ARTICLE TEMPLATE

# On the Analysis of Last-Mile Relief Delivery on a Tree Network: Application to the 2015 Nepal Earthquake

Abhinav Khare[a], Rajan Batta[a] and Jee Eun Kang[a]

[a]Department of Industrial & Systems Engineering, University at Buffalo, Buffalo, NY 14260 USA

**Abstract**
The last mile delivery in humanitarian relief supply often happens on a tree or an almost-tree network allowing split deliveries. We present a relief delivery model incorporating a tree network for last mile delivery. We developed a mixed integer programming (MIP) formulation with the goal of minimizing the unsatisfied demand of the population. For better computational performance, we reformulated the MIP exploiting the tree network structure and found that this gave an order of magnitude reduction in computational time. To further improve computational efficiency, we developed a heuristic solution method based on a decomposition scheme applied to the tree network formulation. This led to the Capacitated Vehicle Routing Problem on trees with split deliveries (TCVRP-SD), for which we derived a closed-form solution. This decomposition scheme resulted in a further order of magnitude reduction in computation time. To demonstrate the application of our approach we applied our model to the humanitarian logistics relief operation encountered in the 2015 Nepal earthquake.

**KEYWORDS**
tree graphs, split delivery, last mile delivery, mixed-integer programming

## 1. Introduction and Contribution

According to the World Disasters Report 2016 of the International Federation of Red Cross and Red Crescent Societies, 698,077 people died and damage worth 1,399 billion dollars was caused by natural disasters around the world in the period 2006-15 (31). Supply of relief after a disaster is crucial for saving lives and enabling faster recovery. In practice, most humanitarian logistics operations are conducted in two stages, each over its own transportation network. In the first stage, a primary network (air, road, water) is used for transporting aid from different humanitarian organizations and countries to identified staging areas/depots in the vicinity of the affected region. In the second stage, a secondary network (road) is used for last mile delivery from the depots to the affected people. However, in most disasters, the secondary network present in affected areas is partially or fully compromised (40; 35). For instance, post a disaster like earthquake, a densely connected road network as shown in network Figure 1(a) is converted to a sparse network as shown Figure 1(b) as most of the roads are damaged and can't be used for transportation. Usually emergency repairs are performed on the secondary network to ensure at least minimal connectivity with the depot for relief transportation which results a tree or an almost-tree structure as shown in Figure 1(c). Usually tree networks arise when the cost of road construction is much more than the routing cost (37). In the scenario of post disaster natural disaster, recovery time is equivalent to cost and time of road construction is much more than the relief transportation time. Thus, a minimal connectivity is guaranteed during emergency repairs a tree-like network emerges. Motivated by this observation, we focus our analysis in this paper under the assumption that the secondary network is a tree. We also
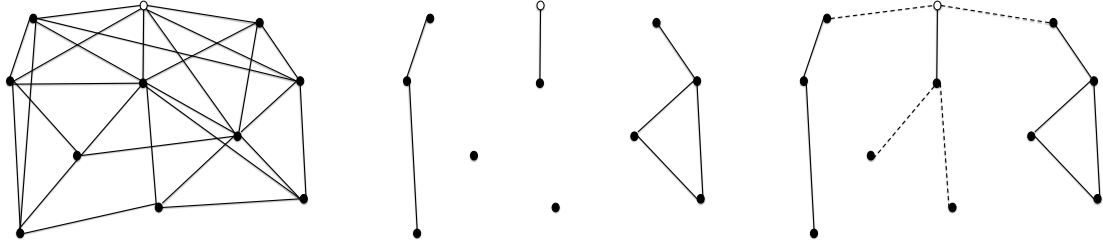
**Figure 1.** (a) A densely connected network (b) the network after a disaster (c) the network to ensure minimum connectivity from the depot

assume that split deliveries are allowed in the our vehicle routing model. In post disaster rescue and relief on the ground split delivery happens and there is existing literature that also assume split delivery (53; 57). This feature is required because in a lot of disaster relief delivery scenario the demand at a node in a period exceeds the capacity of one vehicle and multiple vehicles are utilised to serve a node. We also found that these two assumptions to be true in multiple cases including the relief logistics operations for the Nepal Earthquake in 2015, the second stage network were almost-tree structured mountain trail and split delivery happened using porters and pack animals (56). Assuming the tree structure is not only realistic in cases of network damage and remote mountain transportation but also allows us to develop efficient solution algorithms for relief supply optimization, which we believe have wide applicability in disaster relief.

We develop a mixed integer program (MIP) that captures both stages of post-disaster relief in a single model. It is a multi-network, multi-period relief supply model, and its solution using a standard commercial software like CPLEX is not practical except for toy instances. To address this computational concern, we reformulate the model under the assumption that the second stage network is a tree. Applying CPLEX to the tree formulation results in an order of magnitude improvement in computational efficiency but still leaves us at a point where realistic sized problems cannot be solved. For this reason we develop a heuristic solution method for the tree network case based on decomposition principles. This results in a further order of magnitude reduction in computational effort.

The 2015 Nepal Earthquake inspired our work. In this earthquake, the second stage network consisted of mountain trails, many of which were compromised. The actual trails used for supplies closely resembled a tree network, and transportation over these trails was done by secondary vehicles and pack animals. The first stage network consisted of primary vehicles and staging areas, where the primary vehicles delivered relief supplies as well as the secondary vehicles. Our case study is based on the Nepal scenario. We believe that our model and approach can be applied to a wide range of natural disaster settings.

Our principle technical contribution is the development of an efficient solution method for the case where the second stage network in post-disaster relief has a tree structure and split delivery is allowed. In the decomposition solution approach we introduced the Capacitated Vehicle Routing Problem on trees with split deliveries (TCVRP-SD) and derived a closed-form solution for its optimal objective function value. Using this, we not only develop a computationally efficient solution method but also extend the literature on vehicle routing problem on trees (TVRPS) and vehicle routing model on split delivery (SDVRP). The other contribution is the demonstration of this method via a case study based on an earthquake scenario in the mountainous region of Nepal. It is important to note that our method can also be used to generate heuristic solutions for cases where the second-stage network is an almost tree (i.e. a tree network with some additional arcs and a few cycles). This is why we believe that our findings have wide applicability.

2

## 2. Literature Review

We divide our literature review into three parts. The first part focuses on vehicle routing problems on tree networks. We have used the material in these papers to develop the model with tree route formulation. Next, we discuss the literature on vehicle routing problem (SDVRP) with split deliveries. We extend the work on SDVRP to tree network to solve the decomposition of our model. The third part focuses on last-mile delivery in humanitarian relief operations.

### 2.1. Vehicle Routing on Tree Networks

Research on vehicle routing on tree shaped networks is scant. Labbe et al. published seminal work on the Capacitated Vehicle Routing on Trees in which they defined the Capacitated Vehicle Routing Problem on Trees (TCVRP) (38) which defined the problem, showed it to be NP-hard and proivded the first solution. Basnet et al. (12) developed two new heuristics to solve TCVRP and compared their computational performance to heuristics developed by Labbe et al. Hamaguchi et al. solved a version of TCVRP that seeks a set of tours of the vehicles with minimum total lengths. (30). Mulsea presented five variants of TCVRP and five fast approximation algorithms to solve it (44). Chandran and Raghavan presented a new paradigm in modeling TCVRP's (17) in which they used the tree-like structure in modeling the constraints of the model and developed a bin-packing heuristic to solve their TCVRP. Kumar et al. proposed the Capacitated Vehicle Routing on Trees with Backhaul (TCVRPB) in which line-haul customers require delivery from the depot, and back-haul customers have supply that needs to be delivered to the depot and also developed a 2-approximation algorithm to solve the problem (36). In another work, Kumar et al. proposed a TCVRP with a heterogeneous fleet and developed a heuristic that uses bib packing and Lagrangean based assignment algorithm to solve it (37).

### 2.2. Vehicle Routing Problem with Split Deliveries - SDVRP

Vehicle routing problems have the underlying assumption that each node is visited only by one vehicle. However, vehicle routing problem with split deliveries introduced by Dror and Trodeau in 1989-1990 (24; 25) relaxes this assumption and allows a node to be visited by multiple vehicles. They showed this leads to cost savings over VRPs. The literature on this topic is vast and there are three major surveys on this topic (7; 8; 29) which discuss SDVRP, its proposed solution, its variants and its application in multiple scenarios. Archetti et al. proved that if there is a feasible solution to the SDVRP, then there always exist an optimal integer solution (9). It was also proven vehicle capacity equal to two SDVRP has a polynomial time algorithm to solve and for vehicle capacity greater than equal to three, SDVRP is NP-hard (5). Archetti et al. found the maximum possible savings with SDVRP (6) and also showed how that depended on the size of the problem (7). To solve SDVRP exact approaches have been suggested that include the arc flow formulation, the Dror and Trudeau heuristic (23). Then there have been the cutting plane approach by Belenguer et al. (13) and a two stage algorithm with valid inequalities (32) which provided good lower bounds followed by multiple column generation approaches (33; 3; 42) that improved the lower bounds. Multiple heuristics have also been proposed that include tabu search (9), a memetic algorithm (16), hybrid algorithms (19; 10; 33; 34), scatter search (43), an adaptive memory algorithm (2), a tabu search with building vocabulary approach (1), attribute-based hill climbing approach (21), an iterated local search approach (50), branch-and-cut approach (4), a randomized granular tabu search (14), the novel priori splitting approach (18) and the latest maximum-minimum distance clustering method (41).

### 2.3. Last-mile Delivery in Humanitarian Logistics

The area of humanitarian logistics is rich with models and hence we mention a few important ones. Balcik and Beamon developed a two-phase multi-period model for relief distribution

(11). (27). Berkoune et al. modeled a multi-commodity and multi-depot routing problem for minimizing the total duration of all trips for last-mile delivery (15).Using a genetic algorithm. Tzeng et al. solved a multi-period aid distribution problem using fuzzy multi-objective linear programming (51). Vitoriano et al. also proposed a multi-citeria problem described using a double-flow model and solved it using goal programming (52). Ozdamar and Linet developed a model for last-mile relief delivery using helicopters (47). More recent models include stochastic optimization models (45; 46), multi-criteria model (26) and models utilizing drones for last mile delivery (20; 48) These are a sample of papers from the vast literature on the last mile distribution in humanitarian relief.

In reviewing the humanitarian logistics literature for last mile delivery we found a rich body for modeling the last mile delivery problem. Literature on SDVRPS is also vast and we also found its application in last mile delivery post disasters like earthquake (53; 57). These application show that splitting deliveries saves cost and also minimizes the vehicle use which are sometimes scarce in post disaster scenarios. While we clearly see almost-tree transportation networks due to damage post disasters like in case of Nepal earthquake, we did not find any work that applied the model of vehicle routing on trees to last mile distribution problem. We fill this gap in the literature. We first utilize existing methodology Chandran and Raghavan (17) to develop a reformulation of our model that exploits the underlying tree structure of the network. We further developed a decomposition of the reformulated model. To solve the reformulated we extend the literature on both TCVRPs and SDCVRP by developing a variant, we call the vehicle routing on trees with split deliveries, SD-TCVRP. We also find an exact algorithm to solve it and a closed form solution for the optimal objective. Using this, we reduce the computational time 20 orders of magnitude as compared on toy datasets. Using this method real sized problems are also solved within 1 hour.

