



Discrete Optimization

New mixed integer-programming model for the pickup-and-delivery problem with transshipment



A. Rais*, F. Alvelos, M.S. Carvalho

Centro Algoritmi, Departamento de Produção e Sistemas, Universidade do Minho, 4710-057 Braga, Portugal

ARTICLE INFO

Article history:

Received 11 December 2012

Accepted 15 October 2013

Available online 26 October 2013

Keywords:

Integer programming

Pickup-and-delivery

Vehicle routing

Transshipment

Transportation-on-demand

Modeling

ABSTRACT

In recent years, many important real-world applications are studied as “rich” vehicle routing problems that are variants and generalizations of the well-known vehicle routing problem. In this paper we address the pickup-and-delivery version of this problem and consider further generalization by allowing transshipment in the network. Moreover, we allow heterogeneous vehicles and flexible fleet size. We describe mixed integer-programming formulations for the problem with and without time windows for services. The number of constraints and variables in the models are bounded by polynomial size of the problem. We discuss several problem variants that are either captured by our models or can be easily captured through simple modifications. Computational work gave promising results and confirms that **transshipment in network can indeed enhance optimization.**

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

For a given set of vehicles and customers, the classical *Vehicle Routing Problem (VRP)* seeks to find a set of minimum-cost routes of the vehicles for visiting all customers from a central vehicle depot. The *Pickup-and-Delivery Problem (PDP)* is a generalization of the VRP in which the vehicles are required to service customer requests for picking up and delivering transport loads; the pickup and delivery customers are usually paired. An optimal solution is a set of minimum-cost vehicle routes for servicing all customer requests where the variable costs are associated typically with the distances between the customer locations and the fixed costs are incurred for use of the vehicles.

Many real-world logistics and transportation issues entail time windows for services; therefore, addressing also the time windows is a natural extension of the VRP. Inclusion of time windows thus defines an important class of VRP problems commonly known as the *Vehicle Routing Problem with Time Windows (VRPTW)*; the *Pickup-and-Delivery Problem with Time Windows (PDPTW)* is its analogous counterpart in which both pickup and delivery services have time windows imposed.

The PDP and the PDPTW together address an enormous suite of real-world applications (Toth & Vigo, 2002). Common examples abound in transportation and logistics management such as routing and scheduling of public transports,

transportation-on-demand, emergency transportation of goods and personnel, as well as logistics in humanitarian and disaster management (Christiansen, Fagerholt, Nygreen, & Ronen, 2007; Golden, Assad, & Wasil, 2002; Macharis & Bontekoning, 2004; Powell, Bouzaiene-Ayari, & Simao, 2007).

In this paper we consider a generalization of the PDP by focusing closely on the real-world needs and practices for efficient transport systems. As transportation of people and goods is vital for the economy and the quality of life, efficient transport systems are essential for promoting sustainable growth of economic activities and global competitiveness. Indeed, a recent white paper on transport by the European Commission emphasized urgent needs to adopt innovative transport solutions and optimize performance of transport systems through efficient management of multimodal logistic chains and linking long-haul with last-mile transportation coverage (Roadmap to a Single European Transport Area, 2011).

In order to address these issues and facilitate transfers as well as modifications of the transport loads carried by the vehicles, we consider the notion of transshipment in PDP to get the *Pickup-and-Delivery Problem with Transshipment (PDPT)* that has designated transshipment locations in the underlying network. The PDPT is thus a generalization of the PDP for allowing potentially more routing options as well as improving capacity utilization of the vehicles; therefore, it is also a NP-hard combinatorial optimization problem (Savelsbergh & Sol, 1995).

For a general description of the PDPT, we are given a set of vehicles, locations of the vehicle origin and final depots, a set of locations of customer requests for transporting loads from the pickup to the delivery locations, and the network connecting the locations.

* Corresponding author. Tel.: +351 934670456.

E-mail addresses: abdur.rais@algoritmi.uminho.pt (A. Rais), falvelos@algoritmi.uminho.pt (F. Alvelos), sameiro@algoritmi.uminho.pt (M.S. Carvalho).

Each vehicle is initially located at an origin vehicle depot and its route through the network must end at some final vehicle depot. Customers along with the associated requests for either pickup or delivery of the transport loads are represented by their locations in the network. Unlike in the PDP, the network in the PDPT can have designated transshipment locations where vehicles may stop for transfers and adjustments of their transport loads, the drivers may switch their vehicles or get release times for conforming to policy-related matters, fresh or rested drivers may replace the tired drivers, etc. Typically, similar to the PDP, the optimization aspect of the PDPT also seeks to service all customer pickup-and-delivery requests at minimum total cost where the variable costs are associated typically with the distances between the customer locations and the fixed costs are incurred for use of the vehicles.

This general definition of the PDPT allows multiple origin and final depots for the vehicles, a fleet of heterogenous vehicles having different load-carrying capacities, and unlimited transshipments or transfers of the requested transport loads at the designated transshipment locations. Moreover, distinct customer requests may have the same pickup or delivery locations – i.e., a pickup or delivery location can be involved with more than one customer request. Furthermore, customer requests may or may not be split for carrying by different vehicles from the pickup to the delivery locations. As such, many interesting problem variants can arise and the objective function for optimization can be defined in many different ways – we discuss them in Section 5.

Allowing flexibility of transshipment in the network poses additional challenges for modeling the generalized VRP as spatial and time operations involving the vehicles must be properly matched and synchronized. The model must ensure proper representation of the network for facilitating assignments of the customer requests to potentially various modes of transportation. The model must also incorporate many inherent restrictions typical in multimodal transportation, such as: certain regions may support only rail transportation while other regions may allow land as well as sea options; certain transport loads may only be carried by specific transporters (e.g., crude petroleum in large ocean-going tankers); transportation of hazardous materials through any urban areas may be required to use specialized vehicles as well as specific roads; perishable and pharmaceutical products may only be transported in time-critical vehicles having refrigeration capabilities; or, certain transport loads may be “split” and shared by more than one vehicle to complete the task of transportation.

For addressing such requirements in the real-world applications, much closer attention to detail is necessary for making use of the entire transport system consisting of the land, sea and air modes along with all provisions for any differentiated short-haul and long-haul transport options. For short-haul transportation, transshipment typically allows more flexibility without incurring much additional costs while potentially decreasing the quality of service for inconvenience to passengers or risks of deterioration of goods. However, this may also entail synchronization aspects which are often difficult to satisfy in practice. For long-haul transportation, transshipment typically relates to multimodal transportation which generally involves additional cost, time and storage.

Having flexibility in such systems has become increasingly important in the current globalized as well as competitive economy and this essentially provides impetus for multimodal planning and exploiting all potential options for optimization. However, fruitful research in this domain has been deemed to be in its early stages (Macharis & Bontekoning, 2004) and, to the best of our knowledge, comprehensive work addressing the relevant needs and issues within the framework of optimization is quite recent (Drexler, 2012a, 2012b, 2013). The author has focused on the PDPT and other generalizations of the PDP for numerous real-world scenarios – we will discuss them with literature review in Section 2.

In this paper, we address the PDPT with and without the time windows for services and present *Mixed Integer Programming (MIP)* models for solving the problems. We allow vehicle fleet to be heterogenous and its size to be flexible. The number of constraints and variables in the models are shown to be bounded by polynomial size of the problem. We discuss several problem variants that are either captured by our models or can be easily captured through simple modifications. Computational work gave promising results and confirms that transshipment in network can indeed enhance optimization.

The rest of the paper is organized as follows. We discuss literature relevant to the PDPT in Section 2. In Section 3, we provide a simple example to illustrate the impact of transshipment. We present MIP models for the PDPT in Section 4, followed by a discussion on related problem variants in Section 5. We provide computational analysis in Section 6 and concluding remarks in Section 7.

