The Multi-Commodity Multi-Vehicle Minimum Latency Problem

Mohammad Moshref-Javadi and Seokcheon Lee

School of Industrial Engineering, Purdue University, 315 N. Grant St., West Lafayette, IN 47907, USA

Abstract

We propose a novel problem, so-called Knapsack-Minimum Latency Problem (Knapsack-MLP), whose objective is to minimize the total waiting time along the lines of the conventional Traveling Repairman Problem. In this problem, multiple capacitated vehicles distribute, from a supply depot, multiple types of commodities to customers. Two classes of decisions are involved: determine the quantities of commodities to load on each of the vehicles and select the sequence of visits to the customers and supply depot. The problem is formulated in two different Mixed-Integer Linear Program (MILP) models. These models are developed based on the formulations of network flow problem and assignment problem. The latter formulation is also used to yield good upper bound. The Knapsack-MLP is useful for distributing commodities in customer-oriented logistics systems.

Keywords
Minimum latency problem, deliveryman problem, traveling repairman problem, logistics, routing

1. Introduction

Logistics and transportation are parts of production, manufacturing, and service systems. These could be inside and outside facilities and among different manufacturing or service systems. Transportations and logistics cost could include even 70% of the final product costs [1]. Better, efficient logistics contributes to higher performance of the system and lower transportation wastes which result in lower costs of the final products.

One of the major problems in transportation and logistics is dispatching vehicles in order to distribute commodities to customers. This includes distribution of not only products, but also services. The problem of scheduling visits of the vehicles to customers is called the Routing problem. There are many sub-classes of routing problems, which include the Vehicle Routing Problem (VRP) [2-3], Traveling Salesman Problem (TSP) [4-5], Chinese Postman Routing Problem [6], and Minimum Latency Problem [7]. In this paper we focus on the Minimum Latency Problem.

The Minimum Latency Problems (MLP) is a class of routing problems. The basic problem in this class is the Traveling Repairman Problem (TRP), also known as the Deliveryman Problem [8]. This problem is fairly new in comparison with the TSP, which is the basic problem in the class of Vehicle Routing Problems. The TSP and VRPs were developed from the 1950s, while the TRP and MLPs were examined from the 1980s. In the TRP, there is one vehicle and the vehicle has to visit every node exactly once. The objective is to minimize the total waiting time of customers. On the other hand, the TSP minimizes the length of tour of the vehicle. Generally speaking, VRPs are server-oriented, while MLPs stand from the customer point of view.

MLPs have considerable real world applications, such as:

- Humanitarian logistics: distribution of food, water, and clothes to affected areas; dispatching ambulances to patients’ locations [9]
- Machine scheduling problems: scheduling jobs on sequence-dependent setup time machine [10]
MLPs have recently received significant attention in research and practice since 1986: Minimum Latency Problem on a straight line [13-14], minimum latency problems with non-zero service time [15-16], Dynamic Minimum Latency Problems [17-18], MLP with time window [15, 19-20], scatter pattern of customer nodes in distribution networks [21], multiple vehicle MLP [9, 16, 20, 22-26], capacitated vehicles [12, 24, 26-28], and directed MLP [25, 29-30]. Some authors, also, proposed latency-based objective functions different from the sum of waiting times: minimize maximum latency [9, 14], minimize variation from target time [31], and minimize profit-based latencies [23, 32-33].

Several formulations have been developed for the class of Minimum Latency Problems. Bianco et al. [34] formulated the TRP based on the network flow problem formulation. Two exact algorithms were presented to solve the problem. Heilporn et al. [19] gave two formulations for the TRP with time window. The first formulation uses the arc flow problem model and the second formulation utilizes the sequential assignment model. An exact algorithm was proposed for the problem using the polyhedral analysis. Ngueveu et al. [24] proposed a mixed integer formulation of the capacitated multi-vehicle MLP also known as the Cumulative Capacitated Vehicle Routing Problem (CCVRP). A Memetic algorithm was designed and used to deal with this problem. Angel-Bello et al. [30] proposed two formulations for the directed TRP. The mathematical models were built using permutation-based decision variables. Bjelic et al. [20] formulated the heterogeneous TRP with time window. The formulation uses the arc flow network model. Authors utilized a metaheuristic approach based on the Variable Neighborhood method. Lysgaard and Wohlk [26] formulated the capacitated multi-vehicle MLP using set partitioning formulation and proposed the first exact algorithm for the problem. The reader is referred to [35] for a taxonomy and review of the MLPs that includes MLP characteristics, objective functions and solution approaches.