The last mile delivery problems are fraught with data uncertainty and typically have many scenarios. In fact, a common way to solve these problems is to perform scenario analysis where solutions for hundreds of scenarios are developed and analyzed. Thus there is a critical need for reducing the computational effort associated with developing the solution for a specific scenario. We show that by exploiting the tree network structure of the last mile delivery network we can reduce the computation time by orders of magnitude.

## 3. A Two-Stage Relief Delivery Model

The following are the salient features of our model:

(1) **Primary Depot:** This depot serves as a staging area for the relief brought in from humanitarian organizations and aiding countries for onward movement into the affected regions. Set of all nodes is denoted by $V = \{0, 1, 2, ...N - 1\}$. The node $\{0\}$ is the primary depot.

(2) **Secondary Depots:** These depots are staging areas identified in the vicinity of the affected region and store relief supplies brought from the primary depot. The secondary depots also have demand and hence they are subset of demand node set $V - \{0\}$. $S$ denotes the secondary depots.

(3) **Primary Network:** The network to carry the relief supply from a primary depot to the secondary depots via primary vehicles. Primary network contains the primary depot $\{0\}$ and the secondary depots $S$.

(4) **Secondary Network:** The network used for last mile delivery from the secondary depots to the demand nodes in the affected region. Node set $V - \{0\}$ are the demand nodes and define the secondary network. It must be noted that the secondary depots is a subset of demand nodes.

(5) **Multi-Period:** The relief supply operations is done in multiple discrete time periods. $T$

159      is the set of all time periods.

160  (6) **Multi-Commodity:** Transport of multiple commodities like water, food, sanitation and
161      medical supplies is possible. $C = \{0, 1, 2\}$ is the set of relief supplies including water, food,
162      shelter. Index $c = 0$ represent a load for 1 kg of water, $c = 1$ represent load for 1 kg of
163      food, $c = 2$ represent load for 1 Kg of shelter supplies.

164  (7) **Split Deliveries:** More one vehicle can serve a demand node or a secondary depot in a
165      period.

166  (8) **Storage:** Storage of relief items (transported from the primary depot) is available at the
167      secondary depots.

## 3.1. Decision Variables

169  $y_{ijl}^t \in \{0, 1\}$ is 1 if primary vehicle $l$ travels from node $i$ to node $j$ in period $t$, and 0 otherwise,

170  $x_{ijk}^t \in \{0, 1\}$ is 1 if secondary vehicle $k$ travels from node $i$ to node $j$ in period $t$, and 0 otherwise,

171  $r_{ik}^t \in \{0, 1\}$ is 1 if secondary vehicle $k$ is present at depot $i$ in period $t$, and 0 otherwise,

172  $z_{ijk}^{tc} \in \{0, 1, 2..\}$ is the load of commodity $c$ taken by secondary vehicles $k$ from node $i$ to node
173  $j$ in period $t$,

174  $v_{ijl}^{tc} \in \{0, 1, 2..\}$ is the load of commodity $c$ taken by primary vehicle l from node $i$ to node $j$ in
175  period $t$,

176  $d_i^{tc} \in \{0, 1, 2..\}$ is the total load of commodity $c$ dropped by secondary vehicles at node $i$ in
177  period $t$,

178  $d_{ik}^{tc} \in \{0, 1, 2..\}$ is the load of commodity $c$ dropped at node $i$ by secondary vehicle $k$ in period $t$,

179  $D_i^{tc} \in \{0, 1, 2..\}$ is the total load of commodity $c$ dropped at node $i$ by primary vehicle s in
180  period $t$,

181  $S_i^{tc} \in \{0, 1, 2..\}$ is the total load of commodity $c$ stored at node $i$ at the end of period $t$,

182      $i \neq j; i, j \in V = \{0, 2 \ldots \ldots N - 1\}$, where $V$ is the set of all $N$ nodes including demand
183  nodes, primary depot 0 and secondary depots,

184      $c \in C = \{0, 1, 2\}$, where $C$ is the set of relief supplies including water, food, shelter. Index
185  $c = 0$ represent a load of 1 kg of water, $c = 1$ represent load of 1 kg of food, $c = 2$ represent load
186  of 1 Kg of shelter supplies.

## 3.2. Parameters

188  $N$, total number of depots
189  $S$, set of all secondary depots
190  $L$, set of primary vehicles available at the primary depot
191  $K$, set of all secondary vehicles available at the secondary depots
192  $T$, set of all time periods
193  $Up$, parameter for upper bound used in equity constraint
194  $Lo$, parameter for lower bound used in equity constraint
195  $a_{ij}$, travel time for secondary vehicles from node $i$ to node $j$
196  $h_{ij}$, travel time for primary vehicles from node $i$ to node $j$
197  $m_i^{tc}$, demand of commodity $c$ at node $i$ in period $t$
198  $q_s$, capacity of a secondary vehicle $k$
199  $q_l$, capacity of a primary vehicle $l$
200  $f_i$, service time for a secondary vehicle at demand nodes $i$
201  $g_i$, service time for a primary vehicle at node $i$
202  $r_k$, maximum route time allowed for secondary vehicle $k$
203  $r_h$, maximum route time allowed for primary vehicle $h$
204  $s_k$, average speed of secondary vehicle $k$
205  $s_h$, average speed of primary vehicle $h$
206  $B^{tc}$, amount of item $c$ available at primary depot for period $t$

### 3.3. General Model

**Maximize:** $\sum_{i\in V}\sum_{t\in T}\sum_{c\in C} d_i^{tc}$ - $\sum_{i\in V}\sum_{j\in V, i\neq j}\sum_{k\in K}\sum_{t\in T}\alpha a_{ij}x_{ijk}^t - \sum_{i\in S}\sum_{j\in S, j\neq i}\sum_{l\in L}\sum_{t\in T}\alpha h_{ij}y_{ijl}^t$

**St:**

**Primary Vehicle Routing Constraints:**

$$\sum_{i\in S\cup\{0\}, i\neq j} y_{ijl}^t = \sum_{i\in S\cup\{0\}, i\neq j} y_{jil}^t \qquad \forall j\in S\cup\{0\}, l\in L, t\in T \tag{1}$$

$$\sum_{j\in S} y_{0jl}^t = 1 \qquad \forall l\in L, t\in T \tag{2}$$

$$\sum_{i\in W}\sum_{j\in W, j\neq i} y_{ijl}^t \leq |W|-1 \qquad \forall W\subseteq S, |W|\geq 2, l\in L, t\in T \tag{3}$$

$$\sum_{j\in S}\sum_{c\in C} v_{0jl}^{tc} \leq q_l \qquad \forall l\in L, t\in T \tag{4}$$

$$\sum_{i\in S\cup\{0\}}\sum_{j\in S, j\neq i} y_{ijl}^t (h_{ij}+g_i) \leq r_h/s_h \qquad \forall l\in L, t\in T \tag{5}$$

**Secondary Vehicles Routing Constraints:**

$$\sum_{i\in V-\{0\}, i\neq j} x_{ijk}^t = \sum_{i\in V-\{0\}, i\neq j} x_{jik}^t \qquad \forall j\in V-\{0\}, k\in K, t\in T \tag{6}$$

$$\sum_{j\in V-\{0\}} x_{ijk}^t = 1 \qquad \forall i\in S, k\in K, t\in T \tag{7}$$

$$\sum_{i\in U}\sum_{j\in U, j\neq i} x_{ijk}^t \leq |U|-1 \qquad \forall U\subseteq (V-S)-\{0\}, |U|\geq 2, k\in K, t\in T \tag{8}$$

$$\sum_{j\in(V-S)-\{0\}}\sum_{c\in C} z_{ijk}^{tc} \leq q_s \qquad \forall i=S, k\in K, t\in T \tag{9}$$

$$\sum_{i\in(V-S)-\{0\}}\sum_{j\in V-\{0\}, j\neq i} x_{ijk}^t (t_{ij}+f_i) \leq r_k/s_k \qquad \forall k\in K, t\in T \tag{10}$$

$$\sum_{i\in V-\{0\}} x_{ijk}^t \leq 1 \qquad j\in S, k\in K, t\in T \tag{11}$$

$$x_{ijk}^t = 0 \qquad \forall i\in S, j\in S\cup\{0\}, k\in K, t\in T \tag{12}$$

**Loading and Delivery Constraints:**

$$\sum_{c\in C} z_{ijk}^{tc} \leq M x_{ijk}^t \qquad \forall i\in V-\{0\}, j\in V-\{0\}, k\in K, t\in T \tag{13}$$

$$\sum_{c\in C} v_{ijl}^{tc} \leq M y_{ijl}^t \qquad \forall i\in S\cup\{0\}, j\in S\cup\{0\}, l\in L, t\in T \tag{14}$$

$$\sum_{j\in V-\{0\}}\sum_{k\in K} z_{ijk}^{tc} - \sum_{j\in V-\{0\}}\sum_{k\in K} z_{jik}^{tc} = d_i^{tc} \qquad \forall i\in(V-S)-\{0\}, t\in T, c\in C \tag{15}$$

$$\sum_{j\in S\cup\{0\}}\sum_{l\in L} v_{ijl}^{tc} - \sum_{j\in S\cup\{0\}}\sum_{l\in L} v_{jil}^{tc} = D_i^{tc} \qquad \forall i\in S, t\in T, c\in C \tag{16}$$

$$D_0^{tc} = 0 \qquad \forall t\in T, c\in C \tag{17}$$

$$d_i^{tc} \leq m_i^{tc} \qquad \forall i\in V-\{0\}, t\in T, c\in C \tag{18}$$

$$\sum_{k\in K} d_{ik}^{tc} \leq d_i^{tc} \qquad \forall i\in V-\{0\}, t\in T, c\in C \tag{19}$$

$$m_i^{tc} - d_i^{tc} \leq Up/N \sum_{c \in C} \sum_{j \in V-\{0\}} (m_j^{tc} - d_j^{tc}) \qquad \forall i \in V - \{0\}, t \in T, c \in C \qquad (20)$$

$$m_i^{tc} - d_i^{tc} \geq Lo/N \sum_{c \in C} \sum_{j \in V-\{0\}} (m_j^{tc} - d_j^{tc}) \qquad \forall i \in V - \{0\}, t \in T, c \in C \qquad (21)$$

$$\sum_{c \in C} z_{ijk}^{tc} = 0 \qquad \forall i \in V - S, j \in S, k \in K, t \in T \qquad (22)$$

$$\sum_{c \in C} v_{i0l}^{tc} = 0 \qquad \forall i \in S, k \in K, t \in T \qquad (23)$$

$$\sum_{c \in C} d_{ik}^{tc} \geq r_{ik}^{t} \qquad \forall i \in V - \{0\}, k \in K, t \in T \qquad (24)$$

$$\sum_{c \in C} d_{ik}^{tc} \leq M r_{ik}^{t} \qquad \forall i \in V - \{0\}, k \in K, t \in T \qquad (25)$$

$$\sum_{i \in S-\{0\}} r_{ik}^{t} = 0 \qquad \forall i \in V, k \in K, t \in T, c \in C \qquad (26)$$

