Integer Programming Formulations for Vehicle Routing Problems

Author | Ishank Juneja

May 25, 2018

1 Introduction

VRPs are problems concerning the distribution of goods from depots to final users (customers) Solving a VRP is equivalent to determining a set of routes that start and end at the same depot, such that operational constraints, customer requirements are satisfies and cost is minimized. The road network is modelled as a Graph with edges (May be directed) as roads and vertices correspond to (potential) customer locations.

The parameters associated with a **customer/vertex** are

1. Vertex where customer is located
2. Demand associated with Vertex
3. Time windows during which a customer can be serviced
4. Overheads : Like loading and unloading times at depot or customer

The parameters associated with our **fleet of vehicles** is

1. Home depot/node of the vehicle
2. Capacity of vehicle (Number of Passengers). Capacity may be different for different goods
3. Subset of edges of graph that vehicle can traverse (Large Truck/Bus limitation)
4. Cost associated with utilizing a vehicle

Constraints on Routes

It is possible that there is a precedence associated with vertices (Pickup and Delivery problem/ VRP with Backhauls)

Objectives of VRPs may be

1. Minimization of total transportation cost
2. Minimization of the number of vehicles
3. Balancing routes (to maintain traffic balance)
4. Minimization of penalties/tardiness

2 Classes of VRPs

2.1 Capacitated VRP : CVRP

Assumptions :

- Each vehicle undertakes only one route, a route means visiting customers and coming back home
- Demands are known beforehand
- Vehicles are identical
- There is a single central depot (home)
- There are capacity restrictions on vehicles
- The objective is to minimize total cost.

Notation

1. G = (V, A) is a complete Graph (Edge between every pair of distinct vertices)
   - A much stronger condition than the graph being strongly connected
2. V = {0, ..., n} is the set of vertices, 0 is the 'home' node
3. A cost $c_{ij} \geq 0$ is associated with every edge $(i, j) \in A$
   - $c_{ii} = \infty \quad \forall i \in V$
   - If a direct edge (i,j) does not exist in an application then the cost associated with it can be set to ∞
   - If the graph is symmetric then $c_{ij} = c_{ji} \quad \forall i, j \in V$
4. For any edge $e \in E$, $\alpha(e)$ and $\beta(e)$ denote the the endpoint vertices
5. $\Delta^+(i)$ and $\Delta^-(i)$ denote the forward star (Nodes that are directly reachable from i)and backward star (Nodes from which i is directly reachable) of vertex i
6. For any set $S \subseteq V$, $\delta(S)$ denote sets of edges that have one or both endpoints in S, respectively. $\delta(\{i\}) \cong \delta(i)$
7. Each customer i (i = 1,...,n) is associated with a demand $d_i \geq 0$
   - $d_0 = 0$
8. The notation d(S) means $\sum_{i \in S} d_i$
9. The number of identical vehicles is $K \geq K_{min}$, each has capacity C.
   - Assume $d_i \leq C \quad \forall \quad i \in V$
   - K_{min} : Minimum number of vehicles required, obtained by solving associated Bin packing problem (BPP)
10. For any $S \subseteq V \setminus \{0\}$, r(S) is the min. number of vehicles required to serve customers in S (BPP solution associated with just S). A lower bound on r(S) is

$$\lceil \frac{d(S)}{C} \rceil$$

Problem Objective

Find K simple (No vertex repeats) circuits (same start and end point paths) with minimum cost such that

- Each circuit visits the home/depot
- Each customer vertex is only visited by one circuit(vehicle)
- Sum of demands of vertices in a circuit is less than C.
- A variant may have separate capacities C_k, k = (1,...,K) for each vehicle.

2.2 Distance-Constrained VRP

- The maximum capacity constraint is replaced with a max. distance/time constraint
- A length t_{ij} or $t_e \geq 0$ is associated with each edge and the total route length should never exceed a certain T
- There can be service times s_i associated with the stop over at each node
- These service times can be absorbed in the t values as

$$t_{ij} = t'_{ij} + s_i/2 + s_j/2$$

If both capacity and distance have restrictions then the problem is called Distance-Constrained VRP (DCVRP).

2.3 VRP with Time Windows

- Each customer i is associated with the additional parameter $[a_i, b_i]$, i.e. a time window
- The service to each customer must start within the time window. In case of early arrival, the vehicle must wait until a_i

VRPTW is modelled as an asymmetric problem (due to asymmetry introduced by time windows). The Symmetry plays a role in solving but from a formulation point of view the distinction is irrelevant.