The review of the literature indicates that several mathematical models have been proposed for MLPs and two problems have been the focus of research as follows:

- Traveling Repairman Problem: assumes a single vehicle distributing a single type of commodity to all customers. The vehicle’s capacity is assumed to be infinite.
- Capacitated Multi-vehicle MLP: assume multiple capacitated vehicles distributing a single type of commodity to all customers.

However, there are many real cases that multiple types of commodities could be requested by customers. For example, food, water, clothes, and other supplies need to be distributed to aid victims in disaster relief operations. In this paper, we propose two mathematical models for the multi-commodity multi-vehicle capacitated Minimum Latency Problem, which we call the Knapsack-Minimum Latency Problem (Knapsack-MLP).

The remainder of the paper is organized as follows. Section 2 presents the problem description and assumptions. Two problem formulations based on network flow and assignment models are presented in section 3. The proposed models are compared in section 4 and, finally, section 5 concludes the paper.

### 2. Problem Description and Assumptions

The considered problem in this paper is multi-commodity multi-vehicle Minimum Latency Problem, which we call Knapsack-MLP. The Knapsack-MLP assumes a network of customers with multiple vehicles starting their tours from a single depot and distributing commodities to the customers. Each customer can request multiple types of commodities with different quantities. Vehicles’ capacities are assumed to be limited and can be different. The objective of the problem is to minimize the sum of waiting time of the customers. Thus, two key decisions are made in this problem. The first decision is how many of each commodity must be loaded on each vehicle. The second decision is to find the set of customers serviced by each vehicle and the sequence of visits to customers. If we assume each vehicle is a knapsack, we decide about the set and quantities of commodities loaded on each vehicle. This is the reason why we call this problem the Knapsack-MLP. It is assumed that the capacity of the vehicle is large enough to satisfy customer’s demand in one visit. If a customer requests more than capacity of a vehicle, the problem is infeasible. Nevertheless, we propose some modifications to our models to relax this assumption so that a
Moshref-Javadi and Lee

customer can be visited by multiple vehicles and the demand is split among those vehicles. More assumptions of the problem are as follows:

- Locations of customers and depot are known
- Travel time between customers’ locations and between customers and depot is deterministic and known
- Vehicles start tours from the depot, visit a subset of customers and return to the depot in the end
- Vehicles are able to carry all types of commodities
- Refill is not allowed
- Vehicles are heterogeneous in terms of capacity
- Customers are heterogeneous in terms of requested commodities
- Service time at customer location can be zero or non-zero

Figure 1: Problem illustration.

Figure 1 illustrates the problem schematically. In this figure, there is one depot with two vehicles distributing three types of commodities to customers. The numbers in the boxes of the customers indicate the demand quantities of every commodity by the customers. Furthermore, the numbers in the boxes of each vehicle represent the quantities of commodities loaded on the vehicle. Two routes have been calculated, one for every vehicle, with the minimum total waiting time of customers. Thus, the two decisions made in the problem give the number of products to be loaded on each vehicle, as well as the set of customers with the sequence of visits of each vehicle.

3. The Proposed Mathematical Models

In this section, we present two mathematical formulations of the problem.

3.1 Formulation 1

Suppose $G(V,A)$ is a graph where $V$ is the set of vertices ${v_0, v_1, \ldots, v_N}$ including depot ($v_0$) and $A$ is the set of arcs $\{(v_i, v_j) : v_i, v_j \in V, v_i \neq v_j\}$. We also add a dummy node $v_{N+1}$ to be able to start and end all tours at depot. Thus, both $v_0$ and $v_{N+1}$ represent the depot. Corresponding to every arc, there is $c_{ij}$ which indicates the traveling time of the arc between nodes $v_i$ and $v_j$. This model is based on formulations of network flow problem. In this paper, since customers can be assumed on a network, we use the terms node and customer interchangeably. To formulate the problem, the following notations are defined:
Indices