**Storage Constraints:**

$$S_i^{tc} = S_i^{t-1c} + D_i^{tc} - \sum_{j \in V-S, j \neq i} \sum_{k \in K} z_{ijk}^{tc} - d_i^{tc} \qquad \forall i \in S - \{0\}, t \in T - \{0\}, c \in C \qquad (27)$$

$$S_i^{0c} = D_i^{0c} - \sum_{j \in S-\{0\}, j \neq i} \sum_{l \in L} z_{ijk}^{0c} - d_i^{0c} \qquad \forall i \in S - \{0\}, c \in C \qquad (28)$$

$$S_0^{tc} = S_0^{t-1c} - \sum_{j \in S-\{0\}, j \neq i} \sum_{l \in L} v_{0jl}^{tc} + B^{tc} \qquad \forall i \in S - \{0\}, t \in T, c \in C \qquad (29)$$

$$S_0^{0c} = - \sum_{j \in S-\{0\}, j \neq i} \sum_{l \in L} v_{0jl}^{0c} + B^{0c} \qquad \forall c \in C \qquad (30)$$

### 3.4. Model Explanation

**Maximize:** $\sum_{i \in V} \sum_{t \in T} \sum_{c \in C} d_i^{tc}$ - $\sum_{i \in V} \sum_{j \in V, i \neq j} \sum_{k \in K} \sum_{t \in T} \alpha a_{ij} x_{ijk}^{t} - \sum_{i \in S} \sum_{j \in S, j \neq i} \sum_{l \in L} \sum_{t \in T} \alpha h_{ij} y_{ijl}^{t} (*)$

Consider the objective function given by equation $(*)$. The term $m_i^{tc}$ can be removed for optimization purposes as it is a constant. Minimizing unsatisfied demand summed over all commodities, all villages and all periods given by $\sum_{i \in V} \sum_{t \in T} \sum_{c \in C} (m_i^{tc} - d_i^{tc})$ is our primary objective. Minimizing travel cost is our secondary objective. We prioritize our primary objective over the secondary objective by multiplying the secondary objective term by a hyper-parameter $\alpha$. We vary the values of $\alpha$ while testing the solution methods, to find an appropriate range of $\alpha$ for prioritizing the primary objective over secondary objective for a given data set. This is discussed in detail in our case study.

### 3.4.1. Constraints

Constraints $(1) - (5)$ are primary vehicle routing constraints for routing primary vehicles on the primary network on the node set, $S \cup \{0\}$. Constraints $(1) - (2)$ ensure that the route for a primary vehicle is a cycle and that a primary vehicle starts and ends its tour at the primary depot. Constraint $(3)$ is for sub-tour elimination for the primary vehicle. Constraint $(4)$ is the primary vehicle capacity constraint ensuring that the primary vehicle never carries more than its capacity. Constraint $(5)$ ensures that the primary vehicle does not cross its limit of travel in a period. Constraints $(6) - (12)$ are equivalent routing constraints to $(1)-(5)$ for the secondary vehicles on the secondary network on the node set, $V - \{0\}$.

Constraints $(13) - (26)$ govern the load carried and delivery made by primary and secondary vehicles. Constraint (13) ensure that a a load on an arc occurs if a secondary vehicle is traversing it. Constraint (14) do the same for primary vehicles. Constraints $(15) - (16)$ are the relationships between delivery and load variables for secondary vehicles and primary vehicles, respectively. Constraint (17) ensures that there is no delivery at a primary depot. Constraint (18) ensures that delivery of an item to a demand node in a period is less than its demand in that period. Constraint (19) ensures that the total delivery of an item in a demand node for a period is the sum of the items delivered to it by each of the visiting secondary vehicles. Constraints $(20) - (21)$ are equity constraints that ensure that the unsatisfied demand at each village is within a lower and upper bound of the mean unsatisfied demand. Constraints $(22) - (23)$ ensure that the load variables entering the depots are zero, i.e. the vehicles deliver all their load in each period. Constraints $(24) - (25)$ ensure that secondary vehicles deliver their load to the demand nodes that it visits. Constraint (26) ensures that each secondary vehicle serves its assigned secondary depot during each period.

The constraints $(27) - (30)$ are the storage-related constraints. The constraints $(27) - (28)$ establish the relationship between storage variables, delivery variables and load variables at the secondary depots. Constraints $(29) - (30)$ do the same for the primary depot.

We found that using a commercial solver like CPLEX on this formulation does not yield an optimal or near-optimal solution in 24 hours of computation time for realistic data sets like the one we have used in our case study. Exploiting the fact that many secondary networks associated with humanitarian relief efforts have a tree or almost-tree like structure, we now present a formulation of our model for the case where the secondary network is a tree, thereby alleviating the computational difficulty of the problem.

## 4. Model with Tree Formulation

In this section, we model the constraints of routing secondary vehicles on tree networks using the properties of vehicle routes on trees as described by Chandran and Raghavan , illustrating the concept through an example (17). Given a set of nodes serviced by a vehicle in a tree network (2, 3, 4, and 6), the arcs travelled by the vehicle form a tree as shown by the directed arcs in Figure 2(a) and the tour is given by visiting the nodes in a depth first manner as shown in Figure 2(b) (17). We use these properties to re-model the secondary vehicle routing constraints. Consider a node $i$ in this tree network. It has a unique parent $P(i)$ and a set of children $Ch(i)$. The route is constructed for each vehicle by "building upwards" from every node serviced by the vehicle towards the depot as shown in Figure 2(b). This generates a covering sub-tree for each secondary vehicle. Having generated the tree, i.e. the arcs the vehicles take, the route of the vehicles is the nodes visited in the depth first order. Using this, we replace four constraints $(6) - (8), (10) - (12)$ in our original model by three constraints $(31) - (33)$. Constraints $(15), (27), (28)$ are replaced by constraints $(34) - (36)$. Following are the reformulated constraints.

$$x^t_{P(i),ik} \geq x^t_{ijk} \qquad \forall i \in (V - S) - \{0\}, j \in Ch(i), k \in K, t \in T \qquad (31)$$

$$x^t_{P(i),ik} \geq r^t_{ik} \qquad \forall i \in (V - S) - \{0\}, k \in K, t \in T \qquad (32)$$

$$x^t_{ijk} = x^t_{jik} \qquad \forall i \in V - \{0\}, j \in V - \{0\}, k \in K, t \in T \qquad (33)$$

$$\sum_{j \in Ch(i)} \sum_{k \in K} z^{tc}_{ijk} - \sum_{j \in P(i)} \sum_{k \in K} z^{tc}_{jik} = d^{tc}_i \qquad \forall i \in (V - S) - \{0\}, t \in T, c \in C \qquad (34)$$

$$S_i^{tc} = S_i^{t-1c} + D_i^{tc} - \sum_{j \in Ch(i),\, k \in K} z_{ijk}^{tc} - d_i^{tc} \qquad\qquad \forall i \in S, t \in T - \{0\}, c \in C \qquad (35)$$

$$S_i^{tc} = D_i^{tc} - \sum_{j \in Ch(i)} \sum_{l \in L} z_{ijk}^{tc} - d_i^{tc} \qquad\qquad \forall i \in S, t = 0, c \in C \qquad (36)$$
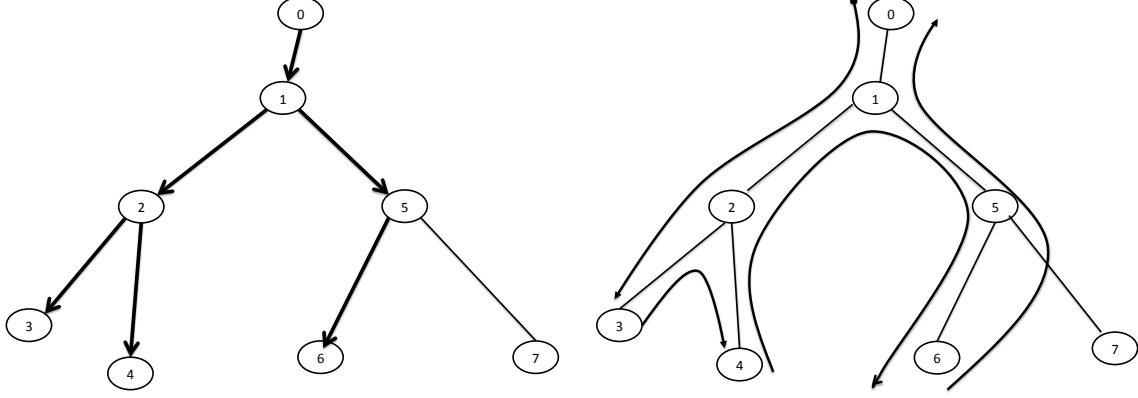


**Figure 2.** (a) Route tree of TCVRP (b) Depth First Order of route

The constraint (31) ensures that if a secondary vehicles $k$ visits node $i$ in period $t$, it travels on arc $(P(i), i)$. The constraint (32) populates the rest of the tree bottom up to the depot, using the fact that if a secondary vehicles uses arc $(ij)$ it will have to use the arc $(P(i), i)$. The constraint (33) makes sure that arcs are travelled twice by a secondary vehicle, once while going down the tree and once while coming up. Constraints (34)-(36) remove summations with the load variables as they are automatically enforced in a tree network.

The tree formulation can be used for situations where the secondary network is an almost-tree by removing arcs from the network to form a tree and by using the resultant tree as an approximation to the secondary network. We tried this approach and found that using a commercial solver like CPLEX on this new model provides an order of magnitude (10 fold) improvement in computational time but still is not suitable for use in realistic sized data sets that typically have to be run several times using different parameter settings. Therefore, we now focus on a decomposition based method which is suitable for realistic data sets.

## 5. A Decomposition Based Solution Method

We decompose the model into three parts: (1) delivery model, (2) capacitated secondary vehicle routing model and, (3) capacitated primary vehicle routing model. The model is separable into aforementioned 3 different parts because decision variable for delivery $(d_i^{tc})$, decision variables for secondary vehicle routing $(x_{ijk}^t, z_{ijk}^{tc}, r_{ik}^t, d_{ik}^{tc})$ and decision variables for primary vehicle routing $(y_{ij,l}^t, v_i ijl^{tc}, S_i^{tc})$ are linked only via very few linking constraints. Constraint (34) is the linking constraint between delivery variables and secondary vehicle routing variables. Constraints $(35) - (36)$ are the linking constraints between delivery variables, secondary routing and the primary variables. In a scenario with small number of linking constraints, it is possible to closely approximate the optimal solution of the model by adding surrogate constraints to the separable models and solving the models individually. The surrogate constraints are added to the separable models to provide the upper bounds to the decision variables variables. In general, a surrogate constraint is an inequality implied by the constraints of an integer program, and is designed to capture useful information that cannot be extracted from the parent constraints individually but is nevertheless is a consequence of their conjunction. In our scenario, the

surrogate constraint in one model provides the upper bounds on the decision variables obtained by conjunction of the linking constraints with constraints of the other two models. These bounds are implicit in the original model and they are added explicitly as surrogate constraints in the decomposed models.
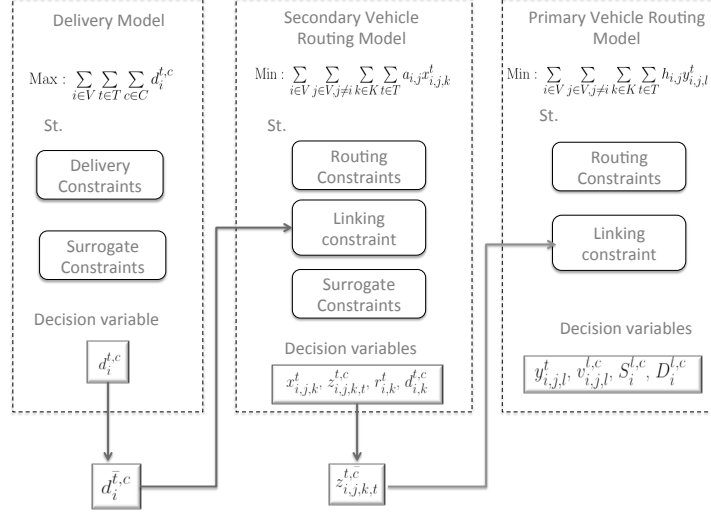


**Figure 3.** Schematic explaining the solution method for the decomposition

We proceed by adding surrogate constraints to the delivery model and secondary vehicle routing models. Constraints $(38) - (40)$ described in Section 5.1 are the surrogate constraints for the delivery model and provide the upper bound on decision variable $d_i^{tc}$ which is implicit in the original model. constraints $(56) - (57)$ described in Section 5.2 are the surrogate constraints for the secondary vehicle routing model and provide the upper bounds to $z_i^{t,c}$. We note that the addition of surrogate constraints increases the solution space and thereby alters the original formulation. In our experimentation with a toy data set described in section 6.2 and 6.3, we found a very small difference in the objective function values due to addition of the surrogate constraints. However, addition of the surrogate constraints facilitates our decomposition.