Problem Objective

Find a collection of K simple circuits with minimum cost such that

1. Each circuit visits the home/depot
2. Each customer vertex is only visited by one circuit(vehicle)
3. Sum of demands of vertices in a circuit is less than C.
4. The service for customer i starts within the window $[a_i, b_i]$ and the vehicle spends s_i time units at the customer. Not necessary that $a_i + s_i \leq b_i$, i.e. the service end time is irrelevant.

By taking $a_i = 0$, $b_i = \infty$ for each $i \in V \setminus \{0\}$ we get the CVRP.

2.4 VRP with Backhauls

1. The customer set $V \setminus \{0\}$ is partitioned into two subsets
   - Linehaul (Regular customers) : L = {1,...,n}
   - Backhaul (Like Suppliers) B = {n+1,...,n+m}
2. As before, $d_i \geq 0$ is associated with each customer, as either the demand or the quantity to be picked up

Problem Objective

Find a collection of K simple circuits with minimum cost such that

1. Each circuit visits the home/depot
2. Each customer vertex is only visited by one circuit(vehicle)
3. Sum of demands of L and B vertices in a circuit do not separately exceed C (To meet the capacity restriction on the vehicles)
4. In each circuit, all L vertices are visited before any B vertices
    - The origin from this comes from logistical restrictions on how goods can be packed in a truck
5. $K \geq max\{K_L, K_B\}$
    - K_L and K_B denote the min. number of vehicles required to service L and B individually
6. VRPB generalises to CVRP when B $= \phi$

2.5 VRP with pickup and delivery - VRPPD

1. Delivery demand d_i and pickup demand p_i (for homogeneous quantities), are associated with each customer i
2. O_i denotes the origin for the demand d_i
3. D_i denotes the vertex that is the destination for the pickup demand p_i

Problem Objective

Find a collection of K simple circuits with minimum cost such that

1. Each circuit visits the home/depot
2. Each customer vertex is visited by exactly one circuit(vehicle)
3. The instantaneous load of the vehicle travelling along a circuit should never exceed C
4. If for any i, O_i is different from the home/depot, then it must be served in the same circuit as i and before i itself
5. For each i, if D_i is different from home/depot, then it must be serevd in the same circuit as i, after i

VRPPD reduces to CVRP when $O_i = D_i = 0$ and $p_i = 0 \quad \forall\, i \in V$

3 Modelling as Programming Problems

1. Vehicle Flow formulations
    - A binary variable x_{ij} associated with each arc
    - If multiple vehicles use the same edges (Not possible in the current model) then the binary variable can be replaced by an Integer counter which counts the number of times an edge has been used
    - Cannot be used when cost depends on
        - Vertex sequence
        - Type of vehicle
2. Commodity Flow Formulation
    - Additional integer variables associated with the edges
3. Set-Partitioning problem (SPP) using binary variables

3.1 Vehicle Flow Models

VRP 1 - General (Asymmetric) CVRP

Characteristics

- 2 index x_{ij} type formulation
- $\mathcal{O}(n^2)$ binary variables x_{ij}

$$x_{ij} = \begin{cases} 1 & \text{if } (i,j) \in A \ \text{ Belongs to the optimal solution} \\ 0 & \text{Otherwise} \end{cases}$$

- Expression for total cost is

$$\text{Minimize} \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij} \tag{1}$$

  - Since c_{ij} is added to the accumulated sum only if x_{ij} is 1

Subject to the conditions,

$$\sum_{i \in V} x_{ij} = 1 \quad \forall j \in V \backslash \{0\} \tag{2}$$

– From any customer vertex the total inflow should be of exactly 1 edge

$$\sum_{j \in V} x_{ij} = 1 \quad \forall i \in V \backslash \{0\} \tag{3}$$

– The outflow from any customer vertex should be exactly 1 edge

$$\sum_{i \in V} x_{i0} = K, \tag{4}$$

– Exactly K distinct edges must terminate onto the depot vertex, since each of the K circuits will be terminating at the depot

$$\sum_{j \in V} x_{0j} = K, \tag{5}$$

– Exactly K distinct edges must go out of the depot vertex, one for each circuit

– The equations (2) – (4) impose a total of $2|V|$ constraints

$$\sum_{i \notin S} \sum_{j \in S} x_{ij} \geq r(S) \quad \forall S \subseteq V \backslash \{0\},\, S \neq \phi, \tag{6}$$