2. Literature review

Survey of the VRP that includes the PDP can be found in Assad (1988) and Desrosiers and Dumas (1988) and a general overview of the PDP along with the solution methods then available is given in Savelsbergh and Sol (1995). Specific surveys addressing these two broad problem classes are found in Desaulniers, Desrosiers, Erdmann, Solomon, and Soumis (2002) and Mitrović-Minić (1998) while a fairly comprehensive and recent survey of many variants of the VRP including the PDPT is found in Parragh, Doerner, and Hartl (2008). Classification of related problems including the PDPT is given in Berbeglia, Cordeau, Gribkovskaia, and Laporte (2007). Recent advances in the exact solution methods for solving the problems are reviewed in Cordeau, Desaulniers, Desrosiers, Solomon, and Soumis (2002) and Cordeau and Laporte (2007) and dynamic PDPT is addressed in Bouros, Sacharidis, Dalamagas, and Sellis (2011).

To the best of our knowledge, only a handful of papers have so far addressed the PDPT. In three seminal papers (Drexler, 2012a, 2012b, 2013), the author underlined key hurdles for tackling various synchronization aspects including the PDPT and several related problem variants. He introduced different classes of the problems for various vehicle types such as lorries, tractors, trailers and semi-trailers. He has highlighted practical issues arising from a real-world application as a backdrop: Milk collection business in rural Germany. He also highlighted the key hurdles confronting the efforts for developing proper solution methodologies and stressed the needs for focusing on synchronization of the vehicles regarding spatial, temporal, and load-carrying aspects.

A complete MIP model for the PDPT is given in Cortés, Matamala, and Contardo (2010). The authors considered passenger transportation and used decision variables for the arcs as well as the nodes in the underlying network. Their model provides options to passengers for transfer from one vehicle to another at the designated transfer nodes. For each transfer node, they split the node into two and connect the two parts by an arc to explicitly capture the arrival and the departure of a vehicle – thus handling precedence relationships for the transfers of the passengers. Each vehicle is allowed to use a transfer node at most once but via duplication of the transfer nodes in the network, a vehicle may use a transfer node more than once. In total, their model has 23 different sets of constraints including standard arc variables for the vehicle flows and binary node variables to explicitly track the passengers along their entire routes through the network – they do not use flow variables for the flow of the passengers. They presented a branch-and-cut algorithm based on Benders decomposition to solve the model and gave computational results. However, the size of the problem instances addressed is small:

up to 6 transport requests, 2 vehicles and 1 transshipment node. Using their developed algorithm, they reported savings of around 90% CPU time when compared to standard MIP solvers.

In Kerivin, Lacroix, Mahjoub, and Quilliot (2008), the authors considered a variant of the PDPT in which the demand loads may be split for carrying from the pickup to the delivery locations. The vehicles all have same depot and equal transportation capacity. The complexity of this problem is shown to be NP-hard. They described two different MIP formulations based on specifically-defined auxiliary space–time graphs and gave a few valid inequalities with associated separation routines. They also developed a branch-and-cut algorithm for solving the problem and conducted computational work on random problem instances having 6–10 locations and 5–15 demands. For CPU time limit of 5 hours, their algorithm solved most of the smaller problem instances and a few having 15 demands.

The authors in Grunert and Sebastian (2000) considered the problem of long-haul transportation in postal and package delivery in Germany and posed it on an acyclic network with transshipment options. The customer requests are allowed to be split so that different portions can be carried by different vehicles. Their model is based on the multicommodity network flow and used separate flow variables with discretized times to explicitly capture the details of the requests and the vehicles on each arc of the network. As such, they introduced one new node for each pair of the original node and the time period and defined two types of arcs. They used integer variables for the vehicle flows and continuous variables for the request flows. Several exact and heuristic solution approaches for solving the model were proposed but they neither implemented nor performed any computational works.

In Mues and Pickl (2005), the PDPT is considered for path-based MIP model and solving by column generation solution approaches. The authors described one model for the network having exactly one transshipment location and discussed extension to another model for handling several transshipment locations. They mentioned that the computational results on the first model are promising; however, they neither provided all details nor discussed how to handle the pricing problems in the column generation technique for solving the second model.

In Dondo, Méndez, and Cerdá (2009), the authors addressed a PDPT for supply chain optimization and gave a MIP model based on seven different sets of constraints. Their problem concerns distribution of commodities from production plants to dispersed customers using distribution centers for transshipments. They illustrated an example and discussed its solution.

In Takoudjou, Deschamps, and Dupas (2012) the authors have described a MIP model for the PDPT in which transport loads can be split and each vehicle can do at most one transfer of load. The authors used 25 different constraints in their model. For computational work, they used heuristics to solve the test problems generated.

The authors in Masson, Ropke, Lehuédé, and Péton (2013) considered a variant of the PDPT in which direct “shuttle routes” are considered between the pickup and the delivery points. They described three models and used branch-and-cut-and-price algorithm for extensive computational analysis.

Recent heuristic approaches addressing the PDPT can be found in Masson, Lehuédé, and Péton (2013a, 2013b) and Qu and Bard (2012). Insertion heuristics and neighborhood searches have been typically used for developing the solution methodologies. The authors have addressed real-world problem instances arising from aircraft route planning and transportation of disabled people.

In Oertel (2000), the PDPT is addressed for several theoretical and structural results. The author used tabu search algorithm for solving the problem and considered at most one transshipment per request and two potential transshipment locations. Computational results

on randomly-generated and real-world problem instances having up to 70 requests are given.

The authors in Shang and Cuff (1996) developed heuristics to solve the PDPT for minimizing the total cost, tardiness and the travel time. Motivated by a practical health-care problem need for an efficient pickup-and-delivery system to access the patient records before their actual arrival at the clinics, they used concurrent scheduling and insertion heuristic for computing solutions to a multi-objective optimization model for handling the real-world application.

Another work that relates to the PDP for flexibility in transport networks but does not exactly use transshipment is Lin (2011). The author used two types of vehicles, one for long-haul transportation and several others for short-haul transportation. The vehicle for long-haul transportation is used between two key nodes or hubs in the network while the short-haul vehicles are used for collection and distribution of the transport loads at each hub. Somewhat similar to the PDPT, this variant of the PDP only has the hubs providing “restricted” transshipment between the long-haul and the short-haul vehicles. The author gave a MIP model and used hybrid heuristic algorithm consisting of LP relaxation and constructive heuristic for the long-haul transportation problem. A transportation model is then used to assign the routes to the light vehicles and solve for the pickups and the deliveries to the customers. Extensive computational results show impressive gains from this combination of the solution methods used for the addressed problem.

Certain applications in robotics share some similarity with PDPT and various heuristics and computer algorithmic aspects of problem solving can be found in <http://brian.coltin.org/>.

The concept of transshipment is not new and its impact for optimization in the transportation networks can be determined. Its usefulness in the PDP has been established in Mitrović-Minić and Laporte (2006) via empirical analysis on data from a San Francisco-based courier service company that utilized transshipment to serve several cities using homogeneous, uncapacitated vehicles. The authors considered up to 100 transport requests and 4 transshipment locations and confirmed significant savings compared to the non-transshipment option. The savings also increased with the increasing number of requests and the lengths of the time windows.

Several variants and generalizations of the PDP having practical real-world significance have been studied in recent years by several researchers. Interactive decision support systems are typically used in practice to solve the large VRPs and their variants; fully-automated logistics planning system to deal with the enormous complexity inherent in these problems seem unlikely in the near future (Irnich, 2008). Developing such system would certainly require integration of multiple resources (e.g., various types of vehicles, vehicle drivers, transportation load containers, etc.) with simultaneous routing, conforming to policy guidelines, coordination as well as synchronization of spatial, temporal and transport load aspects. Most research focus has been rather academic and the existing solution techniques are not easily adaptable for commercial purposes (Irnich, 2008).

