WP, Barnes W (2000) Solving the pickup and delivery problem with time windows using reactive tabu search. *Transport Res B-Meth* 34:107–121
- Noda I (2005) Scalability of dial-a-ride systems: A case study to assess utilities of ubiquitous mass user support. In: Ishida T, Gasser L, Nakashima H (eds.) *Massively Multi-Agent Systems I: First International Workshop, MMAS 2004, Kyoto, Japan, December 10-11, 2004, Revised Selected and Invited Papers. Massively Multi-agent Systems in Public Space*. Springer-Verlag GmbH, Lecture Notes in Computer Science, 323–334
- Noda I, Ohta M, Shinoda K, Kumad Y, Nakashima H (2003) Evaluation of usability of dial-a-ride systems by social simulation. In: Hales D, Edmonds B, Norling E, Rouchier J (eds.) *Multi-Agent-Based Simulation III. MABS Techniques for Real World Modelling*. Springer-Verlag GmbH, Lecture Notes in Computer Science, 167–181
- Palmer K, Dessouky MM, Abdelmaguid T (2004) Impacts of management practices and advanced technologies on demand responsive transit systems. *Transport Res A-Pol* 38:495–509
- Pankratz G (2005a) Dynamic vehicle routing by means of a genetic algorithm. *International Journal of Physical Distribution and Logistics Management* 35:362–383
- Pankratz G (2005b) A grouping genetic algorithm for the pickup and delivery problem with time windows. *OR Spectrum* 27:21–41
- Pisinger D, Ropke S (2005) A general heuristic for vehicle routing problems. *Comput Oper Res* To appear, available online
- Popken DA (2006) Controlling order circuitry in pickup and delivery problems. *Transport Res E-Log* 42:431–443
- Potvin JY, Rousseau JM (1992) Constrained-directed search for the advance request dial-a-ride problem with service quality constraints. In: Balci O, Shrada R, Zenios ZA (eds.) *Computer Science and Operations Research: New Developments in Their Interfaces*. Pergamon Press, Oxford, 457–74
- Potvin JY, Rousseau JM (1993) A parallel route building algorithm for the vehicle routing and scheduling problem with time windows. *Eur J Oper Res* 66:331–340
- Potvin JY, Shen Y, Dufour G, Rousseau JM (1995) Learning techniques for an expert vehicle dispatching system. *Expert Systems with Applications* 8:101–109

- Psaraftis HN (1980) A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. *Transport Sci* 14:130–154
- Psaraftis HN (1983a) Analysis of an  $O(n^2)$  heuristic for the single vehicle many-to-many Euclidean dial-a-ride problem. *Transport Res B-Meth* 17:133–145
- Psaraftis HN (1983b) An exact algorithm for the single vehicle many to many dial-a-ride problem with time windows. *Transport Sci* 17:35–357
- Psaraftis HN (1983c) k-interchange procedures for local search in a precedence-constrained routing problem. *Eur J Oper Res* 13:391–402
- Psaraftis HN (1986) Scheduling large-scale advance-request dial-a-ride systems. *Am J Math Manag Sci* 6:327–367
- Psaraftis HN (1988) Dynamic vehicle routing problems. In: Golden BL, Assad AA (eds.) *Vehicle Routing: Methods and Studies*. Elsevier (North-Holland), Amsterdam, 223–248
- Rebibo KK (1974) A computer controlled dial-a-ride system. traffic control and transportation systems. In: *Proceedings of 2nd IFAC/IFIP/IFORS Symposium Monte Carlo, September 1974*. North-Holland, Amsterdam
- Recker WW (1995) The household activity pattern problem: General formulation and solution. *Transport Res B-Meth* 29:61–77
- Rekiek B, Delchambre A, Saleh HA (2006) Handicapped person transportation: An application of the grouping genetic algorithm. *Eng Appl Artif Intel* 19:511–520
- Renaud J, Boctor FF, Laporte G (1996) A fast composite heuristic for the symmetric traveling salesman problem. *INFORMS J Comput* 8:134–143
- Renaud J, Boctor FF, Laporte G (2002) Perturbation heuristics for the pickup and delivery traveling salesman problem. *Comput Oper Res* 29:1129 – 1141
- Renaud J, Boctor FF, Ouenniche J (2000) A heuristic for the pickup and delivery traveling salesman problem. *Comput Oper Res* 27:905–916
- Ropke S, Cordeau JF, Laporte G (2005) Models and a branch-and-cut algorithm for pickup and delivery problems with time windows. Tech. Rep. CRT-2005-25, Centre de recherche sur les transports, Université Montreal, Canada
- Ropke S, Pisinger D (2005) An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transport Sci* To appear
- Ropke S, Pisinger D (2006) A unified heuristic for a large class of vehicle routing problems with backhauls. *Eur J Oper Res* 171:750–775
- Roy S, Rousseau J, Lapalme G, Ferland J (1985a) Routing and scheduling for the transportation of disabled persons : The algorithm. Tech. Rep. TP 5596E, Centre de Recherche sur les Transports, Montréal, Canada
- Roy S, Rousseau JM, Lapalme G, Ferland J (1985b) Routing and scheduling for the transportation of disabled persons : The tests. Tech. Rep. TP 5598E, Centre de Recherche sur les Transports, Montréal, Canada