– Since the number of vehicles entering S through the cut ($V \setminus S$, S) must be more than the minimum number of vehicles required to service S (The global optimality may require us to use more than the minimum no. of vehicles)

$$\sum_{i \notin S} \sum_{j \in S} x_{ij} = \sum_{i \in S} \sum_{j \notin S} x_{ij} \tag{7}$$

– In (7) LHS is the number of edges entering S and RHS is the number of edges leaving S

Using (7), (6) can be restated as

$$\sum_{i \notin S} \sum_{j \in S} x_{ij} \geq r(V \backslash S) \quad \forall S \subset V, 0 \in S. \tag{8}$$

Using (2), (3), (4) and (5) along with (6) we get

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - r(S) \quad \forall S \subseteq V \backslash \{0\}, S \neq \phi, \tag{9}$$

In the (6)-(9), the number of constraints is $\mathcal{O}(2^{|V|})$

It is practically impossible to solve directly the LP relaxation of of the problem in this form
Practically an exponential number of constraints are solved using separation procedures (A topic in OR)

An alternate formulation for VRP-1 has a Polynomial Cardinality of constraints.
Introducing u_i, $i \neq 0$ as the (continuous var.) load of a vehicle after visiting node i
The alternate constraints are

$$u_i - u_j + Cx_{ij} \leq C - d_j \quad \forall\, i, j \in V \backslash \{0\}, i \neq j, \text{such that } d_i + d_j \leq C \tag{10}$$

$$d_i \leq u_i \leq C \quad \forall i \in V \backslash \{0\} \tag{11}$$

Explanation : When $x_{ij} = 0$, constraint (10) is superfluous since it spells $u_i - u_j \leq C - d_j$ which is already given by (11) since $u_i \leq C$ and $u_j \geq d_j$ (Multiply by -1 and add). When $x_{ij} = 1$ they impose $u_j \geq u_i + d_j$

- The origin of the above constraints is involved and is given in the references [3] and [4]

– Lastly, of course

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in V \tag{12}$$

3.2 Symmetric CVRP

The previous constraints and formulations can be modified to obtain a formulation for the S–CVRP

VRP–2

- Only one among x_{ij} and x_{ji} must be used. Pick $i < j$
- The variables x_{0j} and x_{j0} have the range $\{0, 1, 2\}$ if 1 customer routes are to be allowed

VRP 3

Symmetric counterpart of VRP 1 but the 2 index notation is dropped. Instead edges e, are treated as being a part of an abstract set E

VRP–4 : 3 index formulation

VRP 4 : is a 3 index formulation that explicitly associates a vehicle index with an edge

- More involved constraints can be imposed on the 3 index VRP model with the integer variable x_{ijk}
- x_{ijk} counts the number of times arc $(i,j) \in A$ is traversed by vehicle $k \in \{1, ..., K\}$ in the optimal solution. Number of such variables will be $\mathcal{O}(n^2K)$
- We also need an additional $\mathcal{O}(nK)$ binary variables of the type y_{ik} which indicate whether customer i is served by vehicle k in the optimal solution.

$$\text{Minimize} \sum_{i\in V}\sum_{j\in V} c_{ij} \sum_{k=1}^{K} x_{ijk} \tag{1}$$

Subject to the constraints

$$\sum_{k=1}^{K} y_{ik} = 1 \quad \forall i \in V\backslash\{0\} \tag{2}$$

So that every customer is served by exactly one vehicle, which is a condiditon necessary for optimality if

$$\tag{3}$$

VRP 5 : is te symmetric undirected version of VRP 4

VRP 6 : Solves the problem of VRP 1 but with the additional constraint that the graph is not complete, ie some edges are missing. Can be done by associating an M = INF cost with these missing edges.
But VRP 6 considers the case of a sparse graph $|A| \ll n^2$.
m TSP : Replacing a single depot with K vertices (one for each vehicle). Implemented by defining $c_{ij}^{'}$ the cost as

$$c_{ij}^{'} := \begin{cases} c_{ij} & \text{for i,j} \in V\backslash\{0\} \\ c_{i0} & \text{for } i \in V\backslash\{0\}, j \in W \\ c_{0j} & \text{for } i \in W, j \in V\backslash\{0\} \\ \lambda & \text{for } i, j \in W \end{cases} \tag{12}$$

Where W is the set of k home depots and λ is a given weight If we pick $\lambda = M$ then using all the depots becomes necessary.
Cases where vehicles may be left unused are covered by replacing = K constraints with $\leq K$ type constraints.
This may become crucial if there is some fixed cost associated with their use ex. Driver wages.
Non Homogeneous fleet : Easy to account for in 3 index model of VRP 4. A different C_k for each vehicle.

