University of the Aegean

Dep. of Financial & Management Engineering

Design Operation & Production Systems Lab (DeOPSys Lab)

Scheduled Paratransit Transport Systems

G. Dikas, I. Minis

Lab Report, 2013-06

Abstract

In this report we focus on ways to provide individualized services to people with mobility challenges using existing modes of public transport. We study the design of an interesting case, in which a bus operating in a public transport route may diverge from its nominal path to pick-up passengers with limited mobility and drop them off at their destination. We have modeled the design problem by a mixed integer-linear program, and we developed an exact Branch and Price approach to solve it to optimality. The proposed approach includes a labeling algorithm in which we introduced appropriate dominance rules to guarantee optimality. We have compared the efficiency of our approach with that of related algorithms from the literature. Furthermore, we have used the proposed approach to study key aspects of the system design problem, such as the effect of various constraints on the service level, and the tuning of the system's parameters to address different transport environments.

Keywords: Demand responsive transit systems; Paratransit systems; Branch and Price for MILP; door-to-door bus services

1 Introduction to Demand Responsive Transport Services

In this report we focus on flexible public transport services that address the needs of the elderly, as well as of people with a disability. Of interest are ways to provide individualized bus services to people with mobility challenges using existing modes of public transport. Such services promote equality between client groups, while, at the same time, offer high quality transport to people with a disability, and limit public transport expenditures.

Our work is motivated from the need to serve the first and last leg of any public transport trip; that is the distance between the origin (e.g. home) and the departure bus stop, or the arrival bus stop and the destination (e.g. place of work, hospital). Often such distances are considerable, especially in suburban or rural areas. In order to eliminate these trip legs, some municipalities have introduced semi-flexible bus routes (Flipper Project, 2011). The latter include a reference (nominal) route that serves fixed regular bus stops; however, when there are requests from clients with limited mobility, one or more of the vehicles serving the route may diverge from this nominal path, pick up the clients from their origin and eventually drop them to a regular bus stop or to a special destination. Obviously diversions are possible within a certain distance from the nominal path.

The concept of Demand Responsive Transit (DRT) services has attracted attention recently. Mageean and Nelson (2003) evaluated the application of DRT through telematics in five cities across Europe; their results indicated that DRT services may provide efficient public transportation, and an excellent level of service. Similar studies have been carried out by Brake *et al*. (2007) and Nelson *et al.* (2010). Their results indicated that the application of flexible transportation services or DRT services involve changes in business models, citizen behavior, and technology support. Diana *et al.* (2007) studied ways to organize a public transport service in order to reduce environmental impact. They concluded that emissions may be reduced by DRT services in low demand scenarios; they also suggested that smaller vehicles could be more suitable for this kind of service. Criteria for assessing flexible transport have been developed by Ferreira *et al.* (2007). Zografos *et al.* (2008) presented a methodology for developing Flexible Transport Systems (FTS), and described a related application in the City of Helsinki. Mulley and Nelson (2009) considered bus-based flexible public transport services, concluding that such services may revitalize bus public transport. Broome *et al.* (2012) evaluated the efficiency of providing FTS bus transport to the elderly. They concluded that demand may be doubled, while overall satisfaction may increase significantly. Finally, Nguyen-Hoang and Yeung (2010) assessed a flexible demand-responsive form of public transportation, they called paratransit. They concluded that the benefits of paratransit exceed the related costs.

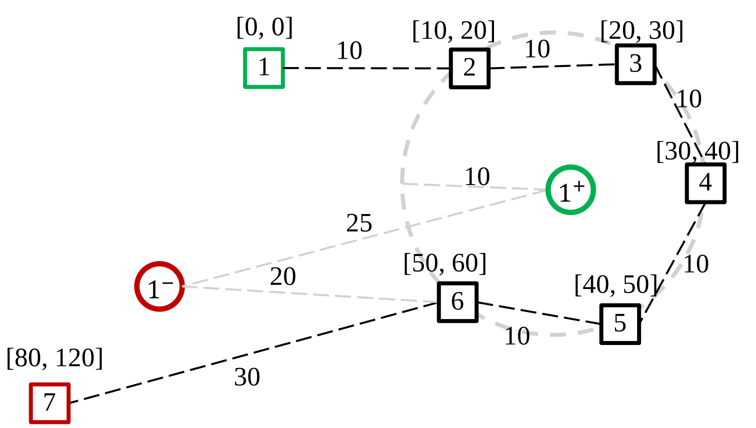
The Demand Adaptive System (DAS) of Daganzo (1984), Crainic *et al.* (2000, 2005), and Malucelli *et al.* (2001) is a special case of DRT and is related to the problem being addressed in this report. In their case, optional bus stops are defined and are activated in order to provide improved service. Crainic *et al.* (2000, 2001) presented the main aspects and the model of DAS. Malucelli *et al.* (2001) developed a meta-heuristic algorithm based on a tabu framework for the single route case. Crainic *et al.* (2005) developed and analyzed solution strategies by combining two general metaheuristic classes for the single route single vehicle problem.

Another case closely related to the one addressed in this report is the Mobility Allowance Shuttle Transit (MAST) services of Quadrifoglio *et al.* (2006, 2007, 2008), Quadrifoglio and Dessouky (2008). In MAST, a vehicle moves repeatedly from an origin point to a destination point and serves additional intermediate bus stops (checkpoints), while it is allowed to deviate within a certain range of the main route to serve additional requests. A latest departure time is defined for each check point and cannot be violated. In Quadrifoglio *et al.* (2008) a MILP formulation with additional logic constraints is developed and solved for the MAST. In this case, the requests are categorized with respect to their pick-up and drop-off points; all requests must be served while the objective function minimizes the weighted sum of: i) the total miles driven, ii) the total ride time of all customers, and iii) the total waiting time of all customers. Note that in order to serve all requests, clients may not picked-up/dropped-off at their desired location but at the nearest predefined checkpoint. The latter problem is addressed by a Branch and Cut framework in which several logical cuts are applied.

The problem addressed by the current work is the Demand Responsive Bus Routing Problem (DRBRP), which considers a semi-flexible public transport system, much like the one described at the beginning of the introduction. The system's route includes several predefined regular bus stops, which the vehicle visits in a prescribed sequence; the earliest and latest departure times of the bus from each regular bus stop are also defined. In our case we consider that the earliest departure time is equal to minimum travel time from the first regular bus stop. The latest departure time limits the deviation of the bus from the nominal route in order to maintain an adequate service level regarding the passengers waiting at the regular bus stops. In addition, there are flexible requests, which may or may not be served. Each request is considered to be known before the start of the trip and comprises a pick-up location, a drop-off location, the earliest and latest times that the client may be transported to the destination (drop-off location). The latest drop off time is defined by the client, while the earliest time is a result of the system's plan.

Each bus in the system has certain capacity for the flexible requests, i.e. for mobility impaired passengers. The problem consists of maximizing the total number of serviced paratransit (flexible) requests, while minimizing the ride time of the paratransit requests. The latter is a significant issue, since the ride time reflects the quality of transportation for the mobility impaired passengers.

The example of Figure 1 illustrates the necessity to consider the ride time of the paratransit requests. In this example the request for pick-up (1) may be served between bus stops 2, 3, 4, 5, 6 with almost the same deviation from the nominal route. However, the corresponding client will be offered the best service if s/he would be picked up after bus stop 6. This may be achieved by adding to the objective function a penalty term reflecting the total ride time of the entire schedule, as mentioned above.