3. An illustrative example

Let us consider a simple example to illustrate a PDPT instance having 8 customers, 2 vehicles and 3 requests. In Fig. 1, we posed the problem on a directed graph having 10 nodes: 8 nodes for the customers and 2 nodes for the origin and the final vehicle depots. Let the two vehicles have the same origin and the final depots shown in the figure by the squares labeled A and G, respectively; assume the capacities of the vehicles 1 and 2 to be 6 and 3 units,

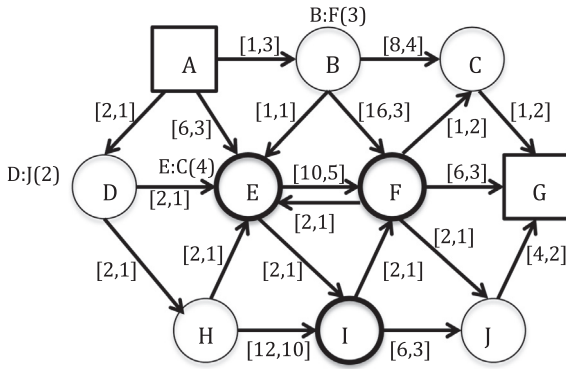


Fig. 1. An example with 8 customers and 3 requests.

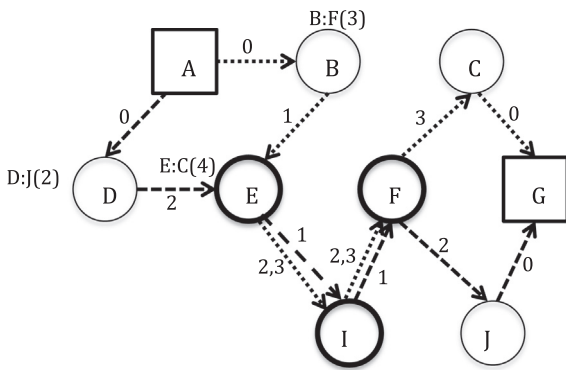


Fig. 2. Optimal solution with transshipment at nodes E and F.

respectively. Associated with each arc ij of the network is a pair $[c_{ij}^1, c_{ij}^2]$ to denote the different transportation costs of the two vehicles; for instance, as shown in the figure, $c_{EF}^1 = 10$ and $c_{EF}^2 = 5$.

The three requests are associated with their pickup nodes and shown in the figure as $p: d(q)$ where q is the quantity of transport load for pickup at node p and delivery at node d . For instance, $D: J(2)$ means that 2 units of transport load should be picked up from node D and delivered to node J . In our example, request 1 is $B: F(3)$ at node B , request 2 is $D: J(2)$ at node D , and request 3 is $E: C(4)$ at node E . Let $\{E, F, I\}$ be the set of transshipment nodes in the network.

As defined, the PDPT on the given network is for finding a set of minimum-cost routes of the two vehicles from node A to node G that satisfies three customer requests at nodes B , D and E . In order to minimize the total cost of the vehicle routes, the two vehicles carrying the requests or the transport loads may transfer, adjust or swap their loads at the transshipment nodes for having different load-carrying capacities as well as transportation costs on the arcs.

In Fig. 2, we show an optimal solution given by the dashed and dotted lines – dotted line for the path of vehicle 1 and dashed line for the path of vehicle 2. We have shown with each arc in the solution the request carried by the vehicle; for instance, vehicle 1 carries only request 1 from node B to node E but jointly the requests 2 and 3 from node E to node I . We used “0” when the vehicle is not carrying any requests.

The vehicles 1 and 2 first traverse empty on arcs AB and AD , respectively, from their common origin depot A . Vehicle 1 then picks up request 1 from node B while vehicle 2 picks up request 2 from node D . Then they both carry their respective transport loads to node E , which is a transshipment node as well as the pickup node for request 3. At node E , vehicle 1 swaps its load with vehicle 2 and picks up request 3; swapping is necessary first to ensure that the two transport loads in vehicle 1 now do not exceed its

capacity. Thereafter, vehicle 1 traverses to node F via node I carrying the loads for requests 2 and 3, while vehicle 2 also traverses the same arcs but carrying only the load for request 1.

At node F , vehicle 2 delivers request 1 and takes request 2 from vehicle 1. Then vehicle 1 traverses to the final depot at node G via node C , where it delivers request 3. Meanwhile, vehicle 2 carries request 2 from node F to J , makes the delivery, and proceeds to the final depot at node G . Two different load transfers are made in this example to obtain an optimal solution. At node E , a swap of the transport loads corresponding to the requests 1 and 2 is done between the two vehicles 1 and 2; at node F , the transport load corresponding to request 2 is transferred from vehicle 1 to vehicle 2.

4. New MIP models

For NP-hard combinatorial optimization problems, “natural” MIP models usually require exponential number of variables or constraints and solving their general instances can be very difficult. For many such problems, as it is often difficult to model and solve for optimal solutions using exact methodologies, heuristic or approximate solutions are typically sought as a compromise. MIP models having polynomial number of constraints and variables can be helpful sometimes regarding this difficulty: although the number of variables and constraints can be many, problem instances of reasonable size can be solved to optimality using effective solvers based on branch-and-bound methodology. Therefore, such MIP modeling can be a viable alternative for attacking many NP-hard problems.

In the subsequent sections, we address the PDPT in its general form: for a given fleet of capacitated vehicles and customer locations with pickup-and-delivery requests, find a set of vehicle routes to satisfy the customer requests at minimum total cost – where costs are associated with the distances between the customer locations as well as the distances between the vehicle depots and the customer locations.

Although most real-world PDPT applications are likely to have imposed time windows for the services, we present a MIP model without the time windows first and then extend it to include the time windows in Section 4.2.

4.1. Pickup-and-delivery problem without time windows

For developing a MIP formulation of the PDPT, we define a few notations from graph theory. Let $G(N, A)$ be a directed graph having the node-set N and the arc-set A . For $i, j \in N$, we denote the arc from i to j as $ij \in A$. We use the term “network” to mean a graph having additional data on its nodes and arcs.

Let K be the set of vehicles indexed by $k = 1, \dots, |K|$; for each vehicle k , we denote its load-carrying capacity by u_k . Let $o(k)$, $d(k) \in N$, respectively, be the initial depot and the final depot of the vehicle $k \in K$.

Let R be the set of customer pickup-and-delivery requests indexed by $r = 1, \dots, |R|$; we denote by q_r the quantity of the request $r \in R$. Associated with each customer request $r \in R$ is a pair $(p(r), d(r))$ with $p(r) \in N$ the pickup node and $d(r) \in N$ the corresponding delivery node; for each request $r \in R$, a transport load of quantity q_r needs to be picked up from $p(r)$ and delivered to $d(r)$.

Denote by $T \subseteq N$ the set of transshipment nodes in G . Let c_{ij}^k be the unit cost of transportation from node $i \in N$ to $j \in N$ using the vehicle $k \in K$.

For the decision variables in the MIP model, let $x_{ij}^k = 1$ if the vehicle k uses the arc ij and $x_{ij}^k = 0$ otherwise $\forall ij \in A$ and $\forall k \in K$. Let $y_{ij}^{kr} = 1$ if the vehicle k carries the request r on the arc ij and $y_{ij}^{kr} = 0$ otherwise $\forall ij \in A$, $\forall k \in K$ and $\forall r \in R$.