Prevention of single customer routes : In a CVRP, a customer j can be served alone in a route iff the remaining K-1 vehicles can satisfy the rest of the demand

$$r(V \backslash \{j\}) \leq K - 1$$

r(.) may be replaced by its trivial lower bound.

3.3 Commodity Flow Models

Formulation : VRP7

$G' = (V', A')$ which is derived from G but has n+1 as another depot like 0. This is done so that routes are paths from 0 to n+1 instead of circuits.
There are two non negative flow variables associated with every edge y_{ji} and y_{ij}. If a vehicle runs from i to j then y_{ij} is the vehicle load and $y_{ji} = C - y_{ij}$ is the vehicle capacity. The equation

$$y_{ij} + y_{ji} = C$$

holds for every edge (i, j) $\in A'$.
The constraints are given below:

$$(1.62) \qquad (\text{VRP7}) \quad \min \sum_{(i,j)\in A'} c_{ij} x_{ij}$$

subject to

$$(1.63) \qquad \sum_{j\in V'} (y_{ji} - y_{ij}) = 2d_i \qquad \forall\, i \in V' \setminus \{0, n+1\},$$

$$(1.64) \qquad \sum_{j\in V' \setminus \{0,n+1\}} y_{0j} = d(V \setminus \{0, n+1\}),$$

$$(1.65) \qquad \sum_{j\in V' \setminus \{0,n+1\}} y_{j0} = KC - d(V \setminus \{0, n+1\}),$$

$$(1.66) \qquad \sum_{j\in V' \setminus \{0,n+1\}} y_{n+1\,j} = KC,$$

$$(1.67) \qquad y_{ij} + y_{ji} = Cx_{ij} \qquad \forall\, (i, j) \in A',$$

$$(1.68) \qquad \sum_{j\in V'} (x_{ij} + x_{ji}) = 2 \qquad \forall\, i \in V' \setminus \{0, n+1\},$$

$$(1.69) \qquad y_{ij} \geq 0 \qquad \forall\, (i, j) \in A',$$

$$(1.70) \qquad x_{ij} \in \{0, 1\} \qquad \forall\, (i, j) \in A'.$$

Figure 1: Constraint and formulation for VRP7. From [1]

3.4 Set-Partitioning Models

Formulation : VRP8

$\mathcal{H} = \{H_1, H_2, ..., H_q\}$ is the collection of all the feasible circuits associated with the original (G, A) formulation. Each circuit H_j has cost c_j. $a_{ij} \in \{0, 1\}$, $a_{i,j} = 1$ if vertex i is visited by route H_j, and =0 otherwise.
The binary variable x_j, $\quad j = 1, ..., q$ equals 1 iff H_j is part of the optimum solution.

$$(1.71) \qquad \text{(VRP8)} \quad \min \sum_{j=1}^{q} c_j x_j$$

subject to

$$(1.72) \qquad \sum_{j=1}^{q} a_{ij} x_j = 1 \qquad \forall\, i \in V \setminus \{0\},$$

$$(1.73) \qquad \sum_{j=1}^{q} x_j = K,$$

$$(1.74) \qquad x_j \in \{0, 1\} \qquad \forall\, j = 1, \ldots, q.$$

Figure 2: Constraint and formulation for VRP8. From [1]

If the cost matrix satisfies the triangle inequality then we can relax 1.71 and get

$$\sum_{j=1}^{q} a_{ij} x_j \geq 1 \qquad \text{(VRP8}'\text{)}$$

Solved using *column generation*

4 VRP with Time Windows

Formulation VRPTW : Commodity Flow type

G = (V, A), the nodes 0 and n+1 are the depots. A feasible route should start at 0 and end at n+1. The time window associated with the depots 0 and n+1 is $[a_0, b_0] = [a_{n+1}, b_{n+1}] = [E, L]$, where E and L are the earliest possible departure and latest possible arrival. $d_0 = d_{n+1} = s_0 = s_{n+1} = c_{0,n+1} = t_{0,n+1} = 0$.