These models are sequentially solved with the solution of one model becoming the input parameters of the following model. Hence, the optimal solution of the delivery model becomes the input parameters of capacitated secondary vehicle routing model, and so on. The linking constraints $(34)$ is converted to constraint $(50)$ with $\bar{d}_i^{tc}$ as a parameter in the secondary vehicle routing model and constraints $(35) - (36)$ are converted to constraints $(67) - (68)$ in the primary vehicle routing model with $\bar{d}_i^{tc}$ and $\bar{z}_{ijk}^{tc}$ as parameters. Figure 3 explains this process using a schematic.

## 5.1. Delivery Model

The Delivery Model (DM) determines the allocation of commodities to the demand nodes in each period. It uses the decision variables $d_i^{tc}$. The objective is given by $\sum_{i \in V} \sum_{t \in T} \sum_{c \in C} d_i^{tc}$. The model can be written as:

**Maximize:** $\sum_{t \in T} \sum_{i \in V} \sum_{cinC} d_i^{tc}$

10

**st:**

$$d_i^{tc} \le m_i^{tc} \qquad\qquad \forall i \in V - \{0\}, t \in T, c \in C \qquad (37)$$

$$\sum_{i \in V-\{0\}} \sum_{c \in C} d_i^{tc} \le |K| q_s \qquad\qquad \forall t \in T \qquad (38)$$

$$\sum_{i \in V-\{0\}} \sum_{Q \in \{1,2...t\}} \sum_{c \in C} d_i^{Qc} \le |Q||L| q_l \qquad\qquad \forall t \in T \qquad (39)$$

$$\sum_{i \in V-\{0\}} \sum_{Q \in \{1,2...t\}} d_i^{Qc} \le \sum_{Q \in \{1,2...t\}} B^{Qc} \qquad\qquad \forall t \in T, c \in C \qquad (40)$$

$$m_i^{tc} - d_i^{tc} \le Up/N \sum_{c \in C} \sum_{j \in V-\{0\}} (m_j^{tc} - d_j^{tc}) \qquad \forall i \in V - \{0\}, t \in T, c \in C \qquad (41)$$

$$m_i^{tc} - d_i^{tc} \ge Lo/N \sum_{c \in C} \sum_{j \in V-\{0\}} (m_j^{tc} - d_j^{tc}) \qquad \forall i \in V - \{0\}, t \in T, c \in C \qquad (42)$$

Constraint (37) ensures that the amount of a commodity delivered to each node in each period, $d_i^{tc}$, does not exceed its demand. Constraint (38) ensures that the total primary vehicle capacity $|K| q_s$ is not exceeded in each period. Constraint (39) ensures that the cumulative capacity of primary vehicles $|Q||L| q_l$ (cumulative over time period) is not exceeded. Constraint (40) ensures that the primary depot does not violate its delivery capacity by requiring that the cumulative delivery to all nodes is less than or equal to the cumulative availability at the primary depot. Constraints $(41) - (42)$ are the equity constraints used in the original and the tree model.

We note that the delivery problem is an extension of the Multiple Bounded Knapsack Problem, described by Detti (22). We developed a special purpose algorithm, Algorithm 2, for solving the delivery model. The procedure and description is Section 7.1 of the Appendix.

### 5.2. Capacitated Secondary Vehicle Routing Model

The capacitated secondary vehicle routing model finds the minimum cost tours for the secondary vehicles given the output from the delivery model. The decision variables are $x_{ijk}^t$, $z_{ijk}^{tc}$, $r_{ik}^t$, $d_{ik}^{tc}$. The objective is minimizing the cost of transportation for the secondary vehicles. The constraints are the routing and loading constraints for the secondary vehicles from the model with tree formulation. $\bar{d}_i^{tc}$'s are the values of the decision variables $d_i^{tc}$ as output from solution method of the delivery model and is input into the secondary vehicle routing problem as parameter.

**Minimize:** $\sum_{i \in V} \sum_{j \in V, j \ne i} \sum_{k \in K} \sum_{t \in T} a_{ij} x_{ijk}^t$

**st:**

$$x_{P(i),ik}^t \ge x_{ijk}^t \qquad \forall i \in (V-S) - \{0\}, j \in Ch(i), k \in K, t \in T \qquad (43)$$

$$x_{P(i),ik}^t \ge r_{ik}^t \qquad \forall i \in (V-S) - \{0\}, k \in K, t \in T \qquad (44)$$

$$x_{ijk}^t = x_{jik}^t \qquad \forall i \in V-\{0\}, j \in V-\{0\}, k \in K, t \in T \qquad (45)$$

$$\sum_{j \in V-S} \sum_{c \in C} z_{0jk}^{tc} \le q_s \qquad \forall k \in K, t \in T \qquad (46)$$

$$\sum_{i \in (V-S)-\{0\}} \sum_{j \in V-\{0\}, j \ne i} x_{ijk}^t (t_{ij} + f_i) \le r_k / s_k \qquad \forall k \in K, t \in T \qquad (47)$$

$$\sum_{c \in C} z_{ijk}^{tc} \leq M x_{ijk}^{t} \qquad\qquad \forall i \in V - \{0\}, j \in V - \{0\}, k \in K, t \in T \quad (48)$$

$$\sum_{j \in Ch(i)} \sum_{k \in K} z_{ijk}^{tc} - \sum_{j \in P(i)} \sum_{k \in K} z_{jik}^{tc} = \bar{d}_{i}^{tc} \qquad\qquad \forall i \in (V - S) - \{0\}, t \in T, c \in C \quad (49)$$

$$\bar{d}_{i}^{tc} \leq m_{i}^{tc} \qquad\qquad \forall i \in V - \{0\}, t \in T, c \in C \quad (50)$$

$$\sum_{k \in K} d_{ik}^{tc} \leq \bar{d}_{i}^{tc} \qquad\qquad \forall i \in V - \{0\}, t \in T, c \in C \quad (51)$$

$$\sum_{c \in C} z_{ijk}^{tc} = 0 \qquad\qquad \forall i \in (V - S) - \{0\}, j \in S, k \in K, t \in T \quad (52)$$

$$\sum_{c \in C} d_{ik}^{tc} \geq r_{ik}^{t} \qquad\qquad \forall i \in V - \{0\}, k \in K, t \in T \quad (53)$$

$$\sum_{c \in C} d_{ik}^{tc} \leq M r_{ik}^{t} \qquad\qquad \forall i \in V - \{0\}, k \in K, t \in T \quad (54)$$

$$\sum_{i \in S - \{0\}} r_{ik}^{t} = 0 \qquad\qquad \forall i \in V - \{0\}, k \in K, t \in T, c \in C \quad (55)$$

$$\sum_{i \in V - \{0\}} \sum_{Q \in \{1,2...t\}} \sum_{c \in C} z_{ijk}^{Qc} \leq |Q||L|q_{l} \qquad\qquad \forall t \in T \quad (56)$$

$$\sum_{i \in V - \{0\}} \sum_{Q \in \{1,2...t\}} z_{ijk}^{Qc} \leq \sum_{Q \in \{1,2...t\}} B^{Qc} \qquad\qquad \forall t \in T, c \in C \quad (57)$$
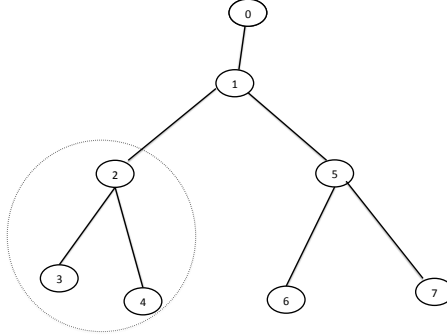


**Figure 4.** Example of a sub-tree of a tree

The secondary vehicle routing problem is a special case of TCVRP. The underlying network is composed of multiple trees with depots at the root nodes and split deliveries allowed. We define a new problem as the capacitated vehicle routing problem on a tree with split deliveries (TCVRP-SD) in which vehicles are routed on a tree with split deliveries allowed. It is a special case of SDVRP in which the underlying network is a tree. Solving the secondary vehicle routing problem is equivalent to solving a TCVRP-SD for each tree in the secondary network for each time period. Theorem 1 provides a closed-form solution for the optimal objective of TCVRP-SD and a unique property of the structure of the routes. We use this to solve the secondary vehicle routing model. For stating Theorem 1, we have to define some notation. Consider a rooted tree $T'$ with $V'$ and $E'$ as the set of nodes and arcs respectively. $P(i)$ is the parent of node $i$, $N(P(i), i)$ is the the number of vehicles that will traverse the arc $(P(i), i)$ each incurring a cost $Cost(P(i), i)$. Consider a sub-tree $S(i)$ rooted at $i$ and $d(S(i))$ is the amount of the commodities delivered to the nodes within $S(i)$. We also define a Parent-Children Pair $(i, Ch(i))$. For instance, in Figure 4, there are 4 Parent-Children pair $(0, \{1\}), (1, \{2, 5\}), (2, \{3, 4\})$ and $(5, \{6, 7\})$. $S(2)$ is the sub-tree rooted at node 2 and the $d(S(2))$ is equivalent to the sum of the deliveries made at node $2, 3$, and $4$. Since split deliveries are allowed in the model, the number of vehicles traveling the arc $(P(2), 2) = (1, 2)$ is equal

to $\lceil d(S(2)/q_s \rceil$ , where $q_s$ is the capacity of each vehicle. It must be noted that minimum costs of an SDVRP are lower than that of VRP on the same network (8). Also, in an SDVRP there is a feasible solution with minimum number of vehicles to complete the deliveries. However the number of vehicles in a minimum cost solution might be more than the minimum number of vehicles possible. In TCVRP-SD, we show that the minimum cost route also utilizes the minimum number of vehicles (8). Now, we are ready to state Theorem 1 and Theorem 2.