- \(i, j\): 1, \ldots, \(N\) represent customers. 0 and \(N+1\) represent depot.
- \(k\): Represents vehicle
- \(c\): Represents commodity

Sets

- \(K\): Set of vehicles
- \(C\): Set of commodities
- \(V\): Set of all customers and depot
- \(V'\): Set of customers

Parameters

- \(q_c\): Volume of commodity \(c\)
- \(Q_k\): Capacity of vehicle \(k\)
- \(c_{ij}\): Travel time between node \(i\) and \(j\)
- \(M\): Large positive constant
- \(D_{ic}\): Demand quantity of customer \(i\) of commodity \(c\)

Variables

- \(t_i^k\): Arrival time of vehicle \(k\) at customer \(i\)
- \(x_{ij}^k\):\( \begin{cases} 1 & \text{if vehicle } k \text{ traverses arc } (i,j) \text{ from customer } i \text{ to customer } j \\ 0 & \text{otherwise} \end{cases}\)
- \(z_{ic}^k\): Non-negative variable. The number of commodity \(c\) transported to customer \(i\) by vehicle \(k\).

Formulation 1 is developed based on the formulation of capacitated multi-vehicle MLP presented in [24]. Our model extends their model to the multi-commodity case. Furthermore, we consider vehicle sharing and demand split in the problem. The proposed formulation of the problem is as follows:

Minimize: \[\sum_{k \in K} \sum_{i \in V'} t_i^k\] \hspace{1cm} (1)

\[\sum_{i \in V} x_{ij}^k = \sum_{j \in V'} x_{ji}^k \quad \forall j \in V', \forall k \in K\] \hspace{1cm} (2)

\[\sum_{k \in K} \sum_{i \in V, j > 0} x_{ij}^k = 1, \quad \forall i \in V'\] \hspace{1cm} (3)

\[\sum_{c \in C} \sum_{i \in V'} z_{ic}^k q_c \leq Q_k, \quad \forall k \in K\] \hspace{1cm} (4)

\[\sum_{j \in V'} x_{0j}^k \leq 1 \quad \forall k \in K\] \hspace{1cm} (5)

\[\sum_{i \in V'} x_{i,N+1}^k \leq 1, \quad \forall k \in K\] \hspace{1cm} (6)

\[M \sum_{j \neq i} x_{ij}^k \geq z_{ic}^k, \quad \forall k \in K, i \in V', c \in C\] \hspace{1cm} (7)

\[\sum_{k \in K} z_{ic}^k = D_{ic}, \quad \forall i \in V', c \in C\] \hspace{1cm} (8)
In this model, the objective function is to minimize the total sum of latencies at customers. Constraints (2) represent flow continuity in tours. Constraints (3) ensure that each node is the predecessor of only one node in the tour of exactly one vehicle. Constraints (4) guarantee that the total amount of demands satisfied by every vehicle does not exceed the capacity of the vehicle. Constraints (5-6) indicate that all tours must start and ends at the depot. Constraints (7) relate the variables of the problem. Constraints (8) ensure that all of the demands are satisfied. Latency at each node is calculated using constraints (9). Note that this constraint eliminates cycles as well [24]. Finally, constraints (10) give the ranges and types of the variables.

If the service time at each customer is also considered, constraint (9) changes to:

\[
t_k^i + c_{ij} + r_j^k - (1 - x_j^k)M \leq t_j^k, \quad \forall i \in V \setminus \{N+1\}, \forall j \in V \setminus \{0\}, \forall i \neq j, \forall k \in K
\]

where \( r_j^k \) is the service time of vehicle \( k \) at customer \( i \).

