International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 10, October - 2013

Integer Programming Model For The Vehicle Routing Problem With Time Windows And Forbidden Path

Dame Ifa S
Graduate School Of Mathematics, the University of Sumatera Utara
Enita Dewi Tarigan
Graduate School Of Mathematics, the University of Sumatera Utara
Nenna Irsa Syahputri
Graduate School Of Mathematics, the University of Sumatera Utara
Herman Mawengkang
Department of Mathematics, The University of Sumatera Utara

Abstract

Generally a Vehicle Routing Problem (VRP) determines the optimal set of routes used by a fleet of vehicles to serve a given set of customers on a predefined graph; the objective is to minimize the total travel cost (related to the travel times or distances) and operational cost (related to the number of vehicles used). In this paper we study a variant of the predefined graph: given a weighted graph G and vertices a and b, and given a set X of forbidden paths in G, find the minimum total travel cost of a-b path P such that no path in X is a subpath of P. Path P is allowed to repeat vertices and edges. We use integer programming model to describe the problem. A feasible neighbourhood approach is proposed to solve the model.

***Keywords**: Vehicle routing problem, forbidden path, integer programming, feasible neighbourhood search*

1. Introduction

The Vehicle Routing Problem (VRP) is defined on a given graph $G = (V, A)$, where $V = \{v_1, v_2, \ldots, v_n\}$ is a set of vertices and $A \subseteq \{(v_i, v_j) : i \neq j, v_i, v_j \in V\}$ is the arc set. An optimal set of routes, composed of a cyclic linkage of arcs starting and ending at the depot, is selected to serve a given set of customers at vertices. The problem aims at minimizing the total travel cost (proportional to the travel times or distances) and operational cost (proportional to the number of vehicles used). This problem was first introduced by Dantzig and Ramser in 1959 to solve a real-world application concerning the delivery of gasoline to service stations. A comprehensive overview of the Vehicle Routing Problem can be found in Toth and Vigo (2002) which discusses problem formulations, solution techniques, important variants and applications.

This paper discusses a variant of VRP in which there may be forbidden route. Forbidden sub-route involving pairs of edges occur frequently (“No left turn”) and can occur dynamically due to rush hour constraints, lane closures, construction, etc. Longer forbidden subpaths are less common, but can arise, for example if heavy traffic makes it impossible to turn left soon after entering a multi-lane roadway from the right. If we are routing a single vehicle it is more natural to find a detour from the point of failure when a forbidden path is discovered.

Logically, a model whereby an algorithm identifies a potential path, and then this path is tried out on the actual network. In case of failure, further tests can be done to pinpoint a minimal forbidden subpath. Because such tests are expensive, a routing algorithm should try out as few paths as possible. In particular it is practically impossible to identify all forbidden paths ahead of time—we have an exponential number of possible paths to examine in the network. Therefore we impose an assumption of having no a priori knowledge of the forbidden paths, and of identifying forbidden paths only by testing feasibility of a path.

In terms of graph the problem can be defined as : given a graph $G(V, A)$, and vertices s and t, and given a set X of forbidden paths in G, find optimal set of route P such that no path in X is a subpath of P. Routes in X *are called* exceptions, and the desired route is called a shortest exception avoiding route. We allow an exception avoiding route to be non-simple graph, i.e., to repeat vertices and edges. In fact the problem becomes hard if the solution is restricted to simple Szeider (2003) . This problem has been called the Shortest Path Problem with Forbidden Paths by Villeneuve and Desaulniers (2005); Ahmed and Lubiw (2009).

2. Preliminary

We are given an directed graph $G(V, A)$ with $n = |V|$ vertices and $m = |A|$ edges where each edge $e \in A$ has a positive weight denoting its length. We are also given a source vertex $s \in V$, a destination vertex $t \in V$, and a set X of route in G. The graph G together with X models a vehicle routing network in which a vehicle cannot follow any route in X because of the physical constraints .. We want to find a shortest route from s to t that does not contain any route in X as a subpath—we make the goal more precise as follows. A route is a sequence of vertices each joined by an edge to the next vertex in the sequence. Note that we allow a route to visit vertices and edges more than once. If a route does not visit any vertex more than once, we explicitly call it a simple route. *A* simple directed route from vertex v to vertex w in G is called a forbidden route or an exception if a vehicle cannot follow the route from v to w because of the physical constraints. Given a set X of forbidden route, a route $(v_1, v_2, v_3, . . . , v_l)$ is said to avoid A if $(v_i, v_i + 1, . . ., v_j) \notin A$ for all i, j such that $1 \leq i < j \leq 1$. A route P from s to t is called a shortest A-avoiding route if the length of P is the shortest among all A-avoiding route from s to t. We will use the term "exception avoiding" instead of "X-avoiding" when A is equal to X, the set of all forbidden paths in G.

