Pickup and multi-delivery problem with time windows

Pham Tuan ANH
National Taiwan University of Science and Technology

Aldy GUNAWAN
Singapore Management University, aldygunawan@smu.edu.sg

Vincent F. YU
National Taiwan University of Science and Technology

Tuan C. CHAU
National Taiwan University of Science and Technology

Pickup and multi-delivery problem with time windows

Pham Tuan Anh[1], Aldy Gunawan[2], Vincent F. Yu[1,3], and Chau Tuan Cuong[1]
[1]Department of Industrial Management, National Taiwan University of Science and Technology, Taipei, Taiwan
[2]School of Computing and Information Systems, Singapore Management University, Singapore, Singapore
[3]Center for Cyber-Physical System Innovation, National Taiwan University of Science and Technology, Taipei, Taiwan
(phamtuananhise@gmail.com, aldygunawan@smu.edu.sg, vincent@mail.ntust.edu.tw, and tuancuong0612@gmail.com)

Abstract. **This paper addresses a new variant of Pickup and Delivery Problem with Time Windows (PDPTW) for enhancing customer satisfaction. In particular, a huge number of requests is served in the system, where each request includes a pickup node and several delivery nodes instead of a pair of pickup and delivery nodes. It is named Pickup and Multi-Delivery Problem with Time Windows (PMDPTW). A mixed-integer programming model is formulated with the objective of minimizing total travel costs. Computational experiments are conducted to test the correctness of the model with a newly generated benchmark based on the PDPTW benchmark instances. Results show that our proposed model is able to solve small-size instances. Alternative approaches for solving larger problems are proposed for future research.**

Keywords. **Pickup and Multiple Delivery Problem, Time Windows, Non-linear constraint, Mixed-Integer Programming**

I. INTRODUCTION

Supply chain networks cover a series of activities and operations to transfer goods from suppliers to end customers. In a supply chain network, logistics plays a key role to manage the efficiency of various planning decisions. Customer demand has not only considerably increased in many recent years, but the level of customer requirements has also seen a rapid rise. This leads to a crucial issue in logistics, especially transportation.

Transportation is one of the best-known logistics problems that should be considered in order to reduce logistics costs. It is critical for operating the flow of goods/products and maintaining customer satisfaction. In modern life, customer requirements have become more and more complicated, and more resources are being consumed to fit these requirements (e.g., time windows, transportation mode, demand, etc.). A typical problem in the transportation field was first introduced by [1], where multiple customers are served by a single pickup location within specific time windows. This model is valuable for some supply chain models that require both pickup and delivery processes, such as last-mile delivery networks, distributor storage with package carrier networks, and third-party logistics (3PL) companies. In these networks, logistics costs occur on a daily basis, where transportation operating cost plays a central part.

To alleviate logistics costs in transportation, vehicle routing problem (VRP) models have been proposed to improve operations and increase economic benefits. Many variants of VRP adopt customer requirements such as the Traveling Salesman Problem (TSP) in [1] and the Multiple Traveling Salesmen Problem (MTSP) in [2] and [3]. These problems utilize a set of salesmen serving a set of customers, where each salesman starts and ends at a single depot. The problems aim to minimize total travel costs without violating some constraints (i.e., each customer node is exactly visited one time).

The capacitated vehicle routing problem is also another extension of VRP, first introduced by [2]. The main constraint must be guaranteed in this problem, which is the loading of each vehicle does not exceed its capacity. Customer requirements in terms of time can be considered as another variant of VRP, called Vehicle Routing Problem with Time Windows (VRPTW). In particular, an additional constraint illustrates that each customer must be served during her/his predefined time windows. VRPTW is also required to meet customer demand when the loading of vehicles is guaranteed.