**Theorem 1:** Given a TCVRP-SD problem, the optimal objective of minimum cost is given by $2 \sum_{i \in V} \lceil d(S(i))/q_s \rceil Cost(P(i), i)$ for each node $i$ where $d(s(i))$ is a demand of a sub-tree rooted at $i$, $q_s$ is the capacity of a vehicle and $\lceil d(S(i))/q_s \rceil$ is the minimum number of vehicle required to deliver in sub-tree $S(i)$.

**Theorem 2:** In a TCVRP-SD the minimum cost solution also utilizes the minimum number of vehicles.

The implication of Theorem 1 is that the values of the delivery variables $\bar{d}_i^{tc}$ can be used to calculate the number of vehicles traveling each arc $(P(i), i)$, $N(P(i), i)$. This, in turn, determines the optimal objective $2 * \sum_{i \in V} N(P(i), i) * Cost(P(i), i)$. Also, the arcs traveled by vehicles within parent-children pairs is determined by $N(P(i), i)$'s, using which the route of a vehicle can be traced. We developed an algorithm, Algorithm 2 that uses these implications of Theorem 1. The description and procedure of the algorithm is in the Appendix, Section 7.2. In Step 1-3, as given in the pseudo code of the algorithm, for each time period, the value of $N(P(i), i)$ is calculated using the parameters $\bar{d}_i^{tc}$. In Step 5, the optimal objective is calculated. Steps 6-31 are the vehicle routing steps. We determine the arcs travelled by the vehicles within parent children pairs $(i, Ch(i))$ starting from the leaves of the tree going all the way upto the depot (root of the tree). For a leaf node within a parent-children pair, Steps 10-20 are used to determine the values of $x_{ijk}^{\bar{t}}$ and $\bar{d}_i^{tc}$. If the node is not a leaf node Steps 21-31 determine the values of $x_{ijk}^{\bar{t}}$ and $\bar{d}_i^{tc}$.

The implication of Theorem 2 is significant for real world applications like disasters response. The reason we observe split deliveries in real disaster scenarios is justified and warranted by Theorem 2. If the resources like vehicles for transportation and transportation network is sparse and almost-tree due to damage, split delivery of relief is the most cost/resource efficient strategy as it minimizes the vehicles in the minimizing cost solution.

### 5.3. Capacitated Primary Vehicle Routing Model

The capacitated primary vehicle routing models solves the problem of finding the minimum cost tours for the primary vehicles given the optimal deliveries from the delivery model and the optimal tours of the secondary vehicles from the secondary vehicle routing model. The decision variables are $y_{ijl}^t$, $v_{ijl}^{tc}$, $S_i^{tc}$ and $D_i^{tc}$. The objective is minimizing the cost of transportation of the primary vehicles. The constraints are primary vehicle routing constraints, loading constraints and storage constraints from the model with tree formulation. $\bar{d}_i^{tc}$ is the optimal value of $d_i^{tc}$ from the delivery model and $\bar{z}_{ijk}^{tc}$ is the optimal value of $z_{ijk}^{tc}$ from the secondary vehicle routing model. They are input as parameters in the model. We found that capacitated primary vehicle routing models can be solved using CPLEX as computational times are reasonable for realistic data sets. The model is as follows:

**Minimize:** $\sum_{i \in S} \sum_{j \in S, j \neq i} \sum_{l \in L} \sum_{t \in T} h_{ij} y_{ijl}^t$

**st:**

$$\sum_{i\in S\cup\{0\},i\neq j} y^t_{ijl} = \sum_{i\in S\cup\{0\},i\neq j} y^t_{jil} \qquad\qquad \forall j \in S \cup \{0\}, l \in L, t \in T \tag{58}$$

$$\sum_{j\in S} y^t_{0jl} = 1 \qquad\qquad \forall, l \in L, t \in T \tag{59}$$

$$\sum_{i\in W}\sum_{j\in W,j\neq i} y^t_{ijl} \leq |W| - 1 \qquad\qquad \forall W \subseteq S, |W| \geq 2, l \in L, t \in T \tag{60}$$

$$\sum_{j\in S}\sum_{c\in C} v^{tc}_{0jl} \leq q_l \qquad\qquad \forall, l \in L, t \in T \tag{61}$$

$$\sum_{i\in S\cup\{0\}}\sum_{j\in S,j\neq i} y^t_{ijl}(h_{ij} + g_i) \leq r_h/s_h \qquad\qquad \forall l \in L, t \in T \tag{62}$$

$$\sum_{c\in C} v^{tc}_{ijl} \leq M y^t_{ijl} \qquad\qquad \forall i \in S \cup \{0\}, j \in S \cup \{0\}, l \in L, t \in T \tag{63}$$

$$\sum_{c\in C} v^{tc}_{ijl} = 0 \qquad\qquad \forall i \in (V - S) - \{0\}, j \in S, k \in K, t \in T \tag{64}$$

$$S^{tc}_i = S^{t-1c}_i + D^{tc}_i - \sum_{j\in Ch(i),\,k\in K} z^{\bar{t}c}_{ijk} - \bar{d}^{tc}_i \qquad\qquad \forall i \in S, t \in T - \{0\}, c \in C \tag{65}$$

$$S^{tc}_i = D^{tc}_i - \sum_{j\in Ch(i)}\sum_{l\in L} z^{\bar{t}c}_{ijk} - \bar{d}^{tc}_i \qquad\qquad \forall i \in S, t = 0, c \in C \tag{66}$$

## 6. Case Study

In our case study, we modeled the relief supply operations that were conducted by the Logistics Cluster of the United Nations World Food Program for the very remote and mountainous regions of Nepal. They called this operation the Remote Access Operation (RAO) (55). In RAO, first, the relief supplies were transported from the district headquarters to identified base camps/ helicopter landing zones (HLZ) via helicopters. Next, they were transported to the villages at higher altitudes and remote locations via porters and mules (other animal packs) on mountain trails. The helicopter network was the primary network while the porter network was the secondary network. Figure 5(a) and 5(b) show helicopter landing zones (primary network) and the mountain trails (secondary network). As evident, the secondary network of mountain trails has multiple almost-trees with very less cycles. The following section described the data set in detail section includes the case study that we preformed.
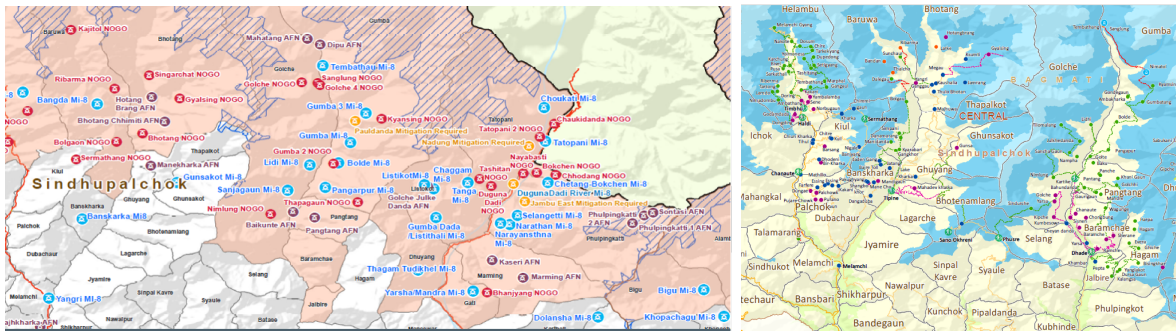


**Figure 5.** (a) Map of Identified Helicopter Landing Zones in Nepal (b) Map of Mountain Trail Network in Sindhupalchowk district, Nepal (39)

The case study was preformed on two data sets: (1) a toy data set created with miniature versions of the Nepal networks for comparing the computational performance of CPLEX and our solution method, and (2) real data from Sindhupachowk district of Nepal (part of RAO, Nepal Earthquake 2015) for testing our solution method's performance on realistic cases (CPLEX was found to be computationally impractical for these cases). CPLEX was used to solve the original and the reformulated model on the toy data set while the decomposition method was tested using both the toy and real data sets. For solving the reformulated model, cycles were removed from the secondary networks using Krushkal's minimum spanning tree algorithm. The following sections contain detailed descriptions of the data sets.

### 6.1. Sindhupalchowk Data Set

The data set contains one hundred sixteen villages as part of the porter network, seven helicopter landing zones/trail heads (in seven of the above villages) and one helicopter depot in the Sindhupalchok district of Nepal. Table 1 is a sample description of our data set. Figure 6 is the map of Sindhupalchok district that contains our data set with the villages, depot at Chautara, and trail heads/HLZ's at Chanaute, Haldi, Timbhu, Sermathang, Tipine, Dhade and Kartike. The cost matrix for the porter network which consists of the time taken by the porters to travel between villages was constructed using the Naismith's rule. The original Naismith's rule from the year 1892 says, that men should allow 1 hour per 3 miles on the map and an additional 1 hour per 2000 feet of ascent. Naismith's 1 h/3 mi + 1 h/2000 ft can be converted to other conventions. We use the convention 12 min/1 km + 10 min/ 100 m.

**Table 1.** Sample of Villages and Municipalities Served in Sindhupalchowk district served in the model

| Node | Village | Municipality | Depot / Trail Head / Village |
|------|---------|--------------|------------------------------|
| 0 | Chautara | Chautara | Depot |
| 1 | Chanaute | Palchok | Trail Head |
| 2 | Mathillo | Palchok | Village |
| 3 | Dungae | Palchok | Village |
| 4 | Farfere | Palchok | Village |
| 5 | Pal chowk | Palchok | Village |
| 6 | Essing | Palchok | Village |
| 7 | Kakani | Palchok | Village |
| 8 | Dhodeni | Kiul | Village |
| 9 | Bir-Kharka | Kiul | Village |
| 10 | Barsang | Kiul | Village |

Table 2 has the population of the municipalities of the villages from the population census of Nepal in 2011. Since we did not have village-level population data, we distributed the population of each municipality into its villages such that the population of the villages followed a normal distribution and summed up to the total population of the municipality. Using this population, we estimated the demand for food and metal roofs in each period at each village. The average number of people in a Nepali family is 4.62 (ReliefWeb2016)). We assumed a requirement of one 26 gauge metal roof per family which weighs approximately 0.45 Kg. The daily food requirement of a person is nearly 600 g providing 2100 Kcal of energy, 65 g and 40g of proteins and fat respectively (in the form of cereals, pulses, oils, salt, sugar etc.) (54). It must be noted that in the case of the RAO, food and shelter related items like tin roofs were given priority. Water was available in the villages via local hand pumps that use the ground water table. There are 1000 porters/animal packs and 8 Mi-8 helicopters as primary and secondary vehicles. The weight carrying capacity of a porter is 30 kg while that of a helicopter is 4 metric ton or 4000 Kg. The limit to travel for each porter is 4 days (96 hours) which includes 4-hour service and rest time at each of the nodes served. For a helicopter, the limit on travel in one period is 8 hours which includes 0.5 hours of service time at each HLZ. The average speed of a helicopter is assumed to be 100 km/hour. The amount of each item available for distribution at the helicopter depot is