- Ruland KS, Rodin EY (1997) The pickup and delivery problem: Faces and branch-and-cut algorithm. *Comput Math Appl* 33:1–13
- Savelsbergh MWP, Sol M (1995) The general pickup and delivery problem. *Transport Sci* 29:17–29
- Savelsbergh MWP, Sol M (1998) DRIVE: Dynamic Routing of Independent VEHicles. *Oper Res* 46:474–490
- Schönberger J, Kopfer H, Mattfeld DC (2003) A combined approach to solve the pickup and delivery selection problem. In: Leopold-Wildburger U, Rendl F, Wäscher G (eds.) *Operations Research Proceedings 2002*. Springer, Berlin-Heidelberg-New York, 150–155
- Sexton T, Bodin LD (1985a) Optimizing single vehicle many-to-many operations with desired delivery times: I. Scheduling. *Transport Sci* 19:378–410
- Sexton T, Bodin LD (1985b) Optimizing single vehicle many-to-many operations with desired delivery times: II. Routing. *Transport Sci* 19:378–410
- Sexton TR, Choi YM (1986) Pickup and delivery of partial loads with "soft" time windows. *Am J Math Manag Sci* 6:369–398
- Shang JS, Cuff CK (1996) Multicriteria pickup and delivery problem with transfer opportunity. *Comput Ind Eng* 30:631–645
- Shen Y, Potvin JY, Rousseau JM, Roy S (1995) A computer assistant for vehicle dispatching with learning capabilities. *Ann Oper Res* 61:189–211
- Shinoda K, Noda I, Ohta M, Kumada Y, Nakashima H (2003) Is dial-a-ride bus reasonable in large scale towns? Evaluation of usability of dial-a-ride systems by simulation. In: Kurumatani K, Chen SH, Ohuchi A (eds.) *Multiagent for Mass User Support: First International Workshop, MAMUS-03 Acapulco, Mexico, August 2003*. Springer-Verlag GmbH, Lecture Notes in Computer Science, 105–119
- Sigurd M, Pisinger D, Sig M (2004) Scheduling transportation of live animals to avoid the spread of diseases. *Transport Sci* 38:197–209
- Solomon M (1987) Algorithms for the vehicle routing problem with time windows. *Oper Res* 35:254–265
- Stein DM (1978a) An asymptotic probabilistic analysis of a routing problem. *Math Oper Res* 3:89–101
- Stein DM (1978b) Scheduling dial-a-ride transportation systems. *Transport Sci* 12:232–249
- Swihart MR, Papstavrou JD (1999) A stochastic and dynamic model for the single-vehicle pick-up and delivery problem. *Eur J Oper Res* 114:447–464
- Teixeira DB, Karash KH (1975) An evaluation of councils on aging dial-a-ride systems in Massachusetts. *Transportation* 4:105–121
- Teodorovic D, Radivojevic G (2000) A fuzzy logic approach to dynamic dial-a-ride problem. *Fuzzy Set Sys* 116:23–33

- Thangiah SR, Awan A (2006) Real-time split-delivery pickup and delivery time window problems with transfer. Tech. Rep. SRT90-2006, Artificial Intelligence Robotics Lab, Slippery Rock University, PA
- Toth P, Vigo D (1996) Fast local search algorithms for the handicapped persons transportation problem. In: Osman IH, Kelly JP (eds.) *Metaheuristics: Theory and Applications*. Kluwer, Boston, MA
- Toth P, Vigo D (1997) Heuristic algorithms for the handicapped persons transportation problem. *Transport Sci* 31:60–71
- Uchimura K, Saitoh T, Takahashi H (1999) The dial-a-ride problem in a public transit system. *Electron Commun Jpn* 82:30–38
- Van der Bruggen LJJ, Lenstra JK, Schuur PC (1993) Variable-depth search for the single vehicle pickup and delivery problem with time windows. *Transport Sci* 27:298–311
- Wilson H, Colvin N (1977) Computer control of the Rochester dial-a-ride system. Tech. Rep. R-77-31, Department of Civil Engineering. MIT Cambridge, MA
- Wilson H, Sussman J, Wang H, Higonnet B (1971) Scheduling algorithms for dial-a-ride system. Tech. Rep. USL TR-70-13, Urban Sysms Laboratory, MIT, Cambridge, MA.
- Wilson H, Weissberg H (1967) Advanced dial-a-ride algorithms research project: Final report. Tech. Rep. R76-20, Department of Civil Engineering. MIT, Cambridge, MA
- Wolfler Calvo R, Colorni A (2002) An approximation algorithm for the dial-a-ride problem. Submitted for publication
- Wolfler Calvo R, De Luigi F, Haastруп P, Maniezzo V (2004) A distributed geographic information system for the daily car pooling problem. *Comput Oper Res* 31:2263–2278
- Wong KI, Bell MGH (2006) Solution of the dial-a-ride problem with multi-dimensional capacity constraints. *Int Trans Oper Res* 13:195–208
- Xiang Z, Chu C, Chen H (2006) A fast heuristic for solving a large-scale static dial-a-ride problem under complex constraints. *Eur J Oper Res* 174:1117–1139
- Xu H, Chen ZL, Rajagopal S, Arunapuram S (2003) Solving a practical pickup and delivery problem. *Transport Sci* 37:347–364
- Yi F, Tian L (2005) On the online dial-a-ride problem with time-windows. In: Megiddo N, Xu Y, Zhu B (eds.) *Algorithmic Applications in Management: First International Conference, AAIM 2005, Xian, China, June 22-25, 2005. Proceedings*. Springer-Verlag GmbH, Lecture Notes in Computer Science, 85–94