Route	Ride Time
$1, 2, 1^+, 3, 4, 5, 6, 1^-, 7$	60
$1, 2, 3, 1^+, 4, 5, 6, 1^-, 7$	50
$1, 2, 3, 4, 1^+, 5, 6, 1^-, 7$	40
$1, 2, 3, 4, 5, 1^+, 6, 1^-, 7$	30
$1, 2, 3, 4, 5, 6, 1^+, 1^-, 7$	25

Figure 1: Why the ride time metric is needed: 1^+ request pick up, 1^- request drop off

The differences between DRBPR and the other demand responsive systems reviewed above are as follows: DAS considers a service in which optional bus stops may be activated or not, while in DRBRP we consider a door-to-door service. The MAST system is closer to the DRBRP case, since it also considers a door-to-door case. However, it differs from DRBRP in the following respects: a) MAST does not consider vehicle capacity constraints, b) all transportation requests must be served, and c) the vehicle departs form a stop at certain time. Furthermore, the approaches used to address the two problems are different; i.e. the former problems are addressed by metaheuristic methods or by exact Branch and Cut methods, while in the current work an exact decomposition technique is proposed.

Note that DRBRP may be viewed as a combination of the Pick-up and Delivery Problem (PDP) and the Team Orienteering Problem (TOP). Both routing problems have been extensively studied over the years. Interested readers may refer to Savelsbergh and Sol (1995), Desaulniers *et al.* (2002) and Cordeau and Laporte (2003) for extensive reviews concerning the PDP. Vansteenwegen *et al.* (2011) review the research conducted of the TOP.

In this report, we introduce a mathematical model that captures all aspects of the proposed semi-flexible bus service. An exact Branch and Price (BP) framework is implemented to solve problems of practical size to optimality. To solve the sub-problem of this framework we modify the labeling algorithm developed for the related Elementary Shortest Path Problem to meet the special constraints of the DRBRP. Furthermore, we study the design of a DRBRP system in three representative environments: urban, suburban, and rural. The design study focused on the tuning of the system parameters to suit the characteristics of each environment. Finally, we investigate the efficiency of the proposed exact method through extensive testing.

The remainder of this report is organized as follows: Section 2 presents the proposed mixed integer linear programming model for the DRBRP. Section 3 describes the proposed solution approach. Section 4 presents the design study and the computational experiments, while in Section 5 we summarize the conclusions of this study.

2 Mathematical Formulation of DRBRP

The DRBRP has been formulated as a linear mixed integer problem. We consider the following notation:

- $G(N, A)$ is the typical directed graph where N is the set of all nodes related to the problem and A is the set of arcs that connect all nodes; N will be defined next

- Let m be the number of the regular bus stops, and v be the total number of bus trips for a certain period, where $K = \{1, \ldots v\}$ is the set of bus trips
- A node is defined per predefined regular bus stop and per bus trip. Thus, let $B = \{1, \ldots, m \times v\}, B \subset N$ be the set of nodes that model the regular bus stops of the network; the sequence of serving these regular bus stops must be preserved
- Let $B_k = \{1 + m \times (k-1), .., m + m \times (k-1): k \in K\}$ be the set of regular bus stop nodes served during bus trip k
- Let n be the number of paratransit requests. In terms of nodes a paratransit request is characterized by its pick-up node i and its delivery node $i + n$
- Let $S = \{1 + m \times (k-1): k \in K\}$ be the set of nodes that correspond to the first regular bus stop of all bus trips
- Let $E = \{m + m \times (k-1): k \in K\}$ be the set of nodes that correspond to the last nodes of all bus trips
- Let vertices, $C = C^+ \cup C^-, C \subset N$ be the set of nodes that correspond to the paratransit requests. $C^+ = \{m \times v + 1, \ldots, m \times v + n\}$ contains the pick-up nodes of the paratransit requests. Nodes in $C^- = \{m \times v + n + 1, \ldots, m \times v + 2n\}$ are the delivery nodes of the paratransit requests
- Following the above definitions, the set of nodes N is defined as $N = \{1,2, \ldots, m \times v + 2n\}$
- $s_{ik}, i \in B \cup C, k \in K$ is the time that each node i is served during bus trip k
- $[a_i, b_i], i \in N$ is the time interval within which each node must be served
- $q_{ik}, i \in N, k \in K$ is the number of paratransit passengers on board during trip $k \in K$ immediately after node i is served
- $d_i, i \in N$ is the demand of paratransit passengers of each node. If it is a regular bus stop then $d_i = 0, i \in B$. If it is a pick-up node $d_i = 1, i \in C^+$. If it is a delivery node $d_i = -1, i \in C^-$. In case the pick up or the delivery locations of a paratransit request coincide with a regular bus stop, then a new node is established at the same location
- $t_{ij}, i, j \in N$ is the time required to travel from node i to node j
- $x_{ijk}, i, j \in N, k \in K$ is assigned the value 1 if the arc from node i to node j is traversed during bus trip k, otherwise 0 is assigned to the variable
- RT_{ik}, is the total ride time of paratransit request $i \in C^+$ in bus trip k

- Q, is the maximum number of paratransit passengers that any vehicle $k \in K$ may carry. Note the only the capacity of the bus in paratransit passengers is of concern, since the regular bus services are assumed to be designed in a way which balances demand and capacity.
- $s_{11} = 0, q_{1k} = 0, k \in K$ and $q_{mk} = 0$.

The objective function of the problem is to maximize the number of paratransit requests served while minimizing the total ride time of the served paratransit requests; it is defined as follows:

$$TP = max \sum_{k \in K} \sum_{i \in C^+} \sum_{j \in N} x_{ijk} - P \sum_{k \in K} \sum_{i \in C^+} RT_{ik} \tag{1}$$

- The value of the penalty constant is given by $P = 1/(n \times T_{max})$, where $T_{max} = b_m - a_1$ is the maximum time of a bus trip. This value guarantees that the last term in Eq. (1) is less than 1, and, thus, the maximum possible number of requests will be served irrespective of the savings in total ride time.

Subject to:

$$\sum_{j \in C \cup B_k} x_{ijk} = 1, \quad i \in B_k \backslash E, k \in K \tag{2}$$

$$\sum_{i \in N} x_{ijk} = 1, \quad j \in E, k \in K \tag{3}$$

$$\sum_{k \in K} \sum_{j \in N} x_{ijk} \leq 1, \quad i \in C^+ \tag{4}$$

$$\sum_{i \in N} x_{ihk} - \sum_{j \in N} x_{hjk} = 0, \quad h \in N \backslash (S \cup E), k \in K \tag{5}$$

$$\sum_{i \in N} x_{ihk} - \sum_{j \in N} x_{j,h+n,k} = 0, \quad h \in C^+, k \in K \tag{6}$$

$$s_{i-1,k} \leq s_{ik}, \quad i \in B \backslash S, k \in K \tag{7}$$

$$s_{ik} + t_{i,i+n} \sum_{j \in N} x_{ijk} \leq s_{i+n,k}, \quad i \in C^+, k \in K \tag{8}$$

$$s_{ik} + t_{ij} - M\left(1 - x_{ijk}\right) \leq s_{jk}, \quad i \in N, j \in N, k \in K \tag{9}$$

$$s_{i+n,k} - s_{ik} = RT_{ik}, \quad i \in C^+, k \in K \tag{10}$$

$$a_i \sum_{j \in N} x_{ijk} \leq s_{ik} \leq b_i \sum_{j \in N} x_{ijk}, \qquad i \in N \backslash E, k \in K \tag{11}$$

$$a_i \leq s_{ik} \leq b_i, \quad i \in E, k \in K \tag{12}$$

$$q_{ik} + d_j - M(1 - x_{ijk}) \leq q_{jk}, \quad i \in N, j \in N, k \in K \tag{13}$$