3. Vehicle Routing Problem

In the late fifties, Dantzig and Ramser (1959) introduced the VRP, which can be viewed as an m-TSP with customer demands and vehicle capacity. An example of such a VRP is shown in Figure 1. The VRP introduced in Dantzig and Ramser (1959), strictly speaking, is called *Capacitated Vehicle Routing Problem* (CVRP), and is one of the simplest vehicle routing problems.

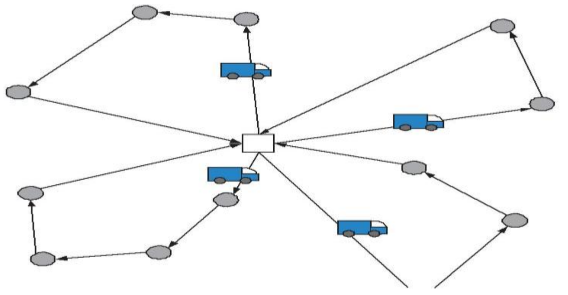

Figure 1: An example of the VRP.

3.1 Problem formulation of VRP

Given a graph $G(V, A)$ with nodes $V = C \cup \{0\}$ and arcs A, where C is the set of customers, and 0 is the depot. Moreover, we have a set R of resources which e.g. can be load and/or time. Each resource $r \in R$ has a resource window $[a_i^r, b_i^r]$ that must be met upon arrival to node $i \in V$, and a consumption $t_{ij}^r \geq 0$ for using arc $(i, j) \in A$. A resource consumption at a node $i \in C$ is modeled by a resource consumption at edge (i, j), and hence usually $t_{0j}^r \geq 0$ for all $j \in C$. A global capacity limit Q can be modeled by imposing a resource window $[0, Q]$ for the depot node 0.

The VRP can now be stated as: Find a set of routes starting and ending at the depot node 0 satisfying all resource windows, such that the cost is minimized and all customers C are visited.

A solution to the VRP will consist of a number of routes

$$
\begin{aligned}
&0 \to i_1^1 \to \cdots \to i_{k_1}^1 \to 0,\\
&0 \to i_1^2 \to \cdots \to i_{k_2}^2 \to 0,\\
&\vdots\\
&0 \to i_1^n \to \cdots \to i_{k_n}^n \to 0
\end{aligned}
$$

where n is the number of vehicles, and k_j is the length of the j'th route.

3.2 Model

In the following let c_{ij} be the cost of arc $(i, j) \in A$, x_{ij} be the binary variable indicating the use of arc (i, j) $\in$ A, and t_{ij}^r (the resource stamp) be the consumption of resource r $\in$ R at the beginning of arc $(i, j) \in A$. Let $\delta^+(i)$ and $\delta^-(i)$ be the set of outgoing respectively ingoing arcs of node $i \in V$. Combining the two index model from Bard et al. [3] with the constraints ensuring the time windows for the ATSP by Ascheuer et al. [1] a mathematical model can be formulated as follows:

$$\min \quad \textstyle\sum_{(i,j)\in A} c_{ij}\, x_{ij} \qquad (1)$$

$$s.t. \quad \textstyle\sum_{(s,j)\in\delta^+(i)} x_{ij} = 1 \qquad \forall\, i \in C \qquad (2)$$

$$\textstyle\sum_{(s,j)\in\delta^-(i)} x_{ij} = \sum_{(s,j)\in\delta^+(i)} x_{ij} \qquad \forall\, i \in V \qquad (3)$$

$$\textstyle\sum_{(s,j)\in\delta^-(i)} \left(T_{ji}^r + t_{ji}^r x_{ij}\right) \leq \sum_{(s,j)\in\delta^+(i)} T_{ji}^r \qquad \forall\, r \in R,\ \forall\, i \in C \qquad (4)$$

$$a_i^r x_{ij} \leq T_{ji}^r \leq b_i^r x_{ij} \qquad \forall\, r \in R,\ \forall\, (i,j) \in A \qquad (5)$$

$$T_{ij}^r \geq 0 \qquad \forall\, r \in R,\ \forall\, (i,j) \in A \qquad (6)$$

$$x_{ij} \in \{0,1\} \qquad \forall\, (i,j) \in A \qquad (7)$$

The objective (1) sums up the cost of the used arcs. Constraints (2) ensure that each customer is visited exactly once, and (3) are the flow conservation constraints. Constraints (4) and (5) ensure the resource windows are satisfied. It is assumed that the bounds on the depot are always satisfied. Note, that no sub-tours can be present since only one resource stamp per arc exists and the arc weights are positive for all $(i, j) \in A : i \in C$.