The model may now be stated as follows:

$$\begin{aligned}
 & \text{Minimize } \sum_{k \in K} \sum_{ij \in A} c_{ij}^k x_{ij}^k \\
 & \text{subject to: } \sum_{j:ij \in A} x_{ij}^k \leq 1 \quad \forall k \in K, \quad \forall i = o(k) \quad (1) \\
 & \sum_{j:ij \in A} x_{ij}^k = \sum_{j:jl \in A} x_{jl}^k \quad \forall k \in K, \quad \forall i = o(k), \quad \forall l = o'(k) \quad (2) \\
 & \sum_{j:ij \in A} x_{ij}^k - \sum_{j:ji \in A} x_{ji}^k = 0 \quad \forall k \in K, \quad \forall i \in N \setminus \{o(k), o'(k)\} \quad (3) \\
 & \sum_{k \in K} \sum_{j:ij \in A} y_{ij}^{kr} = 1 \quad \forall r \in R, \quad \forall i = p(r) \quad (4) \\
 & \sum_{k \in K} \sum_{j:ji \in A} y_{ji}^{kr} = 1 \quad \forall r \in R, \quad \forall i = d(r) \quad (5) \\
 & \sum_{k \in K} \sum_{j:ij \in A} y_{ij}^{kr} - \sum_{k \in K} \sum_{j:ji \in A} y_{ji}^{kr} = 0 \quad \forall r \in R, \quad \forall i \in T \quad (6) \\
 & \sum_{j:ij \in A} y_{ij}^{kr} - \sum_{j:ji \in A} y_{ji}^{kr} = 0 \quad \forall k \in K, \quad \forall r \in R, \quad \forall i \in N \setminus T \quad (7) \\
 & y_{ij}^{kr} \leq x_{ij}^k \quad \forall ij \in A, \quad \forall k \in K, \quad \forall r \in R \quad (8) \\
 & \sum_{r \in R} q_r y_{ij}^{kr} \leq u_k x_{ij}^k \quad \forall ij \in A, \quad \forall k \in K \quad (9) \\
 & x_{ij}^k \in \{0, 1\} \quad \forall ij \in A, \quad \forall k \in K \quad (10) \\
 & y_{ij}^{kr} \in \{0, 1\} \quad \forall ij \in A, \quad \forall k \in K, \quad \forall r \in R. \quad (11)
 \end{aligned}$$

(6) p(r) & d(r)
consider T also
drop-off/pickup
station

(7) Also need to
exclude o(k) & o'(k)

Same order
cannot be split

The objective of the problem is to find a set of minimum-cost vehicle routes for meeting all customer requests. Constraints (1) enforce that each vehicle may initiate at most one route from its origin depot; constraints (2) enforce that the same vehicle must end the route at its final depot. By using “ \leq ” instead of “ $=$ ” in constraints (1), we capture the fact that not all of the available vehicles may have to be used for meeting the customer requests. Alternatively, we may use “ $=$ ” only if every vehicle has its origin depot connected to its final depot by an arc of zero cost in the underlying network.

Constraints (3) maintain flow conservation of the vehicles through the nodes in the network. Constraints (4) and (5), respectively, enforce all pickups and deliveries of the customer requests. Constraints (6) maintain the request flow conservation at the transshipment nodes allowing requests to switch from one vehicle to another while constraints (7) maintain the request flow conservation at the non-transshipment nodes requiring that any vehicle bringing a request must also leave carrying the same request.

Constraints (8) enforce a vehicle flow on an arc if there is some request flow in the same vehicle on the same arc. Capacity of each vehicle on each arc of the network is assured by constraints (9). Constraints (10) and (11), respectively, require the variables x_{ij}^k and y_{ij}^{kr} to be binary.

The model described above correctly solves the PDPT when the underlying network is acyclic. For general networks, however, we must tackle the possibility of subtours in the solution. Several polynomial-size versions of subtour elimination constraints have been discussed in the literature and we have chosen to use the following constraints that are known to provide tight bounds (Oncan, Altinel, & Laporte, 2009). To describe the constraints, let $z_{ij}^k = 1$ if node i precedes (not necessarily immediately) node j in the route of the vehicle k and $z_{ij}^k = 0$ otherwise. Then a polynomial-size set of subtour elimination constraints can be stated as follows:

$$x_{ij}^k \leq z_{ij}^k \quad \forall i, j \in N, \quad \forall k \in K, \quad o(k) \neq i, \quad j \neq o'(k) \quad (12)$$

$$z_{ij}^k + z_{ji}^k = 1 \quad \forall i, j \in N, \quad \forall k \in K, \quad o(k) \neq i, \quad j \neq o'(k) \quad (13)$$

$$z_{ij}^k + z_{jl}^k + z_{li}^k \leq 2 \quad \forall i, j, l \in N, \quad \forall k \in K, \quad o(k) \neq i, j, \quad l \neq o'(k). \quad (14)$$

These constraints essentially enforce precedence relationships between the nodes through linear ordering: for any two nodes $i, j \in N$, either i precedes j or j precedes i but not both. Moreover, for

any $ij \in A$ such that $x_{ij}^k = 1$ for some $k \in K$, the constraints will also force i to precede j since i is an immediate predecessor of j .

Our MIP model is based on multi-commodity network flow consisting of two distinct flows: one flow type for the vehicles from their origin depots to the final depots and another flow type for the requests or the transport loads from their pickup nodes to the corresponding delivery nodes. Since transporting requests requires vehicles, we associate the vehicles with the requests also to capture the request flows. Indeed, this model relates to two well-known problems: the capacitated multi-commodity network flow problem and the fixed-charge network design problem. Linking the two distinct flows requires certain constraints that are similar to the network design constraints.

The model has $|K|$ commodities for the vehicles and $|R||K|$ commodities for the requests in the vehicles. Constraints (1)–(3) together form a multi-commodity network flow problem for the vehicles while constraints (4)–(7) form another multi-commodity network flow problem for the requests in the vehicles; by constraints (6), transfer of requests between the vehicles can only take place at the transshipment nodes. Hence, a feasible solution to the model must match up these two distinct flows, namely the flow of the vehicles and the flow of the requests in the vehicles.

For any feasible solution, we must have: (i) the vehicle routes starting at the origin depots and ending at the final depots; and, (ii) all request flows “covered” by the vehicle flows. For satisfying (ii), note that each path connecting the corresponding pair of pickup and delivery nodes of any particular request must be a subpath of some feasible vehicle route. The “linking” constraints (8) ensure proper match-ups of the two flows as well as feasible vehicle routes for covering the pickup-delivery paths corresponding to the requests.

Polynomial number of variables and constraints. By adding up the total number of variables and constraints, it can be seen that the model has polynomial number of variables and constraints based on the size of the problem instance. First, consider the number of variables in the model. There are a total of $|A||K|$ x -variables and $|A||K||R|$ y -variables. For the worst-case scenario, $|R|$ is bounded by $|N|^2$ while $|K|$ can be assumed to be $O(|R|)$. Hence, the total number of variables is bounded by $|A||N|^4$ or $O(n^6)$ where $n = |N|$.

For the total number of constraints in the model, note that the constraint sets (1) and (3)–(6) are bounded by $O(n^3)$. Constraints (2), (8) and (9) are bounded by $O(n^4)$. Since T is bounded by $|N|$, the constraint set (7) is bounded by $O(n^5)$. Finally, the variables and constraints used for subtour elimination are also known to have polynomial size.

4.2. Pickup-and-delivery problem with time windows

The classical VRP is typically a spatial problem but temporal aspects in routing are important for applications such as postal deliveries, banking, garbage collection, school-bus routing, sales and services. Certain situations may also require timely access, verification and payment upon delivery of products and services. Therefore, much more research focus is given to VRPTW for capturing the real-world applications more accurately.

For the service time windows in the VRPTW, two key considerations are noteworthy:

- Hard time windows – they must be strictly followed; and,
- Soft time windows – they can be violated by incurring some penalty costs.

For PDPT, time windows may be associated for picking up and delivering transport loads. They can be frequent and nested, with several distinct ones at the transshipment nodes – especially in long-haul transportation. While these additional time-window

requirements make the PDPT much more realistic, they also make it much more difficult to solve.