Another extension of VRP is represented by simultaneously considering pickup and delivery activities. This kind of problem, as referenced in [4] and [5], is inspired by the single-vehicle dial-a-ride problem (DARP) and the pickup and delivery problem with time window (PDPTW), respectively. In DARP, a single vehicle with a fixed capacity will pick up and drop off individuals, and the problem allows multiple pickups before delivery. DARP practically belongs to small-size problems; therefore, these kinds of problems are regularly solved for exact solutions, as in [6]. PDPTW is claimed to be one kind of Traveling Salesman Problem with Pickup and Delivery (TSPPD) (Ref. [7]), by considering more requirements such as time windows and multiple vehicles. Since PDPTW is an NP-hard problem, efficient algorithms have been proposed to solve the PDPTW problem such as an adaptive neighborhood search heuristic (ALNS) [8] and an exact algorithm [9].

An extension of PDPTW describes multi-pickup in the supply chain network and is called the multi-pickup and delivery problem with time windows (PMDPTW) [10]. This problem proposes a formulation with time windows and an adaptive large neighborhood search (ALNS) for the solution.

Almost all VRP models are NP-hard models, which are difficult at solving optimal solutions. To deal with VRP variants, many exact and approximate algorithms have been presented and contributed to further research [11]. In particular, PDPTW is also an NP-hard problem, but leads to limitations in solving exact solutions. Linearization in constraints can also be considered in order to look to reformulate non-linear formulations [12] to linear formulations.

To our best knowledge, an extension of PDPTW with multi-delivery has not been considered yet, and thus we analyze this issue herein. In particular, the pickup and multi-delivery problem with time windows (PMDPTW) simultaneously concerns several properties, including (1) both pickup and delivery of a request must be visited by the same vehicle; (2) all vehicles must start and end at a predefined depot; (3) serving nodes within their time windows must be satisfied; (4) loading of vehicles does not exceed the vehicle capacity; and (5) a request comprises a pickup node and one or several delivery nodes, which are represented by the relation precedence. This problem aims to minimize the total travel cost and provides a set of routes for serving all requests that satisfy time windows and demand constraints. In practice, the proposed PMDPTW can be used by TPL companies, whose products are transferred from distribution centers (DC) and warehouses to retailers and wholesalers.

The main contributions of this study are listed as follows. First, a mathematical programming formulation is developed for solving PMDPTW. Second, constraint linearization is presented to convert the model into a MIP model. Third, new benchmark instances for PMDPTW are generated based on existing benchmark instances of PDPTW. Finally, some experiments are conducted to analyze the limitation of this proposed NP-hard problem and provide conclusions.

II. METHODOLOGY

This section derives the PMDPTW model based on the existing PDPTW formulation [8]. We then modify some constraints associated with the delivery process to obtain a new mathematical model for solving PMDPTW. The model considers multiple delivery nodes for each request instead of only one delivery. PMDPTW is obviously more complicated than PDPTW with additional considerations of multi-delivery for each request. Therefore, the difference between PDPTW and PMDPTW is mainly due to the number of delivery nodes in a request.

- PDPTW: each pickup node corresponds with a delivery node to create a completed request. To illustrate this, we simply use the index of nodes to manage requests.
- PMDPTW: each pickup node corresponds with one or several delivery nodes to create a complete request. For illustration, we propose binary parameters R_{ij} (discussed later in this section) to manage requests.

Considering more information on the proposed problem is the reason why the PMDPTW problem obviously becomes more complicated. We can say that the PMDPTW formulation is an extension of PDPTW by modifying constraints related to requests.

The problem instances of PMDPTW contain n requests and a set of m vehicles for serving requests. Each request includes a pickup node and several delivery nodes. Each vehicle can be used to serve one or several requests. Let P be the set of pickup nodes, D be the set of delivery nodes, and V be the set of vehicles. A solution to PMDPTW is a set of feasible routes for serving all requests by a set of vehicles starting and ending at the depot. Section A presents the PDPTW formulation in detail, while Section B discusses the modifications for deriving the PMDPTW model.