**Table 2.** Number of households and population of municipalities of Sindhupalchowk

| Municipalities | Households | Total Population |
|---|---|---|
| Baramchae | 705 | 3,248 |
| Baruwa | 487 | 1,831 |
| Bhanskhara | 588 | 2,259 |
| Bhotang | 624 | 2,582 |
| Bhotenamlag | 792 | 3,551 |
| Chautara | 1,618 | 5,952 |
| Ghunsakot | 449 | 1,902 |
| Golche | 731 | 3,611 |
| Gumba | 674 | 3,431 |
| Hagam | 818 | 3,847 |
| Helmabu | 656 | 2,564 |
| Jalbire | 611 | 2,540 |
| Palchok | 489 | 1,927 |
| Pantang | 487 | 2,481 |

**Table 3.** Number of households and population of municipalities of Sindhupalchowk

| Model | Solution Method | $\alpha$ | Objective | Obj 1 | Obj 2 | Comp Time(secs) |
|---|---|---|---|---|---|---|
| Original | CPLEX | 0.0001 | 479.9006 | 480 | 994 | 3.09 |
| Reformulated | CPLEX | 0.0001 | 479.8923 | 480 | 1077 | 1.24 |
| Reformulated | Decomposition | 0.0001 | 479.8868 | 480 | 1132 | 0.32 |
| Original | CPLEX | 0.01 | 471.02 | 480 | 898 | 3143.14 |
| Reformulated | CPLEX | 0.01 | 469.32 | 480 | 1068 | 102.1 |
| Reformulated | Decomposition | 0.01 | 468.68 | 480 | 1132 | 0.26 |
| Original | CPLEX | 0.1 | 390.2 | 480 | 898 | 4013.05 |
| Reformulated | CPLEX | 0.1 | 373.2 | 480 | 1068 | 136.17 |
| Reformulated | Decomposition | 0.1 | 366.8 | 480 | 1132 | 0.27 |
| Original | CPLEX | 0.5 | 31 | 480 | 898 | 3468.46 |
| Reformulated | CPLEX | 0.5 | -54 | 480 | 1068 | 28.34 |
| Reformulated | Decomposition | 0.5 | -86 | 480 | 1132 | 0.27 |
| Original | CPLEX | 1 | -418 | 471 | 898 | 4452.18 |
| Reformulated | CPLEX | 1 | -525 | 471 | 996 | 26.89 |
| Reformulated | Decomposition | 1 | -661 | 471 | 1132 | 0.13 |
| Original | CPLEX | 1.5 | -866 | 475 | 894 | 3965.33 |
| Reformulated | CPLEX | 1.5 | -1011 | 399 | 940 | 8.22 |
| Reformulated | Decomposition | 1.5 | -1218 | 480 | 1132 | 0.13 |
| Original | CPLEX | 2 | -1294 | 410 | 852 | 202.78 |
| Reformulated | CPLEX | 2 | -1505 | 329 | 917 | 3.97 |
| Reformulated | Decomposition | 2 | -1784 | 480 | 1132 | 0.3 |

10000 kg or 10 metric ton for each period. The operation was carried out for three time periods each equivalent to a period of 4 days. We vary the value of parameter $\alpha$ from 0.001 to 2 and show the difference in results.

## 6.2. Toy Data Set

The toy data set contains nine villages as part of the porter network, two helicopter landing zones, two mountain trails, and one helicopter depot. There are 8 porters/animal packs and two helicopters for transpiration. The distances in the cost matrices and demands in each village is also reduced to smaller numbers. Rest of the parameters are the same as in the Sindhupachowk data set.

## 6.3. Results and Discussion

In the experiment with the toy data set, we used CPLEX to solve the original model, CPLEX to solve the reformulated model, and our decomposition based solution method to solve the reformulated model for different values of $\alpha$. Table 3 compares the results of these three schemes. It

contains the the values of the objective function, the primary objective, Obj $1 = \sum\limits_{i \in V} \sum\limits_{t \in T} \sum\limits_{c \in C} d_i^{tc}$, the secondary objective, Obj $2 = \sum\limits_{i \in V} \sum\limits_{j \in V, i \neq j} \sum\limits_{k \in K} \sum\limits_{t \in T} a_{ij} x_{ijk}^t + \sum\limits_{i \in S} \sum\limits_{j \in S, i \neq j} \sum\limits_{l \in L} \sum\limits_{t \in T} h_{ij} y_{ijl}^t$ and the computational time for the three schemes. We now discuss the results.

We varied $\alpha$ from 0.001 to 2. The primary objective, Obj 1, is at its peak value 480 for $\alpha$ ranging from 0.0001 to 1 (in both the original and reformulated model) and decreases if $\alpha$ is further increased for both the models. The secondary objective achieves its minimum value (for both the models) at $\alpha$ ranging from 0.01 to 2. For overall optimal objective the primary objective must achieve it's maximum while the secondary objective should be at its minimum value for the given maximum value of primary objective. Hence, we concluded that for the toy data set the desirable range of $\alpha$ to achieve the maximum primary objective and the minimum secondary objective is $\alpha \in (0.1, 0.5)$. However, this is true for the the given dataset. The best range of $\alpha$ for which the primary objective is maximized depends on the dataset and the parameters. We show that in the Section 7.

An interesting observation is that the decomposition based solution method always maximizes the priority objective to 480 and is unaffected by the value of $\alpha$. This is because the solution method breaks down the model into delivery and routing models. It always solves the delivery model prior to the routing model, maximizing the primary objective before minimizing the secondary objective.

We observe that the value of the optimal objective for both the original and the re-formulated model are nearly equal for $\alpha \in (0.0001, 0.5)$. This can be explained by the fact that the secondary network in the original model is a 3-almost tree graph with very small number of cycles. Hence, in the reformulated model, when we apply the Krushkal's Minimum Spanning tree to the original network (to obtain a a tree), at most 3 arcs are eliminated from the bi-connected components of the graphs (that contain cycles). Consequently, the minimum spanning tree of the network very closely approximates the original network. Hence, the change in cost of routing secondary vehicles as reflected in the secondary objective is very small. Since the secondary objective contributes in the overall objective by the factor of $\alpha$, the change in the overall objective function is negligible. It can also be observed that the value of the objective from the decomposition based solution method approximates the value for both the original and the reformulated model very well. As expected, the value of primary objective for both the models solved using CPLEX is equal (for all $\alpha$'s). This is because the delivery variables are unaffected by reformulation of constraints and the structure of the underlying network which is a tree in the reformulated model.

It is observed that for most values of $\alpha$ on the toy data set, the computational time for CPLEX on the original model is the highest. We observe that (for $\alpha$ ranging from 0.1 to 1) there is a 10-fold reduction in compuational times when CPLEX is used to solve the reformulated model instead of the original model. The decomposition based solution method shows exceptional computational efficiency as compared to CPLEX, with a 100 fold reduction in computational time, solving each case in the toy data set in less than a second. However, it must be note that the behavior of computational performance under the three schemes would change with increase in data set size. The difference in the efficiency may also change.

We observed that the usage of CPLEX became computationally burdensome as we increased the toy data set size by adding 10 more demands nodes. Hence, for solving the real world data set from Sindhupalchow district we used the decomposition method. For the Sindhupachowk dataset we fixed $\alpha$ at 0.1. The method solved the model in a little more than one hour of computational effort. It yielded a primary objective equal to 30000, secondary objective equal to 15312 and an overall objective equal to 22344. To test the sensitivity of the

**Table 4.** Variation in values of objectives with variation in parameters

| Parameter | Original Value | Changed Value | Obj 1 | Obj 2 | Objective | Comp time (secs) |
|---|---|---|---|---|---|---|
| None | N/A | N/A | 90000 | 15312 | 28468.8 | 3376.12 |
| $q_s$ | 30 | 20 | 60000 | 13326 | 18667.4 | 3469.71 |
| $q_l$ | 4000 | 2000 | 48000 | 14971 | 14502.9 | 3793.33 |
| $L$ | 8 | 4 | 48000 | 14203 | 14579.7 | 3642.56 |
| $K$ | 1000 | 500 | 45000 | 13117 | 13688.3 | 3725.13 |
| $B^{tc}$ | 50000 | 9000 | 81000 | 14159 | 25584.1 | 871.45 |
| $s_h$ | 100 | 50 | 90000 | 22583 | 18708.5 | 27741.7 |

model to different parameters, we varied the parameters and report results in Table 4. The primary objective is bounded by total load carrying capacity, which is equal to $|T||K|q_s$, $|T||L|q_l$ and $\sum_{t \in T} \sum_{c \in C} B^{tc}$. Therefore, changes in the parameters $q_s$, $q_l$, $|L|$ $|K|$ and $B^{tc}$ are reflected in the the objectives. For instance, on changing the load carrying capacity of a porter, $q_s$, from 30 Kg to 20 Kg, the primary objective also reduced from 9000 to 6000. Similar observations can be made for other parameters.

The primary objective worsened as we reduced the values of the parameters while the secondary objective improved. However, the secondary objective is less sensitive to change in parameters than primary objective (magnitude of change is less). A direct implication of these results is that in a scenario like the Nepal earthquake, parameters like number of porters $|K|$, number of helicopters $|L|$ and amount of total supply at the depot $\sum_{t \in T} \sum_{c \in C} B^{tc}$ that can be controlled should be increased without worrying too much about the cost of transportation. Not only is the cost a secondary objective but also it is less sensitive to parameter values.

The major observation from the case study are:

- **Range of $\alpha$:** Range of alpha for achieving optimal objective in the toy data set is in $\alpha \in (0.1, 0.5)$. The value of $\alpha$ is redundant in the decomposition based method as it always optimizes objective 1 before objective 2.
- **Computational Time:** There is 10 fold reduction in computational time using CPLEX to solve the reformulated as compared to the original model (on the toy dataset). There is a further 100 fold reduction in computational time using the decomposition model.
- **Sensitivity Analysis:** The secondary objective of cost of transportation is less sensitive to parameters like number of parameters like number of porters $|K|$, number of helicopters $|L|$ and amount of total supply at the depot $\sum_{t \in T} \sum_{c \in C} B^{tc}$ than expected. Therefore these parameters can be increased without worrying about costs.

In conclusion, we observed that the last mile delivery in disaster relief transportation often happens on almost-tree and utilises split deliveries. Motivated by this, we developed a two stage multi-period vehicle routing model with split deliveries, in which the network for last mile deliveries is a tree or an almost-tree. We developed a reformulation of the model which uses the properties of routing on trees to improve computational efficiency by an order of magnitude. We also developed a decomposition of the model that further improves computational performance by another order of magnitude. To solve the routing problem on a tree network in the decomposed model, we defined the Capacitated Vehicle Routing Problem on a Tree with Split Deliveries (TCVRP-SD) and developed an exact algorithm for its solution. We proved certain mathematical properties of the TCVRP-SD and found a closed-form solution to the optimal objective function value of TCVRP-SD. Our theoretical findings show that using split delivery in a sparse and almost-tree damaged network is the most cost/resource efficient strategy as the minimum cost solution also minimises the number of vehicles.