$$0 \leq q_{ih} \leq Q \sum_{k \in K} \sum_{j \in N} x_{ijk}, \quad i \in N \backslash S \cup E, h \in K \tag{14}$$

$$x_{ijk} \in \{0,1\}, \quad i, j \in N, k \in K \tag{15}$$

$$s_{ik} \geq 0, \quad i \in N, k \in K \tag{16}$$

$$d_i \in \{-1,0,1\}, \quad i \in N \tag{17}$$

Constraints (2) and (3) ensure that all regular bus stops will be served once per bus trip. Constraint (4) ensures that the paratransit pick-up nodes (and thus the respective paratransit requests) may be served once. Equation (5) ensures continuity for all nodes except those corresponding to the starting and ending regular bus stops of each bus trip. Equation (6) ensures that if a paratransit request is served, then the corresponding pick-up and delivery nodes will be served in the same period and during the same trip. Constraint (7) ensures that regular bus stops will be served with the predefined sequence. Constraint (8) ensures that paratransit delivery nodes will be served after their related pick-up nodes. Constraint (9) defines the time flow, assuming that $M \gg 1$. Constraint (10) defines the ride time of paratransit request $i \in C^+$. Note that in case request i is not served $RT_{ik} = 0$ due to Constraint (9), then no penalty is added to the objective function. Constraints (11) and (12) ensure that if a node is served, this will occur in the predefined time interval $[a_i, b_i]$. Constraint (13) defines the change in the paratransit load of the bus. Constraint (14) ensures that the paratransit bus load is greater than zero and less than or equal to Q. Finally, Constraint (15) defines the binary nature of decision variables x_{ijk}.

Concerning the complexity of the above model, consider that according to Golden *et al.* (1987) the Orienteering Problem (OP) belongs to the NP-hard class of problems by reducing OP to the Traveling salesman problem. Laporte and Marcelo (1990) also proved that OP belongs to the class of NP-hard problems by reducing it to the Hamiltonian circuit problem. By applying the same reduction of Golden *et al.* (1987), one may deduce that an orienting problem with pick-ups and deliveries is in the same class of complexity as the Pick-up and Delivery Problem (PDP). Specifically, setting the time limit of an Orienteering Problem with Pick-up and Deliveries (OP-PDP) to the exact time needed to serve all customers with the minimum travelling time, the solution of the PDP and the OP-PDP will be the same, and thus, the two problems are of the same complexity. Hence, since the PDP is an NP-hard problem, then the OP-PDP belongs to the same problem class in terms of complexity. DRBRP may be also reduced to an OP-PDP by considering a problem case in which only two regular bus stops (e.g. the first, the last) represent the origin and destination depot. Thus, the DRBRP also belongs to the same complexity class as OP-PDP and is NP-hard.

3 A Branch and Price approach for the DRBRP problem

In a Branch and Price framework a problem is solved using the column generation method, by relaxing the integrality constraints of the original problem, and solving the resulting large linear program within a branch and bound tree (Desaulniers *et al*., 2005). Column generation comprises two phases that are applied in an iterative manner. The first phase solves the Restricted Master Problem (RMP), which contains a subset of all feasible variables. The second phase solves one or more Sub-Problems (SP), generating additional variables to be added to the RMP. The procedure ends when no more "profitable" variables can be generated by the sub-problem(s). In routing problems, the RMP is usually formulated as a set partitioning problem, and the sub-problem is formulated as a resource constraint shortest path problem. We apply the same principles here.

3.1 The Set Partitioning Problem for the DRBRP

In order to formulate the problem presented in Section 2 as a set partitioning problem, let δ_k be a set that contains all feasible routes of a single bus trip k, and let $\delta = \delta_1 \cup \delta_2 \cup \ldots \cup \delta_v$ be the set of all possible feasible routes of all bus trips. Each route in δ_k must satisfy constraints (2)-(16) by setting $K = \{k\}$. For each route $j \in \delta$ define the set A_j of the paratransit requests served by route j. a_{ij} is a binary coefficient that takes the value 1 if $i \in A_j$. Let rt_{ij} be the ride time of request i served by route j, which is similar but not identical to RT_{ik} of the formulation of Section 2; the latter refers to bus trip k and not to route j. Then the profit p_j of route j can be calculated as follows:

$$p_j = \sum_{i \in A_j} a_{ij} - P \sum_{i \in A_j} rt_{ij} \tag{18}$$

where P is the penalty of Eq. (1). Let y_j be the binary variable which is assigned the value 1 if route j is used, and 0 if not. Then the problem may be formulated by the objective function:

$$max \sum_{j \in \delta} p_j y_j \tag{19}$$

Subject to:

$$\sum_{j\in\delta} a_{ij} y_j \leq 1, \quad i \in C^+ \tag{20}$$

$$\sum_{j\in\delta_k} y_j = 1, \quad k \in K \tag{21}$$

$$y_j \in \{0,1\}, \quad j \in \delta \tag{22}$$

Constraint (20) ensures that all paratransit requests (pick-up nodes) may be served at most once. Constraint (21) ensures that one route must be used per bus trip; thus all regular bus stops will be served once during each bus trip.

3.2 The Sub-Problem for the DRBRP

In the case under consideration one may define $|K|$ sub-problems (SP), one for each bus trip $k \in K$. Each SP will produce additional routes to be added to $\delta_k' \subseteq \delta_k$. The objective function of the SP is to identify the route with the maximum reduced cost of the problem described in the previous section, however it is more efficient to identify more than one routes with positive reduced cost; these are designated as the routes j satisfying the following condition:

$$p_j - \sum_{i\in A_j} a_{ij}\pi_i - \bar{\pi}_k > 0 \tag{23}$$

where π_i are the dual prices related to constraint set (20) and $\bar{\pi}_k$ are the dual prices related to constraints (21). At the same time, these routes should satisfy constraints (2) - (16) by setting $K = \{k\}$.

Below we simplify the notation for the SP, which from now on will refer to one bus trip $k \in K$, as follows: Let $\bar{N} = B_k \cup C^+ \cup C^-$ be the set of all nodes. Its subset B_k contains the nodes that correspond to the regular bus stops of bus trip k, while $C^+$and C^- are also sub-sets of $\bar{N}$ and correspond to the pick-up and drop-off locations of the paratransit requests, respectively.

The SP may be considered as an Elementary Shortest Path Problem with Time Windows, Capacity, Pick-up and Delivery as well as Sequence Restrictions (ESPPTWCPDSR). To solve such resource constraint routing problems, labeling algorithms are commonly applied; see Feillet *et al.* (2004), Irnich and Desauliners (2005). In fact, this is the critical issue in Column Generation techniques, since the performance of the labeling algorithm affects directly the efficiency of solving the relaxed problem, and thus, the size of problems that may be solved to optimality. In the following Section, we extend the labeling algorithm presented in Ropke *et al.* (2009) for the ESPPTWCPD to adapt to the special characteristics of the ESPPTWCPDSR and we present a new structure for the labeling algorithm to solve it.

3.3 Labeling Algorithm for the ESPPTWCPDSR

The proposed approach to solve the ESPPTWCPDSR adapts the algorithm of Ropke *et al.* (2009) for solving the Pick-up and Delivery problems with Time Windows. Ropke *et al.* had, in turn, modified the original labeling algorithm proposed by Feillet *et al.* (2004) for ESPPTW. The proposed algorithm for the ESPPTWCPDSR starts from the first regular bus stop and progressively extends all feasible paths until they reach the last regular bus stop. Dominance rules are applied during the latter procedure and paths of inferior quality are discarded without extending them.