A. Three-index formulation of PDPTW

Indices:

i,j: index of nodes (i.e., pickup or delivery)
k: index of vehicles

Sets:

Each request i is associated with a pickup node i and a delivery node $n+i$
N: a set of nodes from 0 to $2n+1$
P: a subset of nodes from 1 to n represents pickup nodes
D: a subset of nodes from $n+1$ to $2n$ represents delivery nodes
K: a set of vehicles from 1 to m

Parameters:

c_{ij}: transportation cost from i^{th} node to j^{th} node
q_i: load/unload quantity of i^{th} node
a_i: earliest time of i^{th} node
b_i: latest time of i^{th} node
t_{ij}: transportation time from i^{th} node to j^{th} node

Decision Variables:

x_{ij}^k: 1 for k^{th} vehicle transport from i^{th} node to j^{th} node and otherwise 0
B_i^k: starting time of k^{th} vehicle at i^{th} node
Q_i^k: current quantity of k^{th} vehicle at i^{th} node

Objective function:

$$\textbf{Minimize} \sum_{k\in K}\sum_{i\in N}\sum_{j\in N} c_{ij}x_{ij}^k \quad (1)$$

Subject to:

$$\sum_{k\in K}\sum_{j\in N} x_{ij}^k = 1 \; \forall\, i \in P \quad (2)$$

$$\sum_{j\in N} x_{ij}^k - \sum_{j\in N} x_{n+i,j}^k = 0 \; \forall\, i \in P, k \in K \quad (3)$$

$$\sum_{j\in N} x_{0j}^k = 1 \; \forall\, k \in K \quad (4)$$

$$\sum_{i\in N} x_{i,2n+1}^k = 1 \; \forall\, k \in K \quad (5)$$

$$\sum_{j\in N} x_{ji}^k - \sum_{j\in N} x_{ij}^k = 0 \; \forall\, i \in P\cup D, k \in K \quad (6)$$

$$B_j^k \ge \left(B_i^k + t_{ij}\right)x_{ij}^k \; \forall\, i \in N, j \in N, k \in K \quad (7)$$

$$Q_j^k \ge \left(Q_i^k + q_j\right)x_{ij}^k \; \forall\, i \in N, j \in N, k \in K \quad (8)$$

$$B_i^k + t_{i,n+i} \le B_{n+i}^k \; \forall\, i \in P \quad (9)$$

$$a_i \le B_i^k \le b_i \; \forall\, i \in N, k \in K \quad (10)$$

$$max\{0, q_i\} \le Q_i^k \le \min\{Q, Q+q_i\} \; \forall\, i \in N, k \in K \quad (11)$$

$$x_{ij}^k \in \{0,1\} \; \forall\, i \in N, j \in N, k \in K \quad (12)$$

The objective function (1) aims to minimize the total travel costs during the pickup and delivery processes. Constraint (2) ensures that each pickup node must be visited exactly once. Constraint (3) declares that nodes in the same request (i.e., pickup and delivery nodes) must be served by the same vehicle. It forces each request to be served exactly once and by the same vehicle. Constraints (4) and (5) guarantee that every vehicle must start and end at the depot. Constraint (6) represents the flow conservation of each route. Sub-tour eliminations in terms of time and loading variables are maintained by Constraints (7) and (8). Regarding each request, Constraint (9) ensures that the delivery node must be visited later by its pickup node. Constraint (10) makes sure that each node is served within its predefined time window. Finally, Constraint (11) imposes that the loading of each vehicle cannot exceed the vehicle capacity.