**Sharing vehicles**: due to limited capacities of vehicles, multiple vehicles could be needed to satisfy demands of a customer. The proposed model can be modified to incorporate the case in which demands can be split. Under these circumstances, if a vehicle does not have enough capacity to transport all of the requested commodities to a customer, the vehicle satisfies a portion of the customer’s demand and the rest of demand will be transported by at least one other vehicle. In this case, the problem can be modeled first by eliminating constraint (3). In addition, variable \( z_{ic}^k \) is now defined as an integer. Otherwise, \( z_{ic}^k \) would be assigned a non-integer which is not reasonable. The problem which now arises is the calculation of latency at customers whose demands are satisfied by multiple vehicles. More specifically, the latency at node \( i \) \((t_k^i)\) is calculated as soon as a vehicle visits the customer, and the latency at the node becomes the sum of these latencies. This does not truly represent latency at a node based on our earlier definition. Thus, we define the latency as the latest time that all of the requested commodities are delivered to the node. Therefore, latency at node \( i \) is defined as follows:

\[
\text{Latency at node } i: \max_{k} \{ t_k^i \}
\]

Thus, the objective function is defined as:

\[
\min \sum_{i \in V^\prime} \max_{k} \{ t_k^i \}
\]

This non-linear objective function could be linearized by defining variable \( u_i \) and adding the following set of constraints:

\[
u_i \geq t_k^i, \quad i \in V^\prime, k \in K
\]

Hence, the linear objective function is:

\[
\min \sum_{i \in V^\prime} u_i
\]

### 3.2 Formulation 2

The second model uses the assignment problem model. A position-based formulation was proposed by Angel-Bello et al. [30]. Our formulation is different, specifically, with respect to the definition of variable. In the proposed model, \( N+1 \) positions, \( \{h_0, h_1, ..., h_N\} \), are defined for each tour. Dummy position \( h_{N+i} \) is defined in correspondence to dummy node \( v_{N+i} \). Nodes \( \{v_0, v_1, ..., v_{N+i}\} \) are assigned to the defined positions. Initially, it is assumed that each
node can only be visited by exactly one vehicle. Later, we extend the model to incorporate demand split. Hence, each customer may be visited by multiple vehicles. Since all tours start and end at depot, node \( v_0 \) is assigned to position \( h_0 \) and node \( v_{N+1} \) is assigned to position \( h_{N+1} \) in all tours. In addition to the defined notations, the following notations are used:

### Indices
- \( h \): Position: \( 0,...,N+1 \)

### Sets
- \( H \): Set of all positions
- \( H' \): Set of all positions, except positions \( h_0 \) and \( h_{N+1} \)

### Variables
- \( x_{hk}^i \): \( \begin{cases} 1 & \text{if vehicle } k \text{ visits customer } i \text{ in position } h \text{ in its tour} \\ 0 & \text{otherwise} \end{cases} \)

Using the defined notations, the problem is formulated as the following:

### Objective Function

Minimize: \( \sum_{k \in K} \sum_{i \in V'} t_k^i \)  

### Constraints
1. \( \sum_{k \in K} x_{hk}^i = 1 \quad \forall i \in V' \) \hspace{1cm} (17)
2. \( x_{0k}^i = 1, \quad \forall k \in K \) \hspace{1cm} (18)
3. \( x_{N+1,k}^i = 1, \quad \forall k \in K \) \hspace{1cm} (19)
4. \( \sum_{i \in V} x_{hk}^i \leq 1, \quad \forall k \in K, \forall h \in H \) \hspace{1cm} (20)
5. \( \sum_{i \in V} x_{hk}^i \geq x_{h+1,k}^i, \quad \forall k \in K, \forall h \in H \setminus \{N, N+1\} \) \hspace{1cm} (21)
6. \( \sum_{c \in C} z_{ic}^k q_c \leq Q_k, \quad \forall k \in K \) \hspace{1cm} (22)
7. \( M \sum_{h \in H'} x_{hk}^i \geq z_{ic}^k, \quad \forall k \in K, i \in V', c \in C \) \hspace{1cm} (23)
8. \( \sum_{k \in K} z_{ic}^k = D_{ic}, \quad \forall i \in V', c \in C \) \hspace{1cm} (24)
9. \( t_i^k + c_{ij} - (1-x_{hk}^i x_{h+1,k}^i)M \leq t_j^k, \quad \forall i \in V, \forall j \in V, \forall i \neq j, \forall k \in K, h \in H \setminus \{N+1\} \) \hspace{1cm} (25)
10. \( t_i^k, z_{ic}^k \geq 0, \quad \forall i \in V, \forall k \in K, \forall c = 1,...,C \) \hspace{1cm} (26)
11. \( x_{hk}^i \in \{0,1\}, \quad \forall i \in V, k \in K, h \in H \)