$$E \leq min_{i \in V \setminus \{0\}} b_i - t_{oi}$$

$$L \geq min_{i \in V \setminus \{0\}} a_i + s_i + t_{i0}$$

Any arc (i, j) is useless if

$$a_i + s_i + t_{ij} > b_j, \text{ or if } d_i + d_j > C$$

The formulation uses the variables x_{ijk} $(i, j) \in A, k \in K, = 1$ if the edge (i, j) is used by vehicle k and 0 otherwise and time variables $w_{ik}, i \in V, k \in K$, which records when the service at vertex i began when serviced by vehicle k.

$$\textbf{(VRPTW)} : \min \sum_{k \in K} \sum_{(i,j) \in A} c_{ij} x_{ijk}$$

subject to

$$(7.2) \qquad \sum_{k \in K} \sum_{j \in \Delta^+(i)} x_{ijk} = 1 \qquad \forall i \in N,$$

$$(7.3) \qquad \sum_{j \in \Delta^+(0)} x_{0jk} = 1 \qquad \forall k \in K,$$

$$(7.4) \qquad \sum_{i \in \Delta^-(j)} x_{ijk} - \sum_{i \in \Delta^+(j)} x_{jik} = 0 \qquad \forall k \in K,\ j \in N,$$

$$(7.5) \qquad \sum_{i \in \Delta^-(n+1)} x_{i,n+1,k} = 1 \qquad \forall k \in K,$$

Figure 3: Constraint and formulation for VRPTW. From [1]

$$(7.6) \qquad x_{ijk}(w_{ik} + s_i + t_{ij} - w_{jk}) \le 0 \qquad \forall k \in K,\ (i,j) \in A,$$

$$(7.7) \qquad a_i \sum_{j \in \Delta^+(i)} x_{ijk} \le w_{ik} \le b_i \sum_{j \in \Delta^+(i)} x_{ijk} \qquad \forall k \in K,\ i \in N,$$

$$(7.8) \qquad E \le w_{ik} \le L \qquad \forall k \in K,\ i \in \{0, n+1\},$$

$$(7.9) \qquad \sum_{i \in N} d_i \sum_{j \in \Delta^+(i)} x_{ijk} \le C \qquad \forall k \in K,$$

$$(7.10) \qquad x_{ijk} \ge 0 \qquad \forall k \in K,\ (i,j) \in A,$$

$$(7.11) \qquad x_{ijk} \in \{0,1\} \qquad \forall k \in K,\ (i,j) \in A.$$

Figure 4: Constraint and formulation for VRPTW. From [1]

Where $N = V\ \{0, n+1\}$ is the set of customers. 7.2 restricts assignment of every customer to exactly one route. Constraints 7.3 says every vehicle leaves the depot. 7.4 is saying that if car k services customer j then it must leave through some out edge. 7.5 says that all vehicles must eventually come back to depot.
7.6 says that, if the edge (i, j) is utilised by vehicle k, then w_{jk} must be more then time when k arrives at i + time spent there + the edge weight t_{ij}.
7.7 says that all incoming vehicles must come in during $[a_i, b_i]$. 7.8 is related to the start and end times. 7.9 puts a restriction based on the maximum capacity C of vehicles.
The constraint 7.6 can be rewritten in a convenient linear from as

$$w_{ik} + s_i + t_{ij} - w_{jk} \le (1 - x_{ijk})M_{ij} \quad \forall k \in K, (i,j) \in A$$

Where M_{ij} are large constants

Lower Bounds on Cost

A 'network' lower bound can be obtained by relaxing capacity and time constraints. If capacity is not an issue, and if $a_i = b_i$ for all $i \in N$, the problem is the fixed schedule problem which has network lower bound as the optimal solution.
LP lower bound on cost can be obtained by lifting the binary integer restriction on x_{ijk} and solving the LP problem.

5 VRP with Backhauls : VRPB or Line-haul–Back-haul problem

There are two sub sets of customers : line-haul, each of who require product and backhaul wehre a given quantity of product must be picked up.
The total distance travelled by the vehicles is to be minimized. The problem is sub-divided into A(symmetric)VRPB and SVRPB.
G = (V, A) is te graph as always. V = $\{0\} \cup L \cup B$. Where L = {1,...,n} and B = {n+1,...,n+m} ie n Linehaul and m Backhaul customers.