The proposed set-up is a mixed-integer non-linear programming (NLP) model due to Constraints (7) and (8). Since NLP is complicated to solve and consumes large computational times, it could be linearized by some reformulation techniques. Constraints (13) and (14) are then derived by linearizing Constraints (7) and (8), respectively. Let $BigM$ be a very large number and added into the inequalities to handle either-or constraints.

$$B_j^k \geq B_i^k + t_{ij} - BigM\left(1 - x_{ij}^k\right) \forall\, i,j \in N, k \in K \quad (13)$$

$$Q_j^k \geq Q_i^k + q_j - BigM\left(1 - x_{ij}^k\right) \forall\, i,j \in N, k \in K \quad (14)$$

B. Modifications to PMDPTW

The modifications in PMDPTW are mainly caused by the information of requests, where each request comprises a pickup node and a set of delivery nodes. In order to illustrate the relation between pickup nodes and delivery nodes in requests, we propose a two-index parameter for expressing this relationship, denoted by R_{ij} (refer to Table II later). Let R_{ij} indicate the relation between node $i \in P$ and node $j \in D$. The specific modifications in parameters and constraints are further developed as follows.

Sets:

N: a set of nodes from 0 to $n+1$
P: a subset of nodes from 1 to p represents pickup nodes
D: a subset of nodes from $p+1$ to n represent delivery nodes

Parameters:

R_{ij}: binary parameter; $R_{ij} = 1$ if node $i \in P$ is the pick-up node of node $j \in D$ and otherwise $R_{ij} = 0$

Based on the modifications, the PMDPTW formulation is illustrated as follows.

Objective function: (1)

Subject to: (2)-(6), (9)-(12), (13), (14), and:

$$R_{ij}\left(\sum_{h \in N} x_{ih}^k - \sum_{h \in N} x_{j,h}^k\right) = 0 \;\forall\, i \in P, k \in K \quad (15)$$

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

$$B_i^k + R_{ij} t_{i,j} \leq B_j^k \;\forall\, i \in P, j \in D \quad (17)$$

III. COMPUTATIONAL RESULTS

This section generates an example instance to test the PMDPTW formulation. A commercial solver (i.e., CPLEX solver) is then used for solving the PMDPTW model. Moreover, the feasibility of solutions in terms of loading, time windows, and vehicle capacity is also taken into account. Finally, computational results are further analyzed in terms of resource consumption (i.e., computational time).

A. Benchmark instances

An example for generating benchmark instances is now represented. We consider a supply chain network including 3 pickup nodes, 10 delivery nodes, and a depot (i.e., node 0 and a dummy node 14). Detailed information about each node is provided as follows:

- *Coordinate*: the location of each node is defined on the x-y axis. Distance and time matrix are then calculated based on the Euclidean distance between each pair of locations. This is generated in the range $[50 \times 50]$.
- *Service time*: service is ignored in this example, which means it equals 0.
- *Supply/demand quantity of pickup/delivery nodes*: Regarding supply quantity (for pickup nodes), those values are represented by a *positive* integer number. Regarding demand quantity (for delivery nodes), those values are represented by a *negative* integer number.
- *Time windows*: each node has a predefined time window, which must be served within the time window interval. This interval is defined by the earliest time (denoted by a) and the latest time (denoted by b). The planning time horizon is described by the time window of the depot $[0,1000]$. Note that $a < b$ must be guaranteed and interval $[a, b]$ must be in the planning time horizon when generating data

A set of 5 vehicles with a fixed capacity is provided in the network. Details are shown in Table I. Regarding *requests* in the system, each request comprises a pickup node and one or several delivery nodes. Binary parameters R_{ij} representing the relationship between pickup and delivery nodes are generated as shown in Table II.

B. Solutions

We use a commercial solver (i.e., CPLEX solver) to implement the PMDPTW model. Results from a solution include a set of feasible paths, cumulated loading Q_i^k of vehicle k at node i, and arrival time B_i^k of vehicle k at node i. To verify the correctness of the PMDPTW formulation, those above-mentioned results should be checked in detail.