Let L be a label that contains information about the consumption of resources along the related path. $L = [ln, pbs, t, rt, h, p, V, O]$, where:

- ln: is the last node of the path
- pbs: is the last regular bus stop visited by the path
- t: is the arrival time at node ln
- rt: is the total ride time of all paratransit requests belonging to the path up to this point
- h: is the number of paratransit passengers on the bus
- p: is the current profit corresponding to the path up to this point
- $V \subseteq \{1, \dots, n\}$: is the set of paratransit requests for which the passenger has been picked up (but possibly not dropped off)
- $O \subseteq \{1, \dots, n\}$: is the set of those paratransit requests for which the passenger has been picked up but not dropped off. Requests O are said to be *open* or *active*

Considering the extension of a label $l = [ln, pbs, t, rt, h, p, V, O]$ to node j one of the following conditions must hold:

$$m < j \leq n \wedge V(l) \cap \{j\} = \emptyset \tag{24}$$

$$n < j \leq 2n \wedge j \in O(l) \tag{25}$$

$$j < m \wedge pbs(l) = j - 1 \tag{26}$$

$$j = m \wedge O(l) = \emptyset \wedge pbs(l) = j - 1 \tag{27}$$

where m is the number of regular bus stops and n is the number of paratransit requests. Conditions (24) to (26) are similar to those used in the ESPPTWCPD (Ropke *et al.*, 2009). Condition (24) ensures that in case j is a pick-up node it will be added to the path only if the node has not been added previously. In case j is a delivery node, it will be added to the path only if its corresponding pick-up node has been visited previously (25). Condition (26) defines that a regular bus stop is added to the path only if it is the next one according to the predefined sequence; it also ensures that a regular bus stop is added to the path only if it has not been served previously. The final destination node, which is the last regular bus stop of the sequence, will be added to the path only if no open request exists and all other regular bus stops are served, as defined by Condition (27).

If the aforementioned conditions hold, the information of label l_{new} of the path extended to node j is computed as follows:

$$ln(l_{new}) = j \tag{28}$$

$$pbs(l_{new}) = \begin{cases} j, & j \in B_k \\ pbs(l), & j \in C^+ \cup C^- \end{cases} \tag{29}$$

$$t(l_{new}) = \max(t(l) + t_{n(l),j}, a_j) \tag{30}$$

$$rt(l_{new}) = rt(l) + |O(l)| \times t_{n(l),j} \tag{31}$$

$$h(l_{new}) = h(l) + q_j \tag{32}$$

$$p(l_{new}) = \begin{cases} p(l), & j \in B_k \\ p(l) + \bar{p}_{j,} & j \in C^+ \\ p(l) - P \times rt_{j,} & j \in C^- \end{cases} \tag{33}$$

$$V(l_{new}) = \begin{cases} V(l) \cup \{j\}, & j \in C^+ \\ V(l), & j \in C^- \cup B_k \end{cases} \tag{34}$$

$$O(l_{new}) = \begin{cases} O(l), & j \in B_k \\ O(l) \cup \{j\}, & j \in C^+ \\ O(l) \backslash \{j - n\}, & j \in C^- \end{cases} \tag{35}$$

Label l_{new} will be discarded if $h(l_{new}) > Q$ or $t(l_{new}) > b_j$, which are the constraints regarding the capacity of the vehicle and the latest time of arrival.

In order to discard dominated labels, dominance rules are examined between pairs of labels ending to the same node. These rules should ensure that all extensions of the dominated labels do not lead to a better solution than the one of the dominant labels. A label l_1 dominates a label l_2 if all criteria defined by Eqs. (36) to (41) are satisfied:

$$ln(l_1) = ln(l_2) \tag{36}$$

$$pbs(l_1) = pbs(l_2) \tag{37}$$

$$t(l_1) \leq t(l_2) \tag{38}$$

$$p(l_1) \geq p(l_2) \tag{39}$$

$$V(l_1) \subseteq V(l_2) \tag{40}$$

$$O(l_1) \subseteq O(l_2) \tag{41}$$

$$rt(l_1) \leq rt(l_2) \tag{42}$$

Criteria (36), (38), (40) and (41) have been used in case of the ESPPTWCPD (Ropke *et al.*, 2009). We have introduced criteria (37), (39), and (42) to address ESPPTWCPDSR. To show that the dominance rules guarantee optimality, one must show that a) the possible extensions of the dominant label form a superset of all possible extensions of the dominated one; b) if the dominant and dominated labels extend to the same node, then the dominant one would lead to lower or equal resource consumption.

Statement (a) above is guaranteed if the same amounts of resources are available for use for both labels when these are extended. Consider that both labels concern the same node i (Eq. 36). Then if both labels l_1 and l_2 are extended to node j, according to Eq. (38) if l_2 is feasible, l_1 will also be feasible, given that the time to traverse arc (i, j) is the same for both cases. The same consideration can be made for cumulative profit and the total ride time, i.e. resources p and rt, respectively.

These latter considerations also safeguard that statement (b) is valid. It is noted that the same nodes are available when extending either of the two paths, since $V(l_1)$ is a subset of $V(l_2)$and due to Conditions (24) to (26), which ensure that all nodes will be visited once. Similarly, due to Condition (25), drop-off nodes are available for extension only if the respective pick-up node belongs to the path. In ESPPTWCPDSR, regular bus stops must be served following a certain sequence, which is expressed by Condition (27). l_1 will be able to be extended to the same regular bus stop as l_2, only if Eq. (37) is satisfied.

Algorithm 1 presents the pseudo-code of the proposed labeling algorithm for the ESPPTWCPDSR; we call this algorithm “Bus Stop Structure” (BSS). BSS capitalizes on the fact that: a) a path is feasible only if the bus stop sequence is preserved, and b) paths can be dominated only by paths that have reached the same regular bus stop. BSS solves ESPPTWCPDSR by extending all possible paths until the next bust stop and only then discards labels due to dominance criteria (38) to (42), since criteria (36) and (37) are already satisfied. Furthermore, if an extension to the next regular bus stop of the sequence is not feasible, then the label is also discarded and no new label is generated.

Algorithm 1: Labeling Algorithm – Bus Stop Structure

```
Initialize lists of labels L, L_new, L_final, L_nb
Add first label to L_nb
While L_nb ≠ ∅
  While L ≠ ∅
    l = remove(L)
    Extend l to next regular bus stop
    If extension is feasible
      If final bus stop
        Add new label to L_final
      else
        Add new label to L_nb
        Add one new label for every feasible pick-up node to L_new
        Add one new label for every delivery node to L_new
  Remove dominated labels form L_new
  L = L_nb, L_nb = ∅
Return path corresponding to the best label in L_final
```

3.4 Branch and bound

Given that column generation does not guarantee the integer solution of the problem, in our approach we implement a typical branch and bound scheme. That is, if the solution provided by the column generation algorithm contains a variable with a non-integer value, then two new branches are created, each corresponding to a problem that uses a different integer value for this variable. Column generation is applied to solve the problems created in each of these branches. In our case we branch on flow variables based on the value of the variable x_{ijk}. These new branches will include constraints $x_{ijk} = 1$ and $x_{ijk} = 0$ respectively. The variable x_{ijk} that is closest to 0.5 is selected for branching. Interested readers may refer to (Desaulniers *et al.*, 2005) for alternative branching strategies.

4 Tests and results

In this Section the performance of the proposed Branch and Price approach is evaluated first with respect to related algorithms from the literature. Subsequently, the effects of key constraints on the system's performance are investigated. Finally, the transport system is studied under conditions that characterize three distinct environments: a) Urban, b) suburban, and c) rural. Based on this study, design guidelines that concern the tuning of key system parameters have been developed.