5.1 Formulation VRPB Toth and Vigo P1

Define $L_o = L \cup \{0\}$ similarly B_o. Define the graph $\bar{G} = (\bar{V}, \bar{A})$ is a directed graph obtained from G such that $\bar{V} = V$ and $\bar{A} = A_1 \cup A_2 \cup A_3$ where A_1 contains edges from depot and linehaul vertices to linehaul vertices, A_2 contains edges from backhaul vertices to other backhaul vertices or the depot and A_3 are the interface edges between L and B.

$$A_1 := \{(i,j) \in A : i \in L_o, j \in L\}$$

$$A_2 := \{(i,j) \in A : i \in B, j \in B_o\}$$

$$A_3 := \{(i,j) \in A : i \in L, j \in B_o\}$$

$\bar{A}$ essentially removes all the edges which cannot be part of a feasible solution given the constraints.
$\mathcal{L}, \mathcal{B}$ are the family of all the subsets of vertices in L and B respectively. $\mathcal{F} = \mathcal{L} \cup \mathcal{B}$. All other variables carry from the CVRP formulation.

$$\text{(8.1)} \qquad \text{(P1)} \quad v(\text{P1}) = \min \sum_{(i,j)\in\overline{A}} c_{ij}\, x_{ij}$$

subject to

$$\text{(8.2)} \qquad \sum_{i\in\Delta_j^-} x_{ij} = 1 \qquad \forall\, j \in \overline{V} \setminus \{0\},$$

$$\text{(8.3)} \qquad \sum_{j\in\Delta_i^+} x_{ij} = 1 \qquad \forall\, i \in \overline{V} \setminus \{0\},$$

$$\text{(8.4)} \qquad \sum_{i\in\Delta_0^-} x_{i0} = K,$$

$$\text{(8.5)} \qquad \sum_{j\in\Delta_0^+} x_{0j} = K,$$

$$\text{(8.6)} \qquad \sum_{j\in S} \sum_{i\in\Delta_j^-\setminus S} x_{ij} \geq r(S) \qquad \forall\, S \in \mathcal{F},$$

$$\text{(8.7)} \qquad \sum_{i\in S} \sum_{j\in\Delta_i^+\setminus S} x_{ij} \geq r(S) \qquad \forall\, S \in \mathcal{F},$$

$$\text{(8.8)} \qquad x_{ij} \in \{0, 1\} \qquad \forall\, (i, j) \in \overline{A},$$

Figure 5: Constraint and formulation for VRPB P1. From [1]

5.2 Formulation by Mingozi *et al.* P2

A set partitioning model for the AVRPB is described.
Using the terminology of the formulation P1, define two graphs $G_L = (L_o, A_1)$ and $G_B = (B_o, A_2)$ which are sub-graphs of $\bar{G}$. Then for a path P lying entirely in either G_L or entirely in G_B will be feasible if

$$(8.10) \qquad C^L_{\min} \le \sum_{j \in P} d_j \le C \qquad \left(\text{resp., } C^B_{\min} \le \sum_{j \in P} d_j \le C\right),$$

where $C^L_{\min}$ (resp., $C^B_{\min}$) represents the minimum total demand of linehaul customers (resp., backhaul customers) of any feasible route. The values $C^L_{\min}$ and $C^B_{\min}$ can be computed as follows:

$$(8.11) \qquad C^L_{\min} = \max\left\{0, \sum_{j \in L} d_j - (K-1)C\right\}$$

and

$$(8.12) \qquad C^B_{\min} = \max\left\{0, \sum_{j \in B} d_j - (K-1)C\right\}$$

Figure 6: Constraint and formulation for VRPB P2. From [1]

6 VRP with Pick up and Delivery

In the most general case, a fleet of non identical vehicles originating from multiple terminals must satisfy requests defined by,

- Pickup point
- Delivery point
- A demand that is to be transported
- A time window within which service must be provided like VRPTW. Not part of the basic model

The problem being solved is static, ie all requirements are known before making a decision.
The most general problem intended to be solved is the VRP-PD-TW.