Let us analyze the above example. The model provides the optimal solution, which uses two routes for serving all requests, with the total travel cost being 379. In particular, the first route is **0 – 1 – 4 – 3 – 8 – 5 – 10 – 7 – 9 – 11 – 14** and the second route is **0 – 2 – 12 – 6 – 13 – 14**. Figure 1 illustrates the sequences of the two routes in detail. The sequences satisfy the flow constraints: each vehicle starts and ends at the depot, pickup nodes must be visited before associated delivery nodes (the same request with the pickup node), and each node is visited exactly once.

TABLE I
AN SAMPLE DATASET FOR COMPUTATIONAL EXAMPLE

Node	Coordinate		Capacity	Vehicles	Velocity	
			40	5	1	
	<x>	<y>	<service time>	<demand>	<a>	<b>
0	25	25	0	0	0	1000
1	4	10	0	23	0	84
2	27	36	0	31	36	96
3	18	42	0	22	50	142
4	50	29	0	-8	20	200
5	31	29	0	-7	150	200
6	7	22	0	-19	97	157
7	21	10	0	-5	316	376
8	0	49	0	-12	125	185
9	47	8	0	-5	414	474
10	41	42	0	-3	256	316
11	20	34	0	-5	469	529
12	36	47	0	-7	209	269
13	44	47	0	-5	261	321
14	25	25	0	0	0	1000

TABLE II
RELATIONSHIP BETWEEN PICKUP AND DELIVERY NODES

R_{ij}	4	5	6	7	8	9	10	11	12	13
1	1	1	0	1	0	0	1	0	0	0
2	0	0	1	0	0	0	0	0	1	1
3	0	0	0	0	1	1	0	1	0	0

The arrival time and cumulated loading at each node (B_i^k and Q_i^k) are represented as follows (see Table 3 in detail). Regarding the arrival time of each node, it must be satisfied to be visited within the time window. Regarding the loading of vehicles, it does not exceed the vehicle capacity at any cumulated loading. Therefore, the PMDPTW model is valid.

TABLE III
ARRIVAL TIME AND CUMULATIVE LOADING OF EACH NODE

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
B_i^k	0	84	0	142	107	198	0	316	161	414	256	469	0	0	479
	0	0	73	0	0	0	97	0	0	0	0	0	253	261	290
Q_i^k	0	23	0	37	15	18	0	10	25	5	15	0	0	0	0
	0	0	31	0	0	0	12	0	0	0	0	0	5	0	0

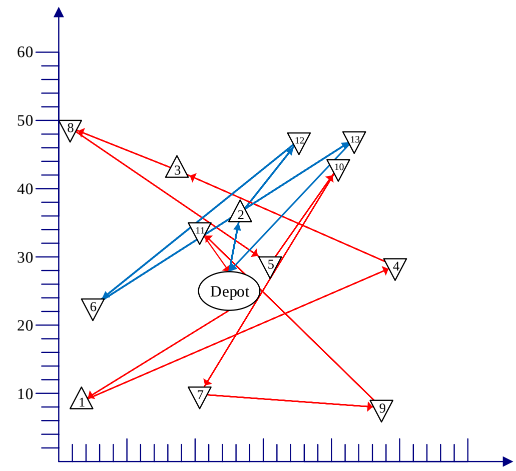

Fig. 1. An example solution for PMPDPTW

C. *Computational analyses*

Since PMDPTW is an NP-hard problem, the computational time is the burden. Thus, an experiment is conducted by increasing the size of the problem (i.e., the number of nodes/requests is increased step by step 8-9-14-18-20-30, up to 30 nodes). Solutions from instances are also provided by using the CPLEX solver. However, we consider the computational time instead of the quality of solutions. Elapsed time for solving instances follows the exponential function with respect to the number of nodes. Detailed results are shown in Figure 2.