The objective function minimizes the sum of the latencies at all customers. Constraint (17) indicates that each customer can be assigned to exactly one position in exactly one tour. Constraints (18) and (19) ensure that each tour starts and ends at the depot. Constraint (20) guarantees that at most one customer is assigned to each position of a tour. Constraint (21) ensures that the positions are filled by customers and depot without any void (unoccupied position) between them and they must be occupied from position 0, then 1, then 2, and so forth. Constraint (22) represents the capacity constraint of the vehicle. Constraint (23) relates the variables and constraint (24) ensures that all demands are satisfied. Constraint (25) calculates the waiting time of customers. Finally, constraint (26) gives the types and ranges of the variables.
Sharing vehicles: Similar to Formulation 1, this model is modified to consider demand split and shared vehicles. To consider this case, constraint (17) is eliminated and objective function (15) is used with constraints (14) to correctly represent the total latencies at nodes. Note that in formulating routing problems based on the assignment problem formulation, objective function can be calculated by:

\[
\text{Minimize: } \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} (\eta_k - h + 1)x_{hk}^i x_{h+1,k}^j c_{ij}
\]

(27)

where \( \eta_k \) is the number of customers serviced by vehicle \( k \) in optimal solution. As a result, computations time could be decreased by removing constraints (25). If only a single vehicle is used, all customers are assigned to the vehicle and \( \eta_k = N. \) On the other hand, assuming multiple vehicles, this formula cannot be used for calculation of latencies since each vehicle can have a different \( \eta_k \) which differs from \( N. \) As a result, this formula cannot be used and is only valid in the case of single vehicle.

The non-linear term \( x_{hk}^i x_{h+1,k}^j \) in equations (25) can be linearized using the following proposition.

**Proposition.** The non-linear term \( x_{hk}^i x_{h+1,k}^j \) is transformed to binary variable \( U_{hk}^{ij} \) by adding the following set of constraints:

\[
\begin{align*}
{x}_{hk}^i + {x}_{h+1,k}^j - U_{hk}^{ij} &\leq 1 & &\forall i \in V, \forall j \in V, \forall i \neq j, \forall k \in K, h \in H \\
{x}_{hk}^i + 2U_{hk}^{ij} &\geq 0 & &\forall i \in V, \forall j \in V, \forall i \neq j, \forall k \in K, h \in H
\end{align*}
\]

(28) (29)

**Proof.** Using Koon et al. [36] approach, if \( x_{hk}^i x_{h+1,k}^j = U_{hk}^{ij} \), then four possible cases are:

(a) If \( x_{hk}^i = x_{h+1,k}^j = 1 \), then \( x_{hk}^i x_{h+1,k}^j = 1 \). Under constraint (28) \( U_{hk}^{ij} = 1 \).

(b) If \( x_{hk}^i = x_{h+1,k}^j = 0 \), then \( x_{hk}^i x_{h+1,k}^j = 0 \). Under constraint (29) \( U_{hk}^{ij} = 0 \).

(c) and (d) If one of variables \( x_{hk}^i \) and \( x_{h+1,k}^j \) is 0 and the other one is 1, then \( x_{hk}^i x_{h+1,k}^j = 0 \) and \( x_{hk}^i + x_{h+1,k}^j = 1 \). Under constraint (29) \( U_{hk}^{ij} = 0 \).

The following theorem is proved for the Knapsack-MLP.