To develop a MIP model of the PDPT with time windows, we define a few additional notations. For an arc $ij \in A$, let τ_{ij}^k be the time required by the vehicle $k \in K$ to go from node i to node j . Denote by t_j^k and \bar{t}_j^k , respectively, the times of arrival and departure of the vehicle k at the node j . Then for each arc $ij \in A$ such that $x_{ij}^k = 1$, we must have $t_j^k \geq \bar{t}_i^k + \tau_{ij}^k$ as well as $\bar{t}_j^k \geq t_i^k$. They can be written using the big- M method as the following constraints:

$$\bar{t}_i^k + \tau_{ij}^k - t_j^k \leq M(1 - x_{ij}^k) \quad \forall ij \in A, \quad \forall k \in K \quad (15)$$

$$t_j^k \leq \bar{t}_j^k \quad \forall j \in N, \quad \forall k \in K. \quad (16)$$

For a request $r \in R$, let $[a_{p(r)}, b_{p(r)}]$ and $[a_{d(r)}, b_{d(r)}]$, respectively, be the associated time windows for the pickup and the delivery operations. Then we must also have the following constraints to ensure compliance with the time intervals:

$$a_{p(r)} \leq t_{p(r)}^k, \quad \bar{t}_{p(r)}^k \leq b_{p(r)} \quad \forall r \in R, \quad \forall k \in K \quad (17)$$

$$a_{d(r)} \leq t_{d(r)}^k, \quad \bar{t}_{d(r)}^k \leq b_{d(r)} \quad \forall r \in R, \quad \forall k \in K. \quad (18)$$

To handle precedence and synchronization of the transport load transfers between vehicles at the transshipment nodes, we also need logical constraints. Let $s_{jr}^{kl} = 1$ if the request $r \in R$ is transferred from some vehicle $k \in K$ to vehicle $l \neq k$ at some node $j \in N$ and $s_{jr}^{kl} = 0$ otherwise. Then we add the following constraints:

$$\sum_{j:ij \in A} y_{ji}^{kr} + \sum_{j:ij \in A} y_{ij}^{lr} \leq s_{jr}^{kl} + 1 \quad \forall r \in R, \quad \forall i \in T, \quad \forall k, l \in K, \quad k \neq l \quad (19)$$

$$t_j^k - \bar{t}_j^l \leq M(1 - s_{jr}^{kl}) \quad \forall r \in R, \quad \forall j \in T, \quad \forall k, l \in K, \quad k \neq l. \quad (20)$$

Constraints (19) and (20) together allow a request r at node j to transfer from vehicle k to vehicle l only if the vehicle k arrives before the departure of vehicle l . The variable s_{jr}^{kl} at node $j \in N$ can be seen as a logical counter for transshipment.

A MIP model for the PDPT with time windows on general graphs is thus given by the constraints (1)–(14) in Section 4.1 together with the constraints (15)–(20) described above. The model can also be seen to have polynomial number of variables and constraints by similar arguments of Section 4.1.

5. Problem variants captured by MIP models

Several different versions of the PDPT address real-world applications. In general, many of their problem requirements can be addressed rather easily in modeling (e.g., restriction on total distance or hours traveled by a vehicle can be enforced by adding explicit constraints); however, factors such as ordered pickups and deliveries (e.g., commonly known as “linehauling” and “backhauling”) or restriction on transshipments (e.g., maximum number of transshipments by a vehicle, use of specific transshipment locations by certain vehicle types, etc.) typically require significantly complex modeling.

In its current form, our MIP model for the PDPT is fairly general and allows multiple origin and final depots for the vehicles, heterogeneous vehicle types having different load-carrying capacities, unlimited transshipment at transshipment nodes, and flexible vehicle fleet size or the flexibility of utilizing minimum number of vehicles for meeting all customer requests. Moreover, distinct customer requests can have same pickup or delivery location – i.e., a pickup or delivery location can have more than one request for transport load pickup or delivery. However, customer requests are not allowed to be “split” for carrying by different vehicles.

The following is a brief summary of several problem variants that are either already captured in our model or can be easily

captured via simple modifications. Some of these variants are not restricted to PDPT only and may apply to PDP or VRP; they are presented here within the context of having transshipment in the network to address a much richer class of VRP problems. Most models in the literature do not allow straight-forward modification for different problem variants.

Splittable pickup-and-delivery requests. Customer requests for pickup and delivery can be heterogeneous or homogeneous in terms of the type and quantity of transport loads. Moreover, a request may be either “split” with portions carried by different vehicles from the pickup to delivery locations, or remain intact as a whole unit for carrying by only one vehicle. Splitting the requests is practical and useful in many real-world applications of PDP such as transportation of petroleum products, food grains, minerals, construction materials, etc. Moreover, for having transshipment, splittable requests in PDPT can greatly enlarge the solution set for optimization.

As defined in our model, the requests are allowed to be heterogeneous in quantity. If we want some requests to be split also so that two or more vehicles may share carrying them from the pickup to the delivery locations, we must relax the corresponding constraints (11) for the y -variables need not be binary; but by constraints (8), they will remain bounded by 1. Therefore, the following constraints should replace constraints (11):

$$y_{ij}^{kr} \geq 0 \quad \forall ij \in A, \quad \forall k \in K, \quad \forall r \in R. \quad (21)$$

If we want to allow only a subset of the requests in R to be split, we can selectively replace constraints (11) by a subset of the constraints (21). Since the y -values in a feasible solution may now be fractional, we must redefine positive y -value as flow of the request having fractional split amount. Thus, in this case the quantity of any customer request r in the problem, denoted by q_r , can be allowed to be larger than the load-carrying capacities of the vehicles even though the problem is infeasible by constraints (9) for the non-split case when $q_r > u_k, \forall k \in K$.

The MIP models described in Grunert and Sebastian (2000) and Kerivin et al. (2008) allow requests to be split but the models in Cortés et al. (2010) and Mues and Pickl (2005) cannot be used for splittable requests.

One pickup-and-delivery request per vehicle. Our model allows more than one request to be carried in the same vehicle at the same time if vehicle capacity is not violated. If we do not want a vehicle to carry more than one request at a time, the following constraints can be added so that at most one request is carried by a vehicle on any arc:

$$\sum_{r \in R} y_{ij}^{kr} \leq 1 \quad \forall ij \in A, \quad \forall k \in K. \quad (22)$$

In this case a vehicle may still service more than one request while going from its origin to final depot but the different pickup-and-delivery requests will be transported sequentially rather than in parallel. At the transshipment nodes, different vehicles may still transfer requests between them if it helps with optimization.

Number and types of vehicles. A common optimization issue in the PDPT is regarding the number of vehicles necessary for meeting the problem needs and it is often addressed as an independent problem. Constraints (1) in our model, however, uses “ \leq ” instead of “ $=$ ” to address exactly this possibility of using only the vehicles necessary based on their capacities. The objective function can be modified further by using fixed costs $f^k > 0$ for using vehicles $k \in K$, replacing the objective function by the following:

$$\sum_{k \in K} \sum_{ij \in A, i \in o(k)} f^k x_{ij}^k + \sum_{k \in K} \sum_{ij \in A} c_{ij}^k x_{ij}^k. \quad (23)$$

Another aspect concerns the types of vehicles. As used in our model, the vehicles are homogeneous in the sense that each may carry any request. However, heterogeneous vehicle types (e.g., freezer or refrigeration capability, secure storage, etc.) may be desirable for specific requests (e.g., transportation of frozen and perishable goods, high-value products, etc.). Different vehicle types may be used for long-haul vis-à-vis short-haul routes: long-haul vehicles transport requests between key locations while short-haul vehicles distribute or collect requests in the neighborhoods of key locations. The key locations may indeed be the designated hubs or transshipment nodes in the network.

Such notion of different vehicle types will require redefining the y -variables for request flow selectively based on vehicle-request compatibility and adding the constraints $y_{ij}^{kr} = 0, \forall ij \in A, \forall k \in K, \forall r \in R$ if the vehicle k is incapable of carrying the request r .