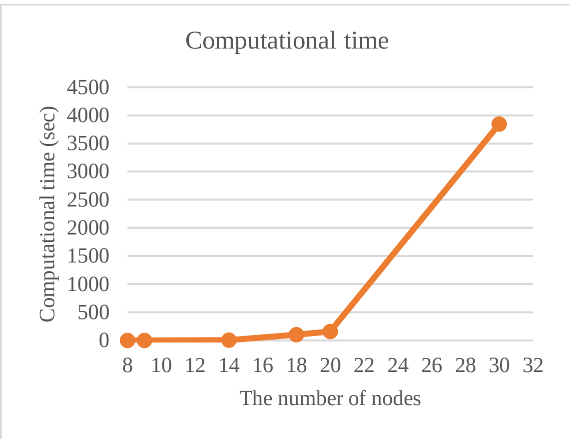

Fig. 2. Graph of time related to the number of nodes

IV. CONCLUSION

This paper introduces a new formulation of the Pickup and Multi-Delivery Problem with Time Windows (PMDPTW) and utilizes the linearization method in the model. In this model we extend the ability to solve problems with requests under multi-delivery locations so as to enhance the efficiency of logistics activities. For solving the problem, the tuple approach is applied to alleviate the burden of computational time.

Numerical experiments are then conducted to validate the correctness of the PMDPTW formulation and to analyze the pattern of the computational time with respect to the size of the problem. Future work could focus on two main streams (i.e., extending concepts and developing efficient approaches). Regarding an extension, we can consider more specific requirements of customers (e.g., delivery options, operation time, etc.). In terms of methods or approaches, we can propose new techniques such as metaheuristics, metaheuristics, branch-and-price-and-cut algorithm, etc.

References

[1] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Manage. Sci.*, vol. 6, no. 1, pp. 80–91, 1959.

[2] G. Clarke and J. W.Wright, "Scheduling of vehicles from a central depot to a number of delivery points," *Oper. Res.*, vol. 12, no. 4, pp. 568–581, 1964.

[3] S. Lin, "Computer solutions of the traveling salesman problem," *Bell Syst. Tech. J.*, vol. 44, no. 10, pp. 2245–2269, 1965.

[4] J. Desrosiers, Y. Dumas, and F. Soumis, "A dynamic programming solution of the large-scale single-vehicle dial-a-ride problem with time windows," *Am. J. Math. Manag. Sci.*, vol. 6, no. 3–4, pp. 301–325, 1986.

[5] T. R. Sexton and Y. M. Choi, "Pickup and Delivery of Partial Loads with 'Soft' Time Windows," *Am. J. Math. Manag. Sci.*, vol. 6, no. 3–4, pp. 369–398, Feb.1986.

[6] G. R. Beck Jr, "Inorganic phosphate as a signaling molecule in osteoblast differentiation," *J. Cell. Biochem.*, vol. 90, no. 2, pp. 234–243, 2003.

[7] G. Berbeglia, J. F. Cordeau, and G. Laporte, "Dynamic pickup and delivery problems," *Eur. J. Oper. Res.*, vol. 202, no. 1, pp. 8–15, 2010.

[8] S. Ropke and D. Pisinger, "An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows," *Transp. Sci.*, vol. 40, no. 4, pp. 455–472, 2006.

[9] R. Baldacci, E. Bartolini, and A. Mingozzi, "An exact algorithm for the pickup and delivery problem with time windows," *Oper. Res.*, vol. 59, no. 2, pp. 414–426, 2011.

[10] S. Naccache, J. F. Côté, and L. C. Coelho, "The multi-pickup and delivery problem with time windows," *Eur. J. Oper. Res.*, vol. 269, no. 1, pp. 353–362, 2018.

[11] G. Laporte, "The vehicle routing problem: An overview of exact and approximate algorithms," *Eur. J. Oper. Res.*, vol. 59, no. 3, pp. 345–358, 1992.

[12] J. F. Cordeau, G. Desaulniers, J. Desrosiers, M. M. Solomon, and F. Soumis, "VRP with Time Windows," in *The vehicle routing problem*, 2001, pp. 157–193.