**Theorem.** At most one demand is split between each pair of routes assuming triangular constraint.

**Proof.** Suppose two nodes \( i_1 \) and \( i_2 \) whose demands are split between two vehicles \( k_1 \) and \( k_2 \). We use a similar approach to [37] to prove this theorem. Assume \( \varphi_{i_1}^{k_1} \) and \( \varphi_{i_1}^{k_2} \) are commodities delivered to customer \( i_1 \) by vehicles \( k_1 \) and \( k_2 \). Also, \( \varphi_{i_2}^{k_1} \) and \( \varphi_{i_2}^{k_2} \) are commodities delivered to customer \( i_2 \) by vehicles \( k_1 \) and \( k_2 \). If the capacity of the vehicles is \( Q \), it is obvious that \( \varphi_{i_1}^{k_1} + \varphi_{i_2}^{k_1} \leq Q \) and \( \varphi_{i_1}^{k_2} + \varphi_{i_2}^{k_2} \leq Q \). Assume, without loss of generality, \( \varphi_{i_1}^{k_1} = \min \{ \varphi_{i_1}^{k_1}, \varphi_{i_2}^{k_1}, \varphi_{i_2}^{k_2}, \varphi_{i_1}^{k_2} \} = 0 \). Thus, in the new solution, we have, \( \varphi_{i_1}^{k_1} = 0 \), \( \varphi_{i_1}^{k_2} = \varphi_{i_2}^{k_1} + \varphi_{i_2}^{k_2} \), \( \varphi_{i_2}^{k_2} = Q - \varphi_{i_1}^{k_2} = Q - \varphi_{i_1}^{k_1} - \varphi_{i_1}^{k_2} \), and \( \varphi_{i_1}^{k_1} = \varphi_{i_2}^{k_1} + \varphi_{i_2}^{k_2} - \varphi_{i_2}^{k_2} = \varphi_{i_2}^{k_1} + \varphi_{i_1}^{k_2} - Q + \varphi_{i_1}^{k_1} + \varphi_{i_1}^{k_2} \). We need to check the demand satisfaction constraints:

\[
\begin{align*}
\varphi_{i_1}^{k_1} + \varphi_{i_1}^{k_2} &= 0 + \varphi_{i_1}^{k_1} + \varphi_{i_2}^{k_2} \\
\varphi_{i_2}^{k_1} + \varphi_{i_2}^{k_2} &= Q - \varphi_{i_1}^{k_1} - \varphi_{i_1}^{k_2} + \varphi_{i_2}^{k_1} + \varphi_{i_2}^{k_2} - Q + \varphi_{i_1}^{k_1} + \varphi_{i_1}^{k_2} = \varphi_{i_2}^{k_1} + \varphi_{i_2}^{k_2}
\end{align*}
\]

(30) (31)

Also, the capacity constraint for \( k_1 \) is:
Formulation 2 assumes that there are $N+2$ positions on every tour. However, only $\eta_k$ of these positions are filled with customer nodes and the remaining positions are not occupied. Thus, we could reduce the number of defined positions on all tours to $\lceil N/R \rceil + 2$ in which $R$ is the number of vehicles. However, by defining this upper bound the formulation is not able to yield optimal solution when $\eta_k > \lceil N/R \rceil$ for at least one $k$ ($\eta_k$ is the number of customers visited by vehicle $k$ in optimal solution). The generated problems were used to examine performance of the upper bound. The results are given in Table 2. Gap is defined as: (UB-Opt)/Opt. The results show that the upper bound is able to reach optimal solution in averagely 0.26 seconds in all problems with 5 customers. Also, the method obtains solution within averagely 4.91% of optimal solution in 74.39 seconds in problems with 10 customers. Although there are gaps between the obtained and optimal solutions, but significant reduction in computations time was achieved.
5. Conclusions

In this paper, we considered the multi-commodity multi-vehicle minimum latency problem, called the Knapsack-MLP. Two key decisions are made in this problem: 1- the quantity of each commodity that is loaded on each vehicle and 2- the set of customers which are serviced by each vehicle and the sequence of visits to customers. We proposed two formulations of the problem. The formulations could also be used when demands are split between vehicles. In this problem, multiple vehicles are used to deliver products to a customer. The formulations also cover the cases in which demand at a node is greater than capacities of vehicles. The proposed formulations were modified such that they represent latency correctly considering multiple commodities. We compared the formulations on 20 generated problems. The results showed that Formulation 1 which was developed based on the network flow formulation is more efficient than Formulation 2, which was built based on the assignment problem model, in terms of CPU time. An upper bound was also proposed using Formulation 2. The upper bound significantly reduces computations time while maintain acceptable solution quality. For future research, one can develop algorithms to solve this problem more efficiently and on large-scale problems.

References