To validate the model and our solution method, we presented a case study using a data set from relief delivery operations in the 2015 Nepal earthquake. We showed that our method

improved the computational efficiency over tradition methods like CPLEX by 20 folds on small datasets. CPLEX became intractable beyond 10-12 nodes. On real-sized data of the case study our method could solve the problem within one hour. Using sensitivity analysis we found that the secondary objective of minimizing cost was more sensitive to the parameters of number of secondary vehicles, the number of primary vehicles and the amount supplied than the primary objective of minimizing unmet demand.

## References

Rafael E Aleman and Raymond R Hill. A tabu search with vocabulary building approach for the vehicle routing problem with split demands. *International Journal of Metaheuristics*, 1 (1):55–80, 2010.

Rafael E Aleman, Xinhui Zhang, and Raymond R Hill. An adaptive memory algorithm for the split delivery vehicle routing problem. *Journal of Heuristics*, 16(3):441–473, 2010.

Claudia Archetti, Nicola Bianchessi, and Maria Grazia Speranza. A column generation approach for the split delivery vehicle routing problem. *Networks*, 58(4):241–254, 2011.

Claudia Archetti, Nicola Bianchessi, and M Grazia Speranza. Branch-and-cut algorithms for the split delivery vehicle routing problem. *European Journal of Operational Research*, 238 (3):685–698, 2014.

Claudia Archetti, Renata Mansini, and Maria Grazia Speranza. Complexity and reducibility of the skip delivery problem. *Transportation Science*, 39(2):182–187, 2005.

Claudia Archetti, Martin WP Savelsbergh, and M Grazia Speranza. Worst-case analysis for split delivery vehicle routing problems. *Transportation science*, 40(2):226–234, 2006.

Claudia Archetti and Maria Grazia Speranza. The split delivery vehicle routing problem: a survey. In *The vehicle routing problem: Latest advances and new challenges*, pages 103–122. Springer, 2008.

Archetti, C., Speranza, M. G., 2012. Vehicle routing problems with split deliveries. International transactions in operational research 19 (1-2), 3–22.

Archetti, C., Speranza, M. G., Hertz, A., 2006. A tabu search algorithm for the split delivery vehicle routing problem. Transportation science 40 (1), 64–73.

Archetti, C., Speranza, M. G., Savelsbergh, M. W., 2008. An optimization-based heuristic for the split delivery vehicle routing problem. Transportation Science 42 (1), 22–31.

Balcik, B., Beamon, B. M., 2008. Facility location in humanitarian relief. International Journal of Logistics 11 (2), 101–121.

Basnet, C., Foulds, L., Wilson, J., 1999. Heuristics for vehicle routing on tree-like networks. Journal of the Operational Research Society 50 (6), 627–635.

Belenguer, J.-M., Martinez, M., Mota, E., 2000. A lower bound for the split delivery vehicle routing problem. Operations research 48 (5), 801–810.

Berbotto, L., García, S., Nogales, F. J., 2014. A randomized granular tabu search heuristic for the split delivery vehicle routing problem. Annals of Operations Research 222 (1), 153–173.

Berkoune, D., Renaud, J., Rekik, M., Ruiz, A., 2012. Transportation in disaster response operations. Socio-Economic Planning Sciences 46 (1), 23–32.

Boudia, M., Prins, C., Reghioui, M., 2007. An effective memetic algorithm with population management for the split delivery vehicle routing problem. In: International Workshop on Hybrid Metaheuristics. Springer, pp. 16–30.

Chandran, B., Raghavan, S., 2008. Modeling and solving the capacitated vehicle routing problem on trees. In: The Vehicle Routing problem: Latest advances and new challenges. Springer, pp. 239–261.

Chen, P., Golden, B., Wang, X., Wasil, E., 2017. A novel approach to solve the split delivery vehicle routing problem. International Transactions in Operational Research 24 (1-2), 27–41.

Chen, S., Golden, B., Wasil, E., 2007. The split delivery vehicle routing problem: Applications, algorithms, test problems, and computational results. Networks: An International Journal

49 (4), 318–329.

Chowdhury, S., Emelogu, A., Marufuzzaman, M., Nurre, S. G., Bian, L., 2017. Drones for disaster response and relief operations: A continuous approximation model. International Journal of Production Economics 188, 167–184.

Derigs, U., Li, B., Vogel, U., 2010. Local search-based metaheuristics for the split delivery vehicle routing problem. Journal of the Operational Research Society 61 (9), 1356–1364.

Detti, P., 2009. A polynomial algorithm for the multiple knapsack problem with divisible item sizes. Information Processing Letters 109 (11), 582–584.

Dror, M., Laporte, G., Trudeau, P., 1994. Vehicle routing with split deliveries. Discrete Applied Mathematics 50 (3), 239–254.

Dror, M., Trudeau, P., 1989. Savings by split delivery routing. Transportation Science 23 (2), 141–145.

Dror, M., Trudeau, P., 1990. Split delivery routing. Naval Research Logistics (NRL) 37 (3), 383–402.

Ferrer, J. M., Martín-Campo, F. J., Ortuño, M. T., Pedraza-Martínez, A. J., Tirado, G., Vitoriano, B., 2018. Multi-criteria optimization for last mile distribution of disaster relief aid: Test cases and applications. European Journal of Operational Research 269 (2), 501–515.

Gibbons, D. E., Samaddar, S., 2009. Designing referral network structures and decision rules to streamline provision of urgent health and human services. Decision Sciences 40 (2), 351–371.

Glover, F., 1968. Surrogate constraints. Operations Research 16 (4), 741–749.

Gulczynski, D. J., Golden, B., Wasil, E., 2008. Recent developments in modeling and solving the split delivery vehicle routing problem. In: State-of-the-Art Decision-Making Tools in the Information-Intensive Age. INFORMS, pp. 170–180.

Hamaguchi, S., Katoh, N., 1998. A capacitated vehicle routing problem on a tree. In: International Symposium on Algorithms and Computation. Springer, pp. 399–407.

IRCS, 2016. World disaster report 2016.
https://www.ifrc.org/Global/Documents/Secretariat/201610/WDR%202016-FINAL_web.pdf

Jin, M., Liu, K., Bowden, R. O., 2007. A two-stage algorithm with valid inequalities for the split delivery vehicle routing problem. International Journal of Production Economics 105 (1), 228–242.

Jin, M., Liu, K., Eksioglu, B., 2008. A column generation approach for the split delivery vehicle routing problem. Operations Research Letters 36 (2), 265–270.

Khmelev, A., Kochetov, Y., 2015. A hybrid local search for the split delivery vehicle routing problem. International Journal of Artificial Intelligence 13 (1), 147–164.

Kovács, G., Spens, K. M., 2007. Humanitarian logistics in disaster relief operations. International Journal of Physical Distribution & Logistics Management 37 (2), 99–114.

Kumar, R., Unnikrishnan, A., Waller, S. T., 2011. Capacitated-vehicle routing problem with backhauls on trees: Model, properties, formulation, and algorithm. Transportation Research Record 2263 (1), 92–102.

Kumar, R., Unnikrishnan, A., Waller, S. T., 2012. Heuristic scheme for heterogeneous vehicle routing problem on trees based on generalized assignment and bin-packing upper bounds. Transportation Research Record 2283 (1), 1–11.

Labbé, M., Laporte, G., Mercure, H., 1991. Capacitated vehicle routing on trees. Operations Research 39 (4), 616–622.

LogisticCluster, 2016. Nepal documents.
https://logcluster.org/countries/NPL

Luis, E., Dolinskaya, I. S., Smilowitz, K. R., 2012. Disaster relief routing: Integrating research and practice. Socio-economic planning sciences 46 (1), 88–97.

Min, J., Jin, C., Lu, L., 2019. Maximum-minimum distance clustering method for split-delivery vehicle-routing problem: Case studies and performance comparisons. Advances in Production Engineering & Management 14 (1), 125–135.

Moreno, L., De AragãO, M. P., Uchoa, E., 2010. Improved lower bounds for the split delivery vehicle routing problem. Operations Research Letters 38 (4), 302–306.

Mota, E., Campos, V., Corberán, Á., 2007. A new metaheuristic for the vehicle routing problem with split demands. In: European Conference on Evolutionary Computation in Combinatorial Optimization. Springer, pp. 121–129.

Muslea, I., 1997. The very offline k-vehicle routing problem in trees. In: Computer Science Society, 1997. Proceedings., XVII International Conference of the Chilean. IEEE, pp. 155–163.

Noyan, N., Balcik, B., Atakan, S., 2015. A stochastic optimization model for designing last mile relief networks. Transportation Science 50 (3), 1092–1113.

Noyan, N., Kahvecioğlu, G., 2018. Stochastic last mile relief network design with resource reallocation. OR Spectrum 40 (1), 187–231.

Ozdamar, L., 2011. Planning helicopter logistics in disaster relief. OR spectrum 33 (3), 655–672.

Rabta, B., Wankmüller, C., Reiner, G., 2018. A drone fleet model for last-mile distribution in disaster relief operations. International Journal of Disaster Risk Reduction 28, 107–112.

ReliefWeb, 2016. Nepal Household Survey.
https://reliefweb.int/report/nepal/nepal-annual-household-survey-201516

Silva, M. M., Subramanian, A., Ochi, L. S., 2015. An iterated local search heuristic for the split delivery vehicle routing problem. Computers & Operations Research 53, 234–249.

Tzeng, G.-H., Cheng, H.-J., Huang, T. D., 2007. Multi-objective optimal planning for designing relief delivery systems. Transportation Research Part E: Logistics and Transportation Review 43 (6), 673–686.

Vitoriano, B., Ortuño, M. T., Tirado, G., Montero, J., 2011. A multi-criteria optimization model for humanitarian aid distribution. Journal of Global Optimization 51 (2), 189–208.

Haijun Wang, Lijing Du, and Shihua Ma. Multi-objective open location-routing model with split delivery for optimized relief distribution in post-earthquake. *Transportation Research Part E: Logistics and Transportation Review*, 69:160–179, 2014.

WHO, UNICEF, 2004. Food and nutrition needs in emergencies.

WFP, 2016. Nepal lesson learned, logistics cluster world, food program.
https://logcluster.org/sites/default/files/logistics_cluster_nepal_lessons_learned_report_160121.pdf

WFP-2, 2016. Remote acess operations.
https://logcluster.org/document/remote-access-operation-snapshot

Wei Yi and Linet Özdamar. A dynamic logistics coordination model for evacuation and support in disaster response activities. *European Journal of Operational Research*, 179(3):1177–1193, 2007.

# 7. Appendix

## 7.1. Algorithm 1

We need to define some notation for describing the algorithm. For each period $t$, there is a maximum delivery $P_t$ given by $\sum_{i \in V} \sum_{c \in C} d_i^{tc}$. In each period $t$, there is an unsatisfied demand at each demand node $(m_i^{tc} - d_i^{tc})$. Let the mean of this unsatisfied demand for all demand nodes in each period be denoted by $M$. Let the upper bound and lower bound on $M$ (ensured by equity constraints $(40) - (41)$ be $U'$ and $L'$ respectively. The algorithm returns $\bar{d}_i^{tc}$, which is input into the capacitated secondary vehicle routing problem as a parameter.