For a one dimensional resource such as *load* a stronger lower bound of the LP relaxation can be obtained by replacing (4) to (6) with $\sum_{(i,j)\in\delta^{+}(S)} x_{ij} \geq r(S)$, where $r(S)$ is a minimum number of vehicles needed to service the set S. All though this model can not be directly solved it is possible to overcome this problem by only including the constraints that are violated. For more details on how to separate the constraint and calculate the value of $r(S)$ the reader is refered to Toth and Vigo [33].

3.3 Model for VRP with forbidden route

Given a set X of forbidden route, a route $(v_1, v_2, v_3, \ldots, v_l)$ is said to avoid A if $(v_i, v_i + 1, \ldots, v_j) \notin A$ for all i, j such that $1 \leq i < j \leq l$. A route P from s to t is called a shortest A-avoiding route if the length of P is the shortest among all A-avoiding route from s to t. We will use the term "exception avoiding" instead of "X-avoiding" when A is equal to X, the set of all forbidden paths in G.

It is necessarily to assume that costumers are not in the forbidden route.

The model can be written as :

$$\min \ \sum_{(i,j)\in A;(i,j)\notin X} c_{ij}\, x_{ij} \qquad (8)$$

$$s.t. \ \sum_{(s,j)\in\delta^{+}(i)} x_{ij} = 1 \qquad \forall\, i \in C, i \notin X \qquad (9)$$

$$\sum_{(s,j)\in\delta^{-}(i)} x_{ij} = \sum_{(s,j)\in\delta^{+}(i)} x_{ij} \qquad i \notin X, \forall\, i \in V \qquad (10)$$

$$\sum_{(s,j)\in\delta^{-}(i)} (T_{ji}^{r} + t_{ji}^{r} x_{ij}) \leq \sum_{(s,j)\in\delta^{+}(i)} T_{ji}^{r} \qquad i \notin X, \forall\, r \in R, \forall\, i \in C \qquad (11)$$

$$a_i^r x_{ij} \leq T_{ji}^{r} \leq b_i^r x_{ij} \qquad (i,j) \notin X, \forall\, r \in R, \forall\, (i,j) \in A \qquad (12)$$

$$T_{ij}^{r} \geq 0 \ x_{ij} \qquad (i,j) \notin X, \forall\, r \in R, \forall\, (i,j) \in A \qquad (13)$$

$$x_{ij} \in \{0,1\}\, x_{ij} \qquad (i,j) \notin X, \forall\, (i,j) \in A \qquad (14)$$

In the above model every vehicle assigned will not be travelling along the forbidden route.

4. The solution basic approach

$\forall\, r \in R,\ \forall\, (i,j) \in A$ (5) $\forall\, r \in R,\ \forall\, (i, j)$

Consider a mixed integer linear programming (MILP) problem with the following form

$$\text{Minimize } P = c^{T} x \qquad (15)$$

$$\text{Subject to } Ax \leq b \qquad (16)$$

$$x \geq 0 \qquad (17)$$

$$x_j \text{ integer for some } j \in J \qquad (18)$$

A component of the optimal basic feasible vector $(x_B)_k$, to MILP solved as continuous can be written as

$$(x_B)_k = \beta_k - \alpha_{k1}(x_N)_1 - \cdots - \alpha_{kj}(x_N)_j - \cdots - \alpha_{k,n-m}(x_N)_{n-m} \qquad (19)$$