Vehicle depots and routes. The origin and final destination depots of the vehicles may be one central depot or multiple depots. Moreover, multiple depots may be identical or non-identical with each vehicle assigned to a distinct depot. As defined, our model is totally flexible in this context of different vehicle depots because all vehicles may use one depot for both origin and final destination or several distinct depots for the origin and final destination of each vehicle. The models typically found in the literature consider the classical notion of the same vehicle origin and final depot.

In some situations, vehicle routes may be restricted to remain within certain distance. Moreover, certain vehicle types may be required to use only designated routes. Both of these aspects can be handled by adding new constraints to our MIP model. For the former, if L^k is the maximum distance of the route allowed for the vehicle $k \in K$ and l_{ij} is the distance of each arc $ij \in A$, then the following constraint need to be added:

$$\sum_{ij \in A} l_{ij} x_{ij}^k \leq L^k. \quad (24)$$

For the latter aspect of restricted routes, x -variables can be defined selectively with some initialized to zero for allowing only certain arcs in a feasible solution. This can be particularly useful when modeling for long-haul vis-à-vis short-haul transportation with transshipment nodes or hubs.

Transshipment can play a significant role when vehicle routes and travel distances are restricted: requests can be exchanged between different vehicles at the transshipment nodes to circumvent restricted routes or maximum distance limits.

Vehicle stops. A vehicle may be allowed to end its route at delivery or transshipment nodes upon completion of its load delivery instead of ending its route at the designated final depot. Essentially, this allows the vehicles to wait at the transshipment or delivery nodes until new requests become available.

Such embellishment for the problem scenario can be addressed in our model by: (i) deleting all vehicle flow constraints to the destination vehicle depot; and, (ii) modifying the constraints for vehicle flow and request flow conservation at the transshipment and delivery nodes. For (ii), with D the set of delivery nodes in the network, we can simply replace the flow conservation equalities (3) and (6) in the model by the following inequalities:

$$\sum_{j:ij \in A} x_{ij}^k - \sum_{j:ji \in A} x_{ji}^k \leq 0 \quad \forall k \in K, \quad \forall i \in T \cup D \setminus \{o(k), o'(k)\} \quad (25)$$

$$\sum_{k \in K} \sum_{j:ij \in A} y_{ij}^{kr} - \sum_{k \in K} \sum_{j:ji \in A} y_{ji}^{kr} \leq 0 \quad \forall r \in R, \quad \forall i \in T \cup D. \quad (26)$$

For consolidation of the transport loads as well as increasing capacity utilization of the vehicles, waiting at the transshipment nodes can be especially important for optimization.

Transshipment. The number of times a vehicle may do transshipment may be restricted (e.g., some vehicles may not do

transshipment at all, some may do transshipment at most once, others may not do transshipment at some particular node more than two times, etc). New constraints can be added to the model to account for such possibilities.

Let us first recall the transshipment variable defined in Section 4: for $j \in T, r \in R, k, l \in K, k \neq l$, binary variable $s_{jr}^{kl} = 1$ if there is transfer of request r from vehicle k to l at node j , and $s_{jr}^{kl} = 0$ otherwise. Now to enforce at most one transshipment per vehicle at each transshipment node, we need to add:

$$\sum_{r \in R} \sum_{l \in K, l \neq k} s_{jr}^{kl} \leq 1 \quad \forall j \in T, \quad \forall k \in K. \quad (27)$$

Restriction of one transshipment per vehicle can be typically found in a network having a central hub where all request loads are first brought and sorted before assignment for delivery is made. Also, the hub can serve as the intermediate point for transfer of requests from one set of vehicles to another set of vehicles when the pickup and the delivery vehicles are distinct. Other scenarios for restrictive transshipment may include long-haul and short-haul transportation planning using different types of vehicles.

In certain situations, we may want to encourage transshipment at certain transshipment nodes. This can be accomplished by adding to the objective function the following term for each such transshipment node $j \in T$:

$$-\sum_{r \in R} \sum_{l \in K, l \neq k} s_{jr}^{kl} \quad \forall k \in K. \quad (28)$$

This new term in the objective function has negligible value compared to the previous part. Therefore, the additional term will only establish preference between solutions having same value. Moreover, the variables for the different transshipment nodes in the term can be weighted also for establishing some relative order of preference. Thus, models can be built to address different problem scenarios when modified objective function and constraints for restrictive transshipment are taken together.

Dial-a-ride problem with transshipment. Amongst the many variants of the PDP, the *Dial-a-Ride Problem* (DARP) is very well known and widely studied. The problem is defined by the customers themselves seeking minimum-cost transportation from their initial locations to final destinations. It is an important present-day problem for having aging population living in fairly dispersed rural and urban areas; for controlled budget with value, access to transportation and emergency services can be key enablers for improving livelihood of many old and physically disabled people (Toth & Vigo, 1997). In Cordeau and Laporte (2007), the authors have extensively surveyed models and solution methods for the DARP. *Dial-a-Ride Problem with Transshipment* (DARPT) is the counterpart of DARP for having transshipment as an option and has been introduced in Masson et al. (2013b).

If DARPT only concerns customer requests for travel, then its model can be obtained from that of the PDPT by setting $q_r = 1, \forall r \in R$. Compared to the DARP, however, DARPT has the potential to address a much broader class of associated problems for having transshipment as an option. Many DARPs typically require explicit constraints to capture travel time limitations as well as waiting times of the riders; for DARPT, total number of transshipment or transfer for each rider may be considered. To obtain models of DARPTs in such situations, the models for the PDPT can be appropriately modified by including additional constraints for addressing these issues.

DARPT may typically address different optimization aspects also for which different objective functions are meaningful. For instance, minimizing total waiting times of the customers or “total vehicle tardiness” can be practical as well as important. Spending

long times in transportation may not be desirable for the customers also and minimizing the maximum travel time or maximum travel distance may be appropriate. These different objectives essentially address minimization of some form of “inconvenience” of the customers that can be captured via appropriate modification of the objective function.

6. Computational results

For computational analysis, we used the GUROBI 5.5.0 optimization software running on Intel i5, 1.7 gigahertz, 64-bit processor with 4 gigabyte RAM and Windows 7 OS. We used problem instances from those in Li and Lim (2001) for the PDP, which are related to the well-known Solomon instances. The datasets are available at <http://www.sintef.no/Projectweb/TOP/PDPTW/> and from the corresponding author.

For analyzing transshipment vis-à-vis non-transshipment options, we used small problem instances as time required by standard MIP solvers to obtain optimal solutions increases sharply with increasing size of the problem instances. From the 10 different datasets of Li and Lim (2001), we picked subsets of the first 10 nodes and the first 14 nodes to create, respectively, our small 10-node and 14-node problem instances. The pickup-and-delivery requests are paired and so the number of nodes in the network, without the vehicle depots, is even.

Each dataset contains $X - Y$ coordinates for the locations of the nodes; hence, the lengths of the arcs in the network are euclidean distances satisfying triangle inequality. For the locations of the vehicle depots, we randomly generated origin and final depots for each vehicle, scattered over the $X - Y$ region formed by the nodes of the network. The number of vehicles equals to the number of requests. Therefore, for a 14-node problem instance, seven different vehicles are assigned to seven distinct randomly-generated origin depots and seven distinct randomly-generated final depots.

We associated a “cost factor” with each vehicle for differentiating the costs of using different vehicles. For each coefficient in the

objective function corresponding to each arc in the network, the “length” or the distance associated with the arc is multiplied by the fractional cost factor of the vehicle to determine the corresponding “effective” cost coefficient.

We varied the capacities of the vehicles randomly as well. And, in order to ensure that a problem remains feasible, at least one vehicle is given enough capacity for carrying the largest transport load request.

Moreover, since we created small test problems from rather large problem instances, we adjusted the time windows to only 50% of the originals given in Li and Lim (2001).