A random test data generator has been developed to provide test instances that simulate practical. The data generator uses special techniques to produce test instances that incorporate the characteristics of each of the three aforementioned environments. The basic input parameters of the generator are the following:

a) m the number of regular bus stops
b) k the number of the scheduled bus trips
c) n the number of paratransit requests
d) ρ the average distance between two sequential regular bus stops
e) u the average vehicle speed
f) f the time period between two consecutive bus trips (headway)
g) d the coefficient of the allowed delay at each bus stop w.r.t. the nominal bus trip e.g. a bus may visit the latter at most $d \times \rho$ after the designated earliest time of arrival to it. Note that d is used to define the latest arrival time at each regular bus stop, and that the related allowable delay is cumulative for the entire trip
h) ξ the width of the drop-off time windows for the paratransit requests

Another significant input of the generator is the allowable region in which the pick-up and drop off locations of the paratransit requests may be generated along the service route. The generator considers two cases of allowable regions: ellipses, and circles. For the former case an ellipse is defined considering two sequential regular bus stops as foci and ω as the length of its major axis. For the latter case, a circle is defined considering the location of a regular bus stop as the center and $\overline{\omega}$ as its radius. The exact generation of the test instances is further described in Dikas and Minis (2013).

4.1 Computational study I: Algorithm efficiency

A range of experiments have been conducted in order to evaluate the efficiency of the proposed BSS algorithm of Section 3. This investigation compares BSS with an algorithm that solves the ESPPTWCPDSR by following the steps of the labeling algorithm proposed in the literature to solve ESPPTWCPD (Ropke *et al.*, 2009). Hereafter, this latter algorithm will be called SS. The experiments investigate the effects of key parameters on: a) required computational time and b) the size of problems each algorithm is able to solve. The parameters studied and the corresponding values are the following:

- $d \in \{0.2, 0.5, 0.8\}$: the maximum allowable delay coefficient
- $m \in \{10, 20\}$: the number of regular bus stops
- $n \in \{10, 15, 20\}$: the number of paratransit requests
- $k \in \{10, 20\}$: the number of bus trips

The pick-up and drop-off locations of paratransit requests were generated assuming the case of the urban environment (see Dikas and Minis (2013) and Section 4.3), in which these locations are distributed along the route uniformly within a corridor created by overlapping ellipses.

The number of experiments generated by the above parameter combinations is $3 \times 2 \times 3 \times 2 = 36$, and each randomly generated experiment was solved five times (36 experiments $\times$ 5 repetitions/experiment =180 instances in total) with the two methods.

Table1 presents the average computational time over the five repetitions per experiment resulting from the two solution algorithms. It also provides for each algorithm the number of times an experiment did not reach a solution within a run time limit of 2 hours on a PC equipped with an Intel Xeon E5335 CPU, and 8 GB of RAM.

Table 1: Comparison of BSS and SS performance

			d=0.2					**d=0.5**					**d=0.8**				
			CPU[1]			**Failures[2]**		**CPU[1]**			**Failures[2]**		**CPU[1]**			**Failures[2]**	
k	**m**	**n**	**BSS**	**SS**	**%Sav.[3]**	**BSS**	**SS**	**BSS**	**SS**	**%Sav.[3]**	**BSS**	**SS**	**BSS**	**SS**	**%Sav.[3]**	**BSS**	**SS**
10	10	10	2.3	2.9	23.4%	-	-	33.3	87.1	61.7%	-	-	111.3	316.1	64.8%	-	-
		15	21.2	33.9	37.5%	-	-	334.6	667.4	49.9%	-	-	1827.2	2132.8	14.3%	-	1
		20	263.7	394.7	33.2%	-	-	1071.0	1388.3	22.9%	2	3	-	-	-	5	5
	20	10	14.0	20.1	30.6%	-	-	43.2	131.7	67.2%	-	-	111.7	418.1	73.3%	-	-
		15	63.2	204.7	69.1%	-	-	501.5	1151.1	56.4%	-	1	1281.3	4821.6	73.4%	-	2
		20	145.7	198.7	26.7%	-	-	1296.9	1085.8	-19.4%	2	3	-	-	-	5	5
20	10	10	2.7	3.5	22.0%	-	-	53.5	116.8	54.2%	-	-	232.9	550.8	57.7%	-	-

	d=0.2					d=0.5					d=0.8				
	CPU[1]			**Failures**[2]		**CPU**[1]			**Failures**[2]		**CPU**[1]			**Failures**[2]	
k m n	**BSS**	**SS**	**%Sav.**[3]	**BSS**	**SS**	**BSS**	**SS**	**%Sav.**[3]	**BSS**	**SS**	**BSS**	**SS**	**%Sav.**[3]	**BSS**	**SS**
15	69.0	98.6	30.0%	-	-	974.6	2451.8	60.3%	-	-	920.6	1241.1	25.8%	2	3
20	305.3	648.5	52.9%	-	-	650.7	1478.5	56.0%	1	4	-	-	-	5	5
10	23.9	27.1	11.8%	-	-	107.3	312.1	65.6%	-	-	422.9	2106.2	79.9%	-	-
20 15	124.9	232.8	46.3%	-	-	1448.7	4809.1	69.9%	-	2	-	-	-	3	5
20	611.4	1002.5	39.0%	-	-	-	-	-	4	5	-	-	-	5	5
Average	**137.3**	**239.0**	**42.6%**	**-**	**-**	**592.3**	**1243.6**	**52.4%**	**-**	**-**	**701.1**	**1655.3**	**57.6%**	**-**	**-**
Sum	**-**	**-**	**-**	**0**	**0**	**-**	**-**	**-**	**9**	**18**	**-**	**-**	**-**	**25**	**31**

[1] Average computational time in seconds for problem instances that both algorithms solved, [2] Number of cases not solved, [3]Present of savings of BSS CPU over SS CPU

The results indicate that in almost all cases BSS obtained the optimal solution in less computational time than SS; on the average BSS required about half the computational time of SS. Furthermore, BSS managed to solve more problem instances, an indication that BSS can deal with problems of larger scale.

4.2 Computational study II: Effect of the problem's constraints

We investigate the effect of significant SPTS constraints in the efficiency of the problem's solutions. For this study, the number of regular bus stops was set to $m = 10$, the number of bus trips to $k = 5$, and the pick-up and drop-off locations of the paratransit requests were generated assuming the urban case.

As a first step, the study considers the case in which the capacity of the vehicle is not constrained, and there are no constraints regarding the earliest or latest drop-off times of the paratransit requests. For this case, Figure 2 presents the percent of requests served and the service level and for $d = \{0.1, 0.3, 1\}$, and for n increasing from 1 to 8 requests. The service level value is provided as the percentage of the maximum possible value of the objective function; the latter is obtained when serving all paratransit requests and the total ride time is zero (clearly an impossibility). As d increases the service level also increases. For $d = 1$ all paratransit requests are served.