In step 1, as given in the pseudo code of the algorithm, $P_t = \sum_{i \in V} \sum_{c \in C} d_i^{tc}$ is calculated such that the bounds on $P$ due to constraints $(38) - (41)$ are satisfied. In steps 1-9, the algorithm distributes the calculated $P$ into individual $d_i^{tc}$'s, such that the constraint $(40)$ is satisfied and constraint $(41)$ is tight. In Steps 11-16, the remaining delivery $R$ is distributed to the nodes such that the value of $N(P(i), i) = \left\lceil \sum_{c} \bar{d}_i^{tc}/q_s \right\rceil$ (described in the next section) does not change.

21

If remaining delivery $R$ is still positive, in steps 18-20, $R$ is distributed to the nodes in ascending order of their distance from the depot (ensuring the upper and lower bound on unsatisfied demand imposed by constraints $(40) - (41)$ at each node). In Step 21, the objective is calculated.

---

**Algorithm 1** Algorithm to solve the delivery model

---

1: **for** $t \in T$ **do**                                             ▷ for each of the time period

2:      $P_t \leftarrow Min(|K|q_s, |L|q_l, \sum\limits_{i \in V - \{0\}} \sum\limits_{c \in C} m_i^{tc}, \sum\limits_{Q \in \{1,2...t\}} \sum\limits_{c \in C} B^{Qc} - \sum\limits_{Q \in \{1,2...t-1\}} \sum\limits_{c \in C} B^{Qc})$

3:      $M \leftarrow (1/(N-1))(\sum\limits_{i \in V - \{0\}} \sum\limits_{t \in T} \sum\limits_{c \in C} m_i^{tc}) - P_t$

4:      $U' \leftarrow U_p * M$

5:      $L' \leftarrow L_o * M$

6:      **if** $]P_t[\geq L' \, or \, [P_t] \leq U'$ **then**

7:          **return** *Model Infeasible*

8:      **else**

9:          $\bar{d}_i^{tc} \leftarrow m_i^{tc} - ]U'[$

10:     $R \leftarrow P_t - \sum\limits_{i \in V} \sum\limits_{c \in C} \bar{d}_i^{tc}$

11:      i = 1

12:      **while** $R \neq 0$ **and** $i \in V - \{0\}$ **do**

13:          **if** $(]\sum\limits_c \bar{d}_i^{tc} + 1[)/q_s = ]\sum\limits_c \bar{d}_i^{tc}[/q_s$ **then**

14:            $\bar{d}_i^{tc} = \bar{d}_i^{tc} + 1$

15:            $R = R - 1$

16:            $i = i + 1$

17:      **while** $R \neq 0$ **do**

18:          **for** $i \in V_a$ **do**

19:            $\bar{d}_i^{tc} = \bar{d}_i^{tc} + 1$

20:            $R = R - 1$

21: **Objective** $= \sum\limits_{t \in T} \sum\limits_{i \in V} \sum\limits_{c \in C} d_i^{tc} = \sum\limits_{t \in T} P_t$

---

### 7.2. Algorithm 2

In Step 1-3 of the algorithm, as given in the pseudo code of the algorithm, for each time period, the value of $N(P(i), i)$ is calculated using the parameters $\bar{d}_i^{tc}$. In Step 5, the optimal objective is calculated. Steps 6-31 are the vehicle routing steps. We determine the arcs travelled by the vehicles within parent children pairs $(i, Ch(i))$ starting from the leaves of the tree going all the way upto the depot (root of the tree). For a leaf node within a parent-children pair, Steps 10-20 are used to determine the values of $x_{ijk}^{\bar{t}}$ and $\bar{d}_i^{tc}$. If the node is not a leaf node Steps 21-31 determine the values of $x_{ijk}^{\bar{t}}$ and $\bar{d}_i^{tc}$.

**Algorithm 2** Algorithm to solve the Capacitated Secondary Vehicle Routing Problem

| | |
|---|---|
| 1: | **for** each time period $t \in T$ **do** |
| 2: |     **for** each node $i \in V - \{0\}$ **do** |
| 3: |         $N(P(i),i) = ]d(S(i))/q_s[$                                         ▷ Using $\bar{d}_i^{tc}$'s |
| 4: |         **Initialize** $V_i = \{0\}$                      ▷ Empty set for vehicles at each node $j$ |
| 5: |         **Objective** $= 2 * \sum_{i \in V} N(P(i),i) * Cost(P(i),i)$ |
| 6: |     **for** each tree $T'$ in the secondary network **do**          ▷ vehicle routing steps begin here |
| 7: |         **for** each parent-children pair $(i, Ch(i))$ **do**      ▷ from leaf nodes going up to the depot |
| 8: |             **for** each child $j \in Ch(i)$ **do**          ▷ each child in the parent-children pair |
| 9: |                 **if** $j$ is a leaf node **then** |
| 10: |                     **Append** $N(P(j),j)$ vehicles to the set $V_j$ from vehicle set $K$ |
| 11: |                     **for** each vehicle $k$ at the node $j$ **do** |
| 12: |                         $q_k = q_s$ |
| 13: |                         **for** each commodity $c$ required at the node $j$ **do** |
| 14: |                             $d_j^c = \bar{d}_j^{tc}$ |
| 15: |                             $d_{j,k}^{tc} = min(d_j^c, q_k)$ |
| 16: |                             $q_k = q_k - d_{j,k}^{tc}$ |
| 17: |                             $d_j^c = d_j^c - d_{j,k}^{tc}$ |
| 18: |                     **Append** vehicle $k$ to vehicle set $V_P(j)$ of node $P(j)$ |
| 19: |                     $x_{P(j),j,k}^t = 1$              ▷ arc traveled once while going down |
| 20: |                     $x_{j,P(j),k}^t = 1$              ▷ arc traveled once while going up |
| 21: |                 **else if** $j$ is not a leaf node **then** |
| 22: |                     $k_d = N(P(j),j) - |V_j|$ |
| 23: |                     **if** $k_d > 0$ **then** |
| 24: |                         **Transfer** load of vehicles in set $V_j$ to $N(P(j),j)$ |
| 25: |                         **Remove** extra vehicles from set $V_j$ and **append** to the set $K$ |
| 26: |                         **Repeat** Steps 10-20 |
| 27: |                     **else if** $k_d = 0$ **then** |
| 28: |                         **Repeat** Steps 10-20 |
| 29: |                     **else if** $k_d < 0$ **then** |
| 30: |                         **Remove** vehicles from set $K$ and **append** to the set $V_j$ |
| 31: |                         **Repeat** Steps 10-20 |
| 32: |     —**Calculate** $z_{ijk}^{tc}$ using the values of $\bar{d}_{ik}^{tc}$'s |

**Theorem 1:** Given a TCVRP-SD problem, the optimal objective of minimum cost is given by $2 \sum_{i \in V} \lceil d(S(i))/q_s \rceil Cost(P(i),i)$ for each node $i$ where $d(s(i)$ is a demand of a sub-tree rooted at $i$, $q_s$ is the capacity of a vehicle and $\lceil d(S(i))/q_s \rceil$ is the minimum number of vehicle required to deliver in sub-tree $S(i)$.


### 7.3. Proof of Theorem 1

**Proof:** Consider a node i in the tree. Let $N(P(i),i)$ be the number of vehicles that will travel the arc $(P(i),i)$ in a minimum cost solution. Given, the cost of traveling each arc and the fact that each arc is travelled twice, once while going down from the depot and once while coming up towards the depot, the minimum cost is equivalent to the solution is given by $2 \sum_{i \in V} N(P(i),i) * Cost(P(i),i)$. Let $\Delta i$ be the set of nodes in the sub-tree $S(i)$. Then the total amount of delivery made in the sub-tree is given by $d(S(i)) = \sum_{r \in \Delta i} d_r$. Since split deliveries are permissible, the minimum number of vehicles that are required to meet the demands of $S(i)$ is $\lceil d(s(i))/q_s \rceil$ (least integer greater than $d(s(2))/q_s$). Since the costs are to be minimized and the total costs are given by $2 * \sum_{i \in V} N(P(i),i) * Cost(P(i),i)$ , minimum number of vehicles that are required to serve $S(i)$ will travel the arc, $(P(i),i)$. This implies the $N(P(i),i) = \lceil d(s(i))/q_s \rceil$. For a leaf node $i$, $d(s(i)) = d_i$d . Therefore the minimum cost is $2 \sum_{i \in V} \lceil d(S(i))/q_s \rceil Cost(P(i),i)$.

This completes the proof.


## 7.4. Proof of Theorem 2

**Theorem 2:** In a TCVRP-SD the minimum cost solution also utilizes the minimum number of vehicles.

     **Proof:** Using Theorem 1, the number of vehicles traveling an arc $(P(i), i)$ for a minimum cost solution of SDVRP is the minimum number of vehicles required to deliver in sub-tree $S(i)$ and is given by $N(P(i), i) = \lceil d(s(i))/q_s \rceil$. Therefore, the number of vehicles traveling from the depot to sub-trees rooted at the depot is also the minimum vehicles required to serve the sub-trees. Hence, the total number vehicles in minimum cost solution is also the minimum number of vehicles required to do a split delivery.


## 7.5. Range of values for $\alpha$ for our model :

Given our objective function $\sum\limits_{i \in V} \sum\limits_{c} d_i^{tc}$ - $\sum\limits_{i=1}^{N} \sum\limits_{j=1}^{N} \sum\limits_{k=1, i \neq j}^{K*H} \alpha a_{ij} x_{ijk}^t - \sum\limits_{i \ S} \sum\limits_{j S, \ j \neq i} \sum\limits_{l=1}^{L} \alpha h_{ij} y_{ijl}^t$ there exists a range of value of parameter $\alpha$ for which the the optimum value of objective is obtained when our priority objective term $\sum\limits_{i \in V} \sum\limits_{c} d_i^{tc}$ reaches its optimum value.

     **Proof:** The objective function can be written in the form $max(A - \alpha.B)$, where:

$A = \sum\limits_{i \in V} \sum\limits_{c} d_i^{tc}$

$B = \sum\limits_{i=1}^{N} \sum\limits_{j=1}^{N} \sum\limits_{k=1, i \neq j}^{K*H} a_{ij} x_{ijk}^t - \sum\limits_{i \ S} \sum\limits_{j S, \ j \neq i} \sum\limits_{l=1}^{L} h_{ij} y_{ijl}^t$

$\alpha \in (-\infty, \infty)$

     Let Set of all feasible solution be $S_f$. Then, $S_f \in ((A_1, B_{A_i}), (A_2, B_{A_2}) .... (A_{max}, B_{A_{max}} ..., (A_n, B_{A_n}))$ where $n$ is the number of feasible solutions $A_i$ is the value of $A$ for the $i^{th}$ feasible solution $B_{A_i}$ is the set of values of $B$ when $A = A_i$ $A_{max}$ is the max value of $A$ in the feasible solution set $B_{A_{max}}$ is the set values of $B$ when $A = A_{max}$

     Let there be a value of $\alpha$ for which $max(A - \alpha.B) = A_{max} - \alpha max(B_{A_{max}})$. For this $\alpha$,
$A_{max} - \alpha max(B_{A_{max}}) > A_j - \alpha max(B_{A_j})$ and
$A_j - \alpha max(B_{A_j}) > A_i - \alpha(B_{A_i}) \quad i \in (1, 2, ... n) - j, max$

Thus we have:

$\alpha < \frac{A_{max} - A_j}{max(B_{A_{max}}) - max(B_{A_j})}$