Note that, this expression can be found in the final tableau of Simplex procedure. If $(x_B)_k$ is an integer variable and we assume that β_k is not an integer, the partitioning of β_k into the integer and fractional components is given by

$$\beta_k = [\beta_k] + f_k,\ 0 \leq f_k \leq 1 \qquad (20)$$

suppose we wish to increase $(x_B)_k$ to its nearest integer, $([\beta]+1)$. Based on the idea of suboptimal solutions we may elevate a particular nonbasic variable, say $(x_N)_{j*}$, above its bound of zero, provided α_{kj*}, as one of the element of the vector α_{j*}, is negative. Let Δ_{j*} be amount of movement of the non variable $(x_N)_{j*}$, such that the numerical value of scalar $(x_B)_k$ is integer. Referring to Eqn.(19), Δ_{j*} can then be expressed as

$$\Delta_{f^*} = \frac{1 - f_k}{-\alpha_{kj*}} \qquad (21)$$

while the remaining nonbasic stay at zero. It can be seen that after substituting (21) into (19) for $(x_N)_{j*}$ and taking into account the partitioning of β_k (8) given in (20), we obtain $(x_B)_k = [\beta] + 1$. Thus, $(x_B)_k$ is now an integer.

It is now clear that a nonbasic variable plays an important role to integerize the corresponding basic variable. Therefore, the following result is necessary in order to confirm that must be a non-integer variable to work with in integerizing process.

Theorem 1. Suppose the MILP problem (1)-(4) has an optimal solution, then some of the nonbasic variables. $(x_N)_j$, $j = 1, \ldots, n$, must be non-integer variables.

Proof.

Solving problem as a continuous of slack variables (which are non-integer, except in the case of equality constraint). If we assume that the vector of basic variables x_B consists of all the slack variables then all integer variables would be in the nonbasic vector x_N and therefore integer valued

It is clear that the other components, $(x_B)_{i \neq k}$, of vector x_B will also be affected as the numerical value of the scalar $(x_N)_{j^*}$ increases to Δ_{j^*}. Consequently, if some element of vector α_{j^*}, i.e., α_{j^*} for $i \neq k$, are positive, then the corresponding element of x_B will decrease, and eventually may pass through zero. However, any component of vector x must not go below zero due to the non-negativity restriction. Therefore, a formula, called the minimum ratio test is needed in order to see what is the maximum movement of the nonbasic $(x_N)_{j^*}$ such that all components of x remain feasible. This ratio test would include two cases.

1. A basic variable, $(x_B)_{i \neq k}$ decreases to zero (lower bound) first.
2. The basic variable, $(x_B)_k$ increases to an integer.

Specifically, corresponding to each of these two cases above, one would compute

$$\theta_1 = \min_{i \neq k | \alpha_{j^*} > 0} \left\{ \frac{\beta_i}{\alpha_{j^*}} \right\} \quad (22)$$

$$\theta_2 = \Delta_{j^*} \quad (23)$$

How far one can release the nonbasic $(x_N)_{j^*}$ from its bound of zero, such that vector x remains feasible, will depend on the ratio test θ^* given below

$$\theta^* = \min(\theta_{1,} \theta_2) \quad (24)$$

obviously, if $\theta^* = \theta_1$, one of the basic variable $(x_B)_{i \neq k}$ will hit the lower bound before $(x_B)_k$ becomes integer. If $\theta^* = \theta_2$, the numerical value of the basic variable $(x_B)_k$ will be integer and feasibility is still maintained. Analogously, we would be able to reduce the numerical value of the basic variable $(x_B)_k$ to its closest integer $[\beta_k]$. In this case the amount of movement of a particular nonbasic variable, $(x_N)_{j^*}$, corresponding to any positive element of vector $\alpha_{j'}$, is given by

$$\Delta_{f'} = \frac{f_k}{\alpha_{kj}} \quad (25)$$

In order to maintain the feasibility, the ratio test θ^* is still needed. Consider the movement of a particular nonbasic variable, Δ, as expressed in Eqns.(21) and (25). The only factor that one needs to calculate is the corresponding element of vector α. A vector α_j can be expressed as

$$\alpha_j = B^{-1} a_j, j = 1, \ldots, n - m \quad (26)$$

Therefore, in order to get a particular element of vector α_j we should be able to distinguish the corresponding column of matrix $[B]^{-1}$. Suppose we need the value of element α_{kj^*}, letting v_k^T be the k-th column vector of $[B]^{-1}$, we then have