6.1 Notation : for n pickup and delivery requests

Request i is associated with 2 nodes, i (for pickup stop) and $n+i$ (for delivery stop). The pickup nodes are P = {1,...,n}, for delivery D = {n+1,...,2n} and $N = P \cup D$
Request i is transporting d_i units of goods from i to n + i. Define a new variable l_i which is a signed version of d_i, ie $\ell_i = d_i, \ell_{n+i} = -d_i$.
K is the set of vehicles, Each vehicle has a set of pick up and delivery points ($N_k = P_k \cup D_k$) associated with it.
Each vehicle gets its own graph $G_k = (V_k, A_k)$. Where $V_k = N_k \cup o(k), d(k)$, where o and d represent the origin and final destination for vehicle k (may be the same). A_k is simply the set

of all possible arcs among the vertices V_k, ie the set $V_k \times V_k$.
The Capacity of vehicle k is C_k
The travel time and cost between nodes $i, j \in V_k,$ by vehicle k are t_{ijk}, c_{ijk} respectively.
Vehicle k leaves unloaded from its origin depot at time $a_{o(k)} = b_{o(k)}$.
A solution for vehicle k corresponds to a pickup and delivery route from o(k) to d(k) lying entirely in G_k.
The service initiation time s_i, at all nodes $i \in N$, must lie in the time-window $[a_i, b_i]$
Early arrival is allowed however the vehicle will need to wait till the a_i.

6.2 Formulation

The model variables are,

$$x_{ijk} := \begin{cases} 1 & \text{if edge (i,j)} \in A_k \text{ is used by vehicle k} \\ 0 & \text{Otherwise} \end{cases}$$

T_{ik} specifies when vehicle k starts service at node $i \in V_k$
L_{ik} gives the load of vehicle k after the service at node $i \in V_k$ has been completed.
The formulation (Taken from section 9.2 of text [1]), follows.

(9.1) $$\min \sum_{k\in K} \sum_{(i,j)\in A_k} c_{ijk} x_{ijk}$$

subject to

(9.2) $$\sum_{k\in K} \sum_{j\in N_k \cup\{d(k)\}} x_{ijk} = 1 \qquad \forall\, i \in P,$$

(9.3) $$\sum_{j\in N_k} x_{ijk} - \sum_{j\in N_k} x_{j,n+i,k} = 0 \qquad \forall\, k \in K,\ i \in P_k,$$

(9.4) $$\sum_{j\in P_k \cup\{d(k)\}} x_{o(k),j,k} = 1 \qquad \forall\, k \in K,$$

(9.5) $$\sum_{i\in N_k \cup\{o(k)\}} x_{ijk} - \sum_{i\in N_k \cup\{d(k)\}} x_{jik} = 0 \qquad \forall\, k \in K,\ j \in N_k,$$

(9.6) $$\sum_{i\in D_k \cup\{o(k)\}} x_{i,d(k),k} = 1 \qquad \forall\, k \in K,$$

(9.7) $$x_{ijk}(T_{ik} + s_i + t_{ijk} - T_{jk}) \le 0 \qquad \forall\, k \in K,\ (i,j) \in A_k,$$

(9.8) $$a_i \le T_{ik} \le b_i \qquad \forall\, k \in K,\ i \in V_k,$$

(9.9) $$T_{ik} + t_{i,n+i,k} \le T_{n+i,k} \qquad \forall\, k \in K,\ i \in P_k,$$

(9.10) $$x_{ijk}(L_{ik} + \ell_j - L_{jk}) = 0 \qquad \forall\, k \in K,\ (i,j) \in A_k,$$

(9.11) $$\ell_i \le L_{ik} \le C_k \qquad \forall\, k \in K,\ i \in P_k,$$

(9.12) $$0 \le L_{n+i,k} \le C_k - \ell_i \qquad \forall\, k \in K, n+i \in D_k,$$

(9.13) $$L_{o(k),k} = 0 \qquad \forall\, k \in K,$$

(9.14) $$x_{ijk} \ge 0 \qquad \forall\, k \in K,\ (i,j) \in A_k,$$

(9.15) $$x_{ijk} \text{ binary} \qquad \forall\, k \in K,\ (i,j) \in A_k.$$

Figure 7: Constraint and formulation for the most General case of VRP-PD

7 References

[1] The Vehicle Routing Problem, Edited by Paolo Toth and Daniele Vigo

[2] NPTEL Lecture series : Advanced Operations Research - Prof. G Srinivasan, IIT Madras

[3] N. Christofides, A. Mingozzi, and P. Toth. The vehicle routing problem. In N. Christofides, A. Mingozzi, P. Toth, and C. Sandi, editors, Combinatorial Opti- mization, Wiley, Chichester, UK, 1979, pp. 315-338

[4] M. Desrochers and G. Laporte. Improvements and extensions to the Miller-Tucker- Zemlin subtour elimination constraints. Operations Research Letters, 10:27-36,1991