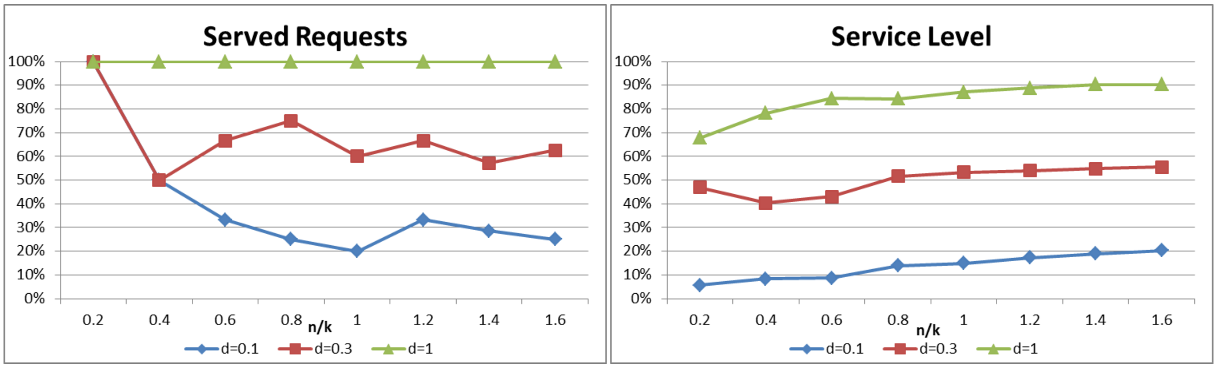


Figure 2: Service performance of SPTS for three different cases of d and assuming no other constraint

Consider now the constraint for the vehicle capacity Q regarding paratransit requests. Figure 3 presents the proportion of requests served and the service level for $d = 1$. In this case also the x axis is the number of paratransit requests per trip (n/k). According to this Figure, low values of Q decrease the service level provided by the system, especially with increasing n/k. However, even with only two reserved seats, the efficiency of the system seems to be sufficient and an increase in capacity leads to a major improvement in system performance. Note that patterns observed in this Figure are due to the specific restrictions imposed on the system (e.g. not possible to serve all requests).

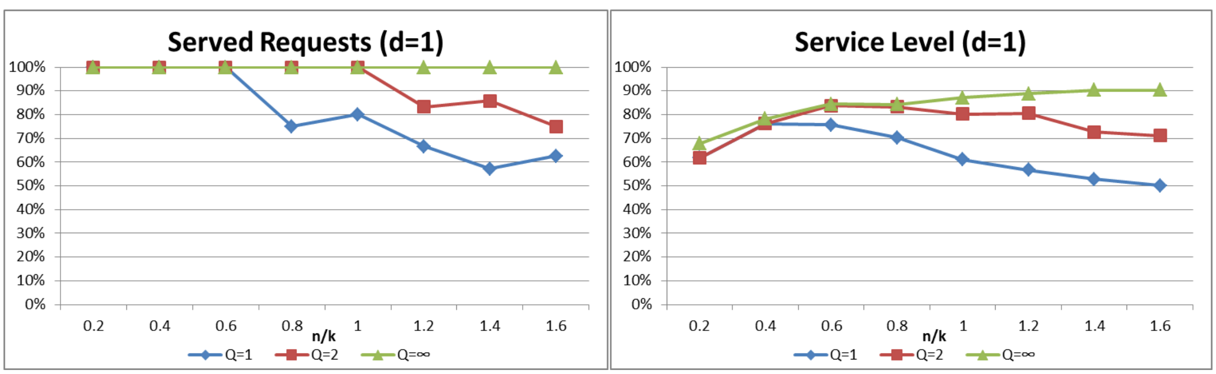


Figure 3: Service performance of SPTS for three different cases of Q, assuming that d=1 and no other constraint

To investigate the effect of the time window constraint for the paratransit requests, consider the case of Figure 4, for which $Q = 2$ and $d = 1$. This Figure illustrates the efficiency of the system for the following three cases of time window width (where f is the headway between sequential bus trips):

- **Tight**: $\xi = b_j - a_j = 1 \cdot f$

- **Loose**: $\xi = b_j - a_j = 3 \cdot f$

- **∞:** No time windows constraints applied

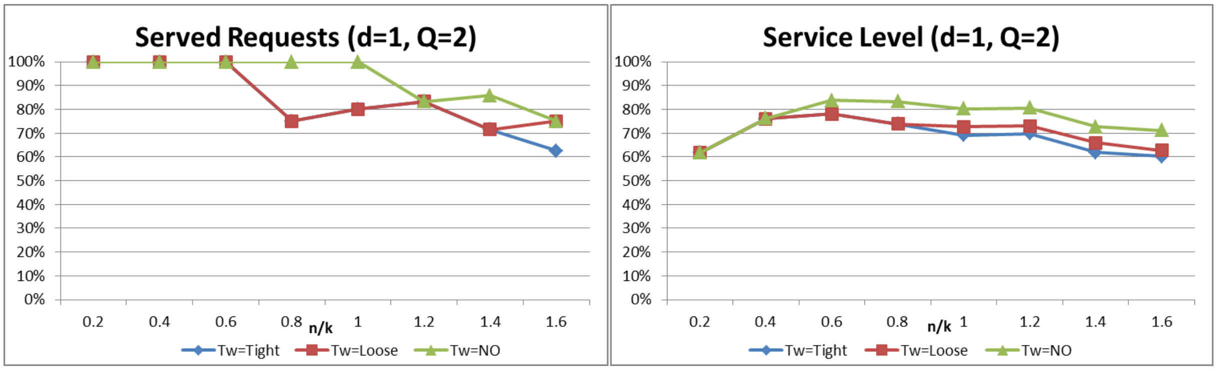


Figure 4: Service performance of SPTS for three different cases of time windows width, assuming that d=1 and Q=2

In this case, the effect of time windows does not appear to be significant, since constraints with respect to d and Q have already affected the performance of the system. Overall, the significant constraints appear to have the following effects on system performance:

- As d increases the service level increases
- Low values of Q decrease the service level
- Narrow time windows reduce the service level.

4.3 Computational study III: SPTS in various transport environments

The analysis of Section 4.2 indicates the significant dependence of the performance of SPTS on certain parameters of the transport environment. This is further investigated below by considering cases of practical significance in three application environments: urban, suburban, rural. In all cases we consider a planning horizon of eight hours. Other key characteristics of these environments are given in Table 2 and discussed further in Dikas and Minis (2013).

Table 2: Parameters settings per application environment

Parameters		Environment		
Description	**Symbol**	**Urban**	**Sub-urban**	**Rural**
Average distance between sequential regular bus stops	$\rho\ (m)$	500	2000	10000
Headway between bus trips	$f\ (mins)$	10	30	60
Total number of bus trips in the planning horizon period	k	48	16	8
Average vehicle speed	$u\ (km/hr)$	25	30	50
Number of regular bus stops	m	20	20	5
Time window width of the paratransit requests	$\xi\ (mins)$	60	60	60

Distribution of pick-up locations of paratransit requests w.r.t the regular bus stops		Uniform in ellipse, Uniform along route	Uniform in ellipse, Higher at the first and third quarters of the route	Uniform in circle, Uniform along route
Distribution of drop-off locations of paratransit requests w.r.t the regular bus stops	Drop-off Demand Distribution	Same as above	Uniform in ellipse, Higher at the second and fourth quarter of the route	Same as above
Ellipse major axis	ω (m)	500	1200	-
Circle radius	ϖ (m)	-	-	4000

Table 3 shows the values of the system attributes of SPTS when operating in the above application environments. The values of these attributes result from the parameters of Table 2. Figure 5 presents the performance of the system for $n = 30$ and $Q = 1$ for the three cases under consideration.

In the urban environment case, adequate service level is achieved for values of $d > 0.6$. This indicates that SPTS is applicable in urban cases, in which limited delay is expected, provided there is adequate capacity to serve the paratransit requests (in this example $Q/(n/k) = 1.6$), and wide time windows for such requests (in this example $\xi/f = 6$).