For proper comparison of the computational results, the same vehicle depot locations are maintained for two different computational runs: the problems are first solved without transshipment, following which they are solved again with the same dataset but allowing transshipment at all nodes in the network. The results are tabulated in Tables 1–3.

All 10-node and 14-node problem instances were solved to optimality. The advantage of transshipment in PDPT can be seen from the optimal values obtained: compared to the non-transshipment problem instances, optimal values are better in five of the 10 10-node instances and eight of the 10 14-node instances when transshipment is allowed.

Our computational work is based on networks having euclidean datasets satisfying triangle inequality, which are not particularly conducive or effective for transshipment in network. However, even for this special cost structure, we achieved some reduction of optimal value that can translate into enormous cost savings. In general, real-world networks may have much different cost structure and potential gain from transshipment can be significant.

Although not in each case, the CPU times required to solve the problem instances to optimality are often larger for the transshipment models compared to non-transshipment counterparts. This should be expected because the transshipment models in general have many more constraints and variables than their non-transshipment counterparts.

Table 1
Optimal values and CPU times for PDPT without time windows (10 nodes).

| Instance | Optimal value | CPU time | Optimal value (transshipment) | CPU time (transshipment) | Gap in optimal values (%) |
|----------|---------------|----------|-------------------------------|--------------------------|---------------------------|
| 10n1a | 1121.58 | 4.24 | 1083.09 | 9.57 | 3.43 |
| 10n2a | 931.06 | 33.92 | 867.01 | 8.54 | 6.88 |
| 10n3a | 1221.65 | 54.46 | 1221.65 | 62.15 | 0 |
| 10n4a | 1105.11 | 108.79 | 1105.11 | 671.03 | 0 |
| 10n5a | 848.89 | 291.84 | 833.89 | 311.48 | 1.77 |
| 10n6a | 1018.35 | 1.88 | 1018.35 | 8.78 | 0 |
| 10n7a | 1272.59 | 223.85 | 1249.51 | 137.92 | 1.81 |
| 10n8a | 750.52 | 206.56 | 750.52 | 808.83 | 0 |
| 10n9a | 1031.93 | 516.64 | 1028.19 | 1139.01 | 0.36 |
| 10n10a | 1224.71 | 65.12 | 1224.71 | 94.47 | 0 |

Table 2
Optimal values and CPU times for PDPT with time windows (10 nodes).

| Instance | Optimal value | CPU time | Optimal value (transshipment) | CPU time (transshipment) | Gap in optimal values (%) |
|----------|---------------|----------|-------------------------------|--------------------------|---------------------------|
| 10n1a | 1121.58 | 5.11 | 1083.09 | 15.85 | 3.43 |
| 10n2a | 936.15 | 26.4 | 933.88 | 24.52 | 0.24 |
| 10n3a | 1221.65 | 1.59 | 1221.65 | 8.93 | 0 |
| 10n4a | 1124.66 | 102.97 | 1123.18 | 283.71 | 0.13 |
| 10n5a | 872.92 | 76.18 | 865.62 | 112.47 | 0.84 |
| 10n6a | 1018.35 | 0.53 | 1018.35 | 0.79 | 0 |
| 10n7a | 1249.51 | 8.4 | 1249.51 | 113.01 | 0 |
| 10n8a | 750.52 | 35.74 | 750.52 | 82.16 | 0 |
| 10n9a | 1031.93 | 64.65 | 1031.93 | 100.06 | 0 |
| 10n10a | 1269.23 | 4.81 | 1267.58 | 7.51 | 0.13 |

Table 3
Optimal values and CPU times for PDPT with time windows (14 nodes).

| Instance | Optimal value | CPU time | Optimal value (transshipment) | CPU time (transshipment) | Gap in optimal values (%) |
|----------|---------------|----------|-------------------------------|--------------------------|---------------------------|
| 14n1a | 695.78 | 6027.34 | 683.95 | 11470.86 | 1.70 |
| 14n2a | 599.57 | 5.04 | 594.72 | 17.86 | 0.81 |
| 14n3a | 671.5 | 852.47 | 650.52 | 1161.82 | 3.12 |
| 14n4a | 592.61 | 94.21 | 589.35 | 2683.32 | 0.55 |
| 14n5a | 566.13 | 521.39 | 566.13 | 3997.69 | 0 |
| 14n6a | 1021.28 | 2076.35 | 1005.24 | 18182.82 | 1.57 |
| 14n7a | 1048.71 | 272.34 | 1013.79 | 1079.37 | 3.33 |
| 14n8a | 819.11 | 1699.11 | 816.6 | 9368.72 | 0.31 |
| 14n9a | 704.02 | 71.26 | 704.02 | 1354.29 | 0 |
| 14n10a | 536.44 | 27.33 | 535.02 | 85.78 | 0.26 |

7. Conclusion

A key issue for promoting multimodality and interoperability in transportation is the transshipment of people and goods between the vehicles and the modes of transportation. For multimodal transportation and logistics planning, the embellishment of transshipment translates into flexibility for changing the modes and the operators of the transportation system. For additional flexibility, the relevant transportation networks can be designed with the transshipment nodes strategically placed to fulfill the long-haul as well as the short-haul transport needs by different means. This can help on optimizing the objective further by reducing the total cost of the vehicle routes for servicing all customer requests.

It is easy to recognize that transport systems can benefit from transshipment but very few studies have addressed it from the perspective of optimization. In fact, recent surveys and relevant papers point to significant gaps between the needs in the real-world applications and current research work addressing them. Several variants of the PDPT have been addressed but most researchers have focused exclusively on heuristic solution methods – mostly because the problems are NP-hard – and exact methods are seldom considered. Consequently, it remains unclear as to what extent the exact methodologies, such as a MIP model combined with branch-and-cut and branch-and-bound solution approaches, can be useful for tackling these important real-world problems. Our goal has been to make some contribution in this area: we focused on MIP modeling and used general MIP solver for solving the problems.

We have presented new MIP formulations for the PDPT with and without time windows. Our modeling concept essentially exploited the fact that there are two distinct flows through the network when the VRP is combined with the pickup-and-delivery requests: vehicle flow and request flow. Solving the problem therefore requires proper “matching” of these two flows and our models made use of multi-commodity flows with linking constraints to serve that purpose. The MIP models are polynomially bounded in the size of the problem and thus can be entered easily into computers for solving by standard optimization software.

For computational analysis, we used a commercial solver that uses simplex method on linear-programming relaxations combined with branch-and-cut and branch-and-bound techniques to solve the MIP models. For selected non-transshipment and transshipment problem instances, we compared the results consisting of the CPU times and the optimal values. We found that, compared to the non-transshipment models, the transshipment models typically require more CPU-time for solving them but have the potential to yield better optimal values: this was evident in eight of the 10 PDPT instances having 14 nodes each (see Table 3).

Computational results also show that the CPU time required for solving a MIP model increases sharply when the size of the

problem increases. However, current development and advancement work in both software and solution techniques for the MIP models are quite promising for tackling larger problem instances in the future.

In our problem instances for computational analysis, the randomly-generated vehicle depots may not be favorably located with respect to the nodes in the network for showing significant advantage of transshipment over non-transshipment. Indeed, to properly establish the potential advantage of transshipment, location planning ought to be done first for determining the favorable locations of the vehicle depots. In real-world applications, however, locations of the vehicle depots as well as transshipment nodes in the network are often pre-determined and such location planning from “scratch” may not be pragmatic. Hence, our work can be seen as having reasonable practical significance.

Acknowledgment

This work has been partially funded by FCT (Fundação para a Ciência e Tecnologia – Portugal) through the project “SearchCol: Metaheuristic search by column generation” (PTDC/EIA-EIA/100645/2008).