$$v_k^T = e_k^T B^{-1} \quad (27)$$

subsequently, the numerical value of α_{kj^*} can be obtained from

$$\alpha_{kj^*} = v_k^T a_{j^*} \quad (28)$$

in Linear Programming (LP) terminology the operation conducted in Eqns. (27) and (28) is called the pricing operation. The vector of reduced costs d_j is used to measure the deterioration of the objective function value caused by releasing a nonbasic variable from its bound. Consequently, in deciding which nonbasic should be released in the integerizing process, the vector d_j must be taken into account, such that deterioration is minimized. Recall that the minimum continuous solution provides a lower bound to any integer-feasible solution. Nevertheless, the amount of movement of particular nonbasic variable as given in Eqns. (21) or (25), depends in some way on the corresponding element of vector α_j. Therefore it can be observed that the deterioration of the objective function value due to releasing a nonbasic variable $(x_N)_{j^*}$ so as to integerize a basic variable $(x_B)_k$ may be measured by the ratio

$$\left| \frac{d_k}{\alpha_{kj^*}} \right| \quad (29)$$

where $|a|$ means the absolute value of scalar a. In order to minimize the deterioration of the optimal continuous solution we then use the following strategy for deciding which nonbasic variable may be increased from its bound of zero, that is,

$$\min_j \left\{ \left| \frac{d_k}{\alpha_{kj^*}} \right| \right\}, j = 1, \ldots, n - m \quad (30)$$

From the "active constraint" strategy and the partitioning of the constraints corresponding to basic (B), superbasic (S) and nonbasic (N) variables we can write

$$\begin{bmatrix} B & S & N \\ & & I \end{bmatrix} \begin{bmatrix} x_b \\ x_N \\ x_S \end{bmatrix} = \begin{bmatrix} b \\ b_N \end{bmatrix} \quad (31)$$

or

$$Bx_b + Sx_S + Nx_N = b \quad (32)$$

$$x_N = b_N \quad (33)$$

The basis matrix B is assumed to be square and nonsingular, we get

$$x_B = \beta - Wx_S - \alpha x_N \qquad (34)$$

where

$$\beta = B^{-1}b \qquad (35)$$

$$W = B^{-1}S \qquad (36)$$

$$\alpha = B^{-1}N \qquad (37)$$

Expression (33) indicates that the nonbasic variables are being held equal to their bound. It is evident through the "nearly" basic expression of Eqn. (34), the integerizing strategy discussed in the previous section, designed for MILP problem can be implemented. Particularly, we would be able to release a nonbasic variable from its bound, Eqn. (33), and exchange it with a corresponding basic variable in the integerizing process, although the solution would be degenerate.

Currently, we are in a position where particular basic variable, $(x_B)_k$ is being integerized, thereby a corresponding nonbasic variable, $(c_N)_{j*}$, is being released from its bound of zero. Suppose the maximum movement of $(x_N)_{j*}$ satisfies

$$\theta^* = \Delta_{j*} \qquad (38)$$

such that $(x_B)_k$ is integer valued to exploit the manner of changing the basis, we would be able to move $(x_N)_{j*}$ into B (to replace $(x_B)_k$) and integer-valued $(x_B)_k$ into S in order to maintain the integer solution. We now have a degenerate solution since a basic variable is at its bound. The integerixing process continues with a new set $[B,S]$. In this case, eventually we may end up with all of the integer variables being superbasic.

5. Conclusion

This paper presents a VRP model in which there are some forbidden route. The framework of the model stems from VRP with time windows. Then we exclude the forbidden route from the previous assigned route. We solve the model using a feasible neighbourhood search.

6. References

[1] Daniel Villeneuve and Guy Desaulniers, "The shortest paths with forbidden paths", European Journal of Operational Research, Vol. 165, No. 1, 2005, pp. 97-107.

[2] G.B. Dantzig and J.H. Ramser, "The truck dispatching problem", Management Science, Vol. 6, 1959, 80.

[3] M. Ahmed and A. Lubiw, "Shortest path avoiding forbidden subpaths", Symposium on Theoretical Aspect of Computer Science, 2009, pp. 63-74.

[4] N. Ascheuer, M. Fischetti, and M. Grätschel, "Solving the asymmetric travelling salesman problem with time windows by branch-and-cut", Mathematical Programming, Vol. 90, No. 3, 2001, pp. 475-506.

[5] P. Toth and D. Vigo, "An overview of vehicle routing problems", In P. Toth and D. Vigo, Editors, The Vehicle Routing Problem, Chapter 1, pp. 1-26, SIAM, 2002.

[6] Stefan Szeider, "Finding paths in graphs avoiding forbidden transitions", Discrete Appl. Math., Vol. 126, No. 2-3, 2003, pp. 261-273.