Table 3: Key SPTS characteristics per application environment

Key SPTS Characteristics	Characteristic	Environment		
Description		Urban	Sub-urban	Rural
Requests per trip	n/k	0.63	1.88	3.75
Available seats per request per trip	$Q/(n/k)$	1.6	0.53	0.27
Possible trips within the time window of the request	ξ/f	6	2	1

For the suburban environment case, the system displays good service levels for even lower values of d (i.e. $d > 0.3$). The latter seems to be due to the ratio ω/ρ which assumes a lower value compared to the one of the urban case. However, there is no significant improvement to the performance for values of $d > 0.4$. This may be attributed to the fact that, even if there is enough time to serve the requests, these requests compete for the same bus seat, which is related to parameter $Q/(n/k)$ which in this case is equal to 0.53.

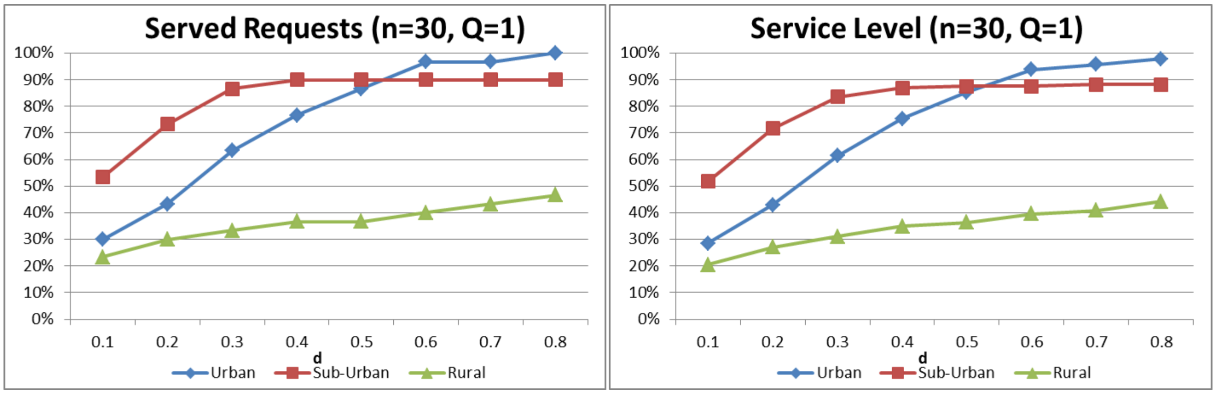


Figure 5: Service level for $n = 30$ and Q=1 for the cases of the three different environments

In the rural environment example, it seems that the system provides insufficient service level for the range values of d examined. This may be attributed to the fact that the ratio of paratransit requests per trip is high $n/k \cong 3.75$ and the ratio of reserved seats per paratransit request per trip $Q/(n/k) \cong 0.27$ is low indicating that there is less than $1/3$ seats available per request per trip. Another contributing factor may be the narrow time windows for such requests with respect to the headway between bus trips $\xi/f = 1$.

Figure 6 illustrates three additional cases for the rural environment in which the value of $Q/(n/k)$is doubled, the value of ξ/f is doubled and both values are doubled, respectively. In all cases, the system performance is improved as expected. The improvement resulting from enhancing the paratransit capacity is more pronounced than the improvement resulting from relaxing the time windows of the paratransit requests. The system with enhanced capacity and relaxed time windows is clearly the best performing, as expected.

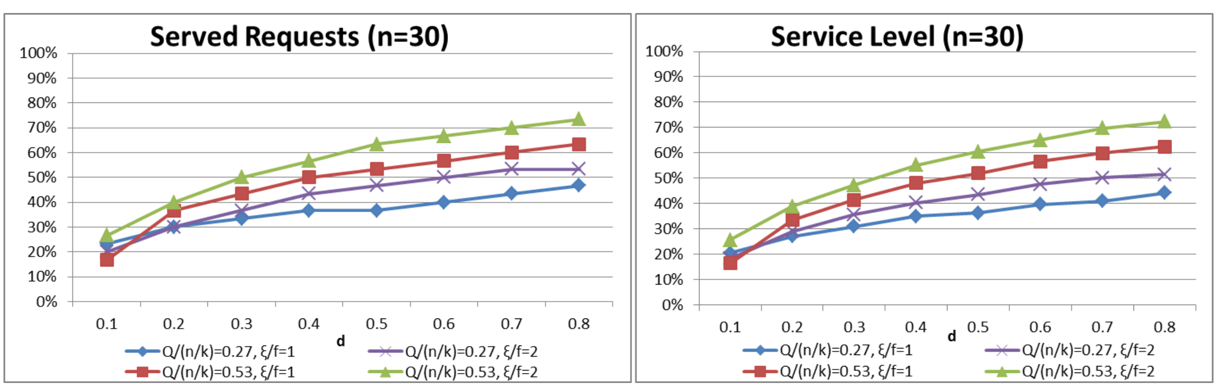


Figure 6: Service level for $n = 30$ and for different cases of the rural environment

From the above considerations, it is concluded that the SPTS designer should consider the following key attributes:

- The level of disruption of the nominal route as defined by the maximum allowable delay coefficient (d)
- The number of available seats per paratransit request per trip ($Q/(n/k)$), and
- the width of the time widow over the service headway (ξ/f).

4.4 System design considerations

Guided by the results of the above study, one may consider a systematic procedure for designing a scheduled paratransit transport system. In the first step of this procedure, the service headway f is determined in a straightforward manner. If the demand for non-paratransit passengers is h passengers per hour per direction and the vehicle stock's capacity is C passengers, then the number of trips per hour per direction are $k = h/C\, hr^{-1}$, or the headway is $f = 1/k\, hr$. Considering the ratio w of passengers that need assistance, then $n = h \cdot w$ is the hourly demand for paratransit passengers per direction, and n/k are the paratransit passengers per trip.

Subsequently, based on the desired service level and using an analysis similar to the one in Section 4.2, the basic SPTS parameters could be defined; that is, a) the maximum allowable delay of the nominal route d, b) the number of available seats $Q/(n/k)$ per paratransit request per trip, and c) the service level in terms of the width of the time window ξ/f over the service headway. At this stage, these latter two parameters are defined through Q and ξ, respectively.

For example, Figure 7 illustrates the selection of the lower value of parameter d in order to serve more than 75% of the paratransit requests in a case within an urban environment with $h = 500\, pax \cdot hr^{-1}, C = 50\, pax$, $k = 10$ and $n/k = 2$. The four curves are related to four combinations of (Q, ξ) values. As expected, in all parameter pair cases the performance of the system improves as parameter d increases. An interesting parameter setting is $Q = 2, \xi = 6 \cdot f$ and $d = 0.4$, which achieves the lowest disruption to the nominal route. Under the same paratransit capacity $Q = 2$, if one improves the waiting times for the paratransit customers to $\xi = 2 \cdot f$, then d should be increased to $d = 0.6$ to provide similar level of served requests.

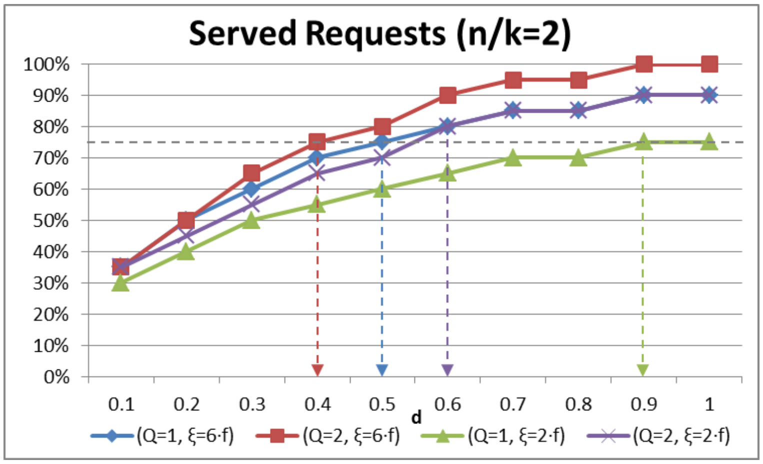


Figure 7: Ratio of served requests provided by the SPTS for different parameters cases assuming that $n/k = 2$

5 Conclusions

In this report a flexible surface transport service is studied, in which a bus may diverge from its nominal path to pick-up passengers with limited mobility and drop them off at their destination. Such a system may offer higher quality of transport to customers with a disability without increasing significantly public transport expenditure.