References

- Assad, A. A. (1988). Modeling and implementation issues in vehicle routing. In B. L. Golden & A. A. Assad (Eds.), *Vehicle routing: Methods and studies* (pp. 7–45). Amsterdam: North-Holland.
- Berbeglia, G., Cordeau, J. F., Gribkovskaia, I., & Laporte, G. (2007). Static pickup and delivery problems: A classification scheme and survey. *TOP*, 15, 1–31.
- Bouros, P., Scharidis, D., Dalamagas, T., & Sellis, T. (2011). Dynamic pickup and delivery with transfers. In D. Pfoser, Y. Tao, K. Mouratidis, M. A. Nascimento, M. Mokbel, & S. Shekhar, et al. (Eds.), *Advances in spatial and temporal databases. Lecture notes in computer science* (Vol. 6849, pp. 112–129). Berlin Heidelberg: Springer.
- Christiansen, M., Fagerholt, K., Nygreen, B., & Ronen, D. (2007). Maritime transportation. In C. Barnhart & G. Laporte (Eds.), *Handbooks in operations research and management science: Transportation*, 14 (pp. 189–284). Amsterdam: North-Holland.
- Cordeau, J. F., Desaulniers, G., Desrosiers, J., Solomon, M. M., & Soumis, F. (2002). VRP with time windows. In Toth, P., & Vigo, D. (Eds.), *The vehicle routing problem* (Vol. 9, pp. 175–193). Philadelphia, PA: SIAM Monographs on Discrete Mathematics and Applications.
- Cordeau, J. F., & Laporte, G. (2007). The dial-a-ride problem: Models and algorithms. *Annals in Operations Research*, 152, 29–46.
- Cortés, C., Matamala, M., & Contardo, C. (2010). The pickup and delivery problem with transfers: Formulation and a branch-and-cut solution method. *European Journal of Operational Research*, 200, 711–724.
- Desaulniers, G., Desrosiers, J., Erdmann, A., Solomon, M. M., & Soumis, F. (2002). VRP with pickup and delivery. In Toth, P., & Vigo, D. (Eds.), *The vehicle routing problem* (Vol. 9, pp. 225–242). Philadelphia, PA: SIAM Monographs on Discrete Mathematics and Applications.
- Desrosiers, J., & Dumas, Y. (1988). The shortest path for the construction of vehicle routes with pick-up, delivery and time constraints. In H. Eiselt & G. Pederzoli (Eds.), *Advances in optimization and control, lecture notes in economics and mathematical systems* (pp. 144–157). Heidelberg Berlin: Springer.
- Dondo, R., Méndez, C., & Cerdá, J. (2009). The supply-chain pick-up and delivery problem with transshipments. In J. Jeżowski & J. Thullie (Eds.), *19th European symposium on computer aided process engineering ESCAPE19* (pp. 1009–1014). Elsevier B.V./Ltd..
- Drexel, M. (2012a). Synchronization in vehicle routing – A survey of VRPs with multiple synchronization constraints. *Transportation Science*, 297–316.
- Drexel, M. (2012b). Rich vehicle routing in theory and practice. *Logistics Research*, 5, 47–63.
- Drexel, M. (2013). Applications of the vehicle routing problem with trailers and transshipments. *European Journal of Operational Research*, 227, 275–283.
- Golden, B. L., Assad, A. A., & Wasil, E. A. (2002). Routing vehicles in the real world: Applications in solid waste, beverage, food, dairy, and newspaper industries. In Toth, P., & Vigo, D. (Eds.), *The vehicle routing problem* (Vol. 9, pp. 245–286). Philadelphia, PA: SIAM Monographs on Discrete Mathematics and Applications.
- Grunert, T., & Sebastian, H. J. (2000). Planning models for long-haul operations of postal and express shipment companies. *European Journal of Operational Research*, 122, 289–309.
- Irnich, S. (2008). A unified modeling and solution framework for vehicle routing and local search-based metaheuristics. *INFORMS Journal on Computing*, 20, 270–287.
- Kerivin, H. L. M., Lacroix, M., Mahjoub, A. R., & Quilliot, A. (2008). The splittable pickup and delivery problem with reloads. *European Journal of Industrial Engineering*, 2–2, 112–133.

- 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* (pp. 333–340). Los Alamitos, CA: IEEE Computer Society.
- Lin, C. K. Y. (2011). A vehicle routing problem with pickup and delivery time windows, and coordination of transportable resources. *Computers & Operations Research*, 38, 1596–1609.
- Macharis, C., & Bontekoning, Y. M. (2004). Opportunities for OR in intermodal freight transport research: A review. *European Journal of Operational Research*, 153, 400–416.
- Masson, R., Lehuédé, F., & Péton, O. (2013a). An adaptive large neighborhood search for the pickup and delivery problem with transfers. *Transportation Science*, 47, 344–355.
- Masson, R., Lehuédé, F., & Péton, O. (2013b). The dial-a-ride problem with transfers. *Computers & Operations Research*, 41, 12–23. <http://dx.doi.org/10.1016/j.cor.2013.07.020>.
- Masson, R., Ropke, S., Lehuédé, F., & Péton, O. (2013). A branch-and-cut-and-price approach for the pickup and delivery problem with shuttle routes. *European Journal of Operational Research*. <http://dx.doi.org/10.1016/j.ejor.2013.08.042>.
- Mitrović-Minić, S. (1998). *Pickup and delivery problem with time windows: A survey*. Technical Report SFU CMPT TR 1998-12. Burnaby, BC, Canada: School of Computing Science, Simon Fraser University.
- Mitrović-Minić, S., & Laporte, G. (2006). The pickup and delivery problem with time windows and transshipment. *INFOR*, 44, 217–227.
- Mues, C., & Pickl, S. (2005). Transshipment and time windows in vehicle routing. In *Proceedings of the 8th international symposium on parallel architectures. Algorithms and networks* (pp. 113–119). Piscataway, NJ: IEEE.
- Oertel, P. (2000). *Routing with reloads*. PhD Thesis. Faculty of Mathematics and Natural Sciences, University of Cologne.
- Oncan, T., Altinel, I. K., & Laporte, G. (2009). A comparative analysis of several asymmetric traveling salesman problem formulations. *Computers & Operations Research*, 36, 637–654.
- Parragh, S. N., Doerner, K. F., & Hartl, R. F. (2008). A survey on pickup and delivery problems Part II: Transportation between pickup and delivery locations. *Journal für Betriebswirtschaft*, 58, 81–117.
- Powell, W. B., Bouzaïene-Ayari, B., & Simao, H. P. (2007). Dynamic models for freight transportation. In C. Barnhart & G. Laporte (Eds.), *Handbooks in operations research and management science: transportation* (Vol. 14, pp. 285–365). Amsterdam: North-Holland.
- Qu, Y., & Bard, J. F. (2012). A GRASP with adaptive large neighborhood search for pickup and delivery problems with transshipment. *Computers & Operations Research*, 2439–2456.
- Roadmap to a Single European Transport Area: Towards a competitive and resource efficient transport system. (28.3.2011). Section (2011) 144 final, European Commission, Brussels.
- Savelsbergh, M. W. P., & Sol, M. (1995). The general pickup and delivery problem. *Transportation Science*, 29, 17–29.
- Shang, J. S., & Cuff, C. K. (1996). Multicriteria pickup and delivery problem with transfer opportunity. *Computers and Industrial Engineering*, 631–645.
- Takoudjou, T. R., Deschamps, J. C., & Dupas, R. (2012). A MIP formulation for the pickup and delivery problem with time window and transshipment. In *Proceedings of the 14th IFAC symposium on information control problems in manufacturing* (Vol. 14-1, pp. 333–338).
- Toth, P., & Vigo, D. (Eds.). (2002). *The vehicle routing problem*. (Vol. 9) Philadelphia, PA: SIAM Monographs on Discrete Mathematics and Applications.
- Toth, P., & Vigo, D. (1997). Heuristic algorithms for the handicapped persons transportation problem. *Transportation Science*, 31, 60–71.