The current report has defined the corresponding problem by maximizing the number of served paratransit requests, reducing the total ride time of the paratransit customers, while serving the nominal route at a pre-defined level. To solve this problem we proposed an exact Branch and Price framework, which includes a labeling algorithm with appropriate dominance rules in order to reach the optimal solution within reasonable computational times. The proposed algorithm improves the computational time by up to 50% with respect to existing approaches for similar problems.

We used this algorithm to a) study the effect of key problem constraints on the service level, and b) support the design of SPTS systems taking into account the characteristics of different transport environments. This study indicated that three system parameters should be considered: a) the allowable level d of disruption of the nominal route, b) the available seats per paratransit request per trip $Q/(n/k)$, and c) the ratio of the time widow related to the requests over the service headway ξ/f. The values of these parameters must be high enough to ensure a good service level, but within reasonable limits in order to avoid major disruptions of the regular service, or high system costs.

Different considerations apply to different environments due to their intrinsic characteristics. For example, in an urban environment, which is characterized by high service rates, the values of $Q/(n/k)$, and ξ/f are relatively high, even for one reserved seat per trip. In this case, the system could be applied efficiently by allowing relatively low disruption to the nominal route (low values of d). In a suburban environment, with moderate service rates and longer distances between regular bus stops, the system could perform sufficiently well for a lower value of allowable delay (d). Finally, in a rural environment, in which ξ/f is relatively low, the values of $Q/(n/k)$ and d need to be higher.

Acknowledgments

This research has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.

References

Brake, J., Mulley, C., Nelson, J. D., Wright, S., 2007. Key lessons learned from recent experience with Flexible Transport Services. *Transport Policy 14*(6), 458-566.

Broome, K., Worrall, L., Fleming, J., Boldy, D., 2012. Evaluation of flexible route bus transport for older people. *Transport Policy 21*, 85-91.

Cordeau, J.-F., Laporte, G., 2003. The Dial-a-Ride Problem (DARP): Variants, modeling issues and algorithms. *4OR*, 89-101.

Crainic, T. G., Malucelli, F., Nonato, M., 2000. A demand responsive feeder bus system. *7th World Congress Intelligent Transport Systems, 7WC-ITS.* Torino, Italy.

Crainic, T. G., Malucelli, F., Nonato, M., Guertin, F., 2005. Meta-Heuristics for a Class of Demand-Responsive Transit Systems. *INFORMS Journal on Computing 17*(1), 10-24.

Daganzo, C. F., 1984. Checkpoint dial-a-ride systems. *Transportation Research B 18*, 315-327.

Desaulniers, G., Desrosiers, J., Solomon, M., 2005. *Column Generation.* New York: Springer Science + Business Media Inc.

Desaulniers, G., Desrosiers, J., Erdmann, A., Solomon, M., Soumis, F., 2002. VRP with Pickup and Delivery. In P. Toth, D. Vigo, *The Vehicle Routing* (pp. 225-242). Philadelphia: SIAM Monographs on Discrete Mathematics and Applications.

Diana, M., Quadrifoglio, L., Pronello, C., 2007. Emissions of demand responsive services as an alternative to conventional transit systems. *Transportation Research Part D: Transport and Environment 12*(9), 183-188.

Dikas, G., Minis, I., 2013. *Lab Report, 2013-05: Generation of test data for Scheduled Paratransit Transport Systems.* Retrieved July 2013, from Design, Operations & Production Systems Lab, Department of Financial and Management Engineering, University of the Aegean: http://labs.fme.aegean.gr/deopsys/?q=SPTS_data_gen.

Feillet, D., Dejax, P., Gendreau, M., Gueguen, C., 2004. An Exact Algorithm for the Elementary Shortest Path Problem with Resource Constraints: Application to Some Vehicle Routing Problems. *Networks 44*, 216-229.

Ferreira, L., Charles, P., Tether, C., 2007. Evaluating Flexible Transport Solutions. *Transportation Planning and Technology 30*(2-3), 249-269.

FLIPPER., 2011. *C4 - Feasibility studies and pilot applications, Deliverable D6 - Feasibility studies report.* INTERREG IVC.

Golden, B. L., Levy, L., Vohra, R., 1987. The orienteering problem. *Naval Research Logistics 34*(3), 307–318.

Irnich, S., Desaulniers, G., 2005. Shortest path problems with resource constraints. In G. Desaulniers, J. Desrosiers, M. Solomon, *Column Generation* (pp. 33-65). Springer.

Laporte, G., Martello, S., 1990. The selective traveling salesman problem. *Discrete Applied Mathematics 26*(2-3), 193-207.

Mageean, J., Nelson, J. D., 2003. The evaluation of demand responsive transport services in Europe. *Journal of Transport Geography 11*(4), 255-270.

Malucelli, F., Nonato, M., Crainic, T. G., Guertin, F., 2001. Adaptive memory programming for a class of demand-responsive transit systems. In S. Voss, J. R. Daduna (Eds.), *Computer-Aided Scheduling of Public Transport* (Vol. 505, pp. 253-273). Berlin, Germany: Springer.

Mulley, C., Nelson, J. D., 2009. Flexible transport services: A new market opportunity for public transport. *Research in Transportation Economics 25*(1), 39-45.

Nelson, J. D., Wright, S. M., Ambrosino, G., Naniopoulos, A., 2010. Recent developments in Flexible Transport Services. *Research in Transportation Economics 29*(1), 243-248.

Nguyen-Hoang, P., Yeung, R., 2010. What is paratransit worth? *Transportation Research Part A: Policy and Practice 44*(10), 841-853.

Quadrifoglio, L., Dessouky, M. M., 2008. Sensitivity Analyses over the Service Area for Mobility Allowance Shuttle Transit (MAST) Services. *Computer-aided Systems in Public Transport Lecture Notes in Economics and Mathematical Systems 600*, 419-432.

Quadrifoglio, L., Dessouky, M. M., Ordonez, F., 2008. Mobility allowance shuttle transit (MAST) services: MIP formulation and strengthening with logic constraints. *European Journal of Operational Research 185*, 481-494.

Quadrifoglio, L., Dessouky, M. M., Palmer, K., 2007. An insertion heuristic for scheduling mobility allowance shuttle transit (MAST) services. *Journal of Scheduling 10*, 25-40.

Quadrifoglio, L., Hall, R. W., Dessouky, M. M., 2006. Performance and design of mobility allowance shuttle transit (MAST) services: Bounds on the maximum longitudinal velocity. *Transportation Science 40*(3), 351-363.

Ropke, S., Cordeau, F., 2009. Branch-and-Cut-and-Price for the Pickup and Delivery Problem with Time Windows. *Transportation Science 43*(3), 267-286.

Savelsbergh, M. W., Sol, M., 1995. The General Pickup and Delivery Problem. *Transportation Science 29*, 17-29.

Vansteenwegen, P., Souffriau, W., Oudheusden, V., 2011. The orienteering problem: A survey. *European Journal of Operational Research* 1-10.

Zografos, K. G., Androutsopoulos, K. N., Sihvola, T., 2008. A methodological approach for developing and assessing business models for flexible transport systems. *Transportation 35*(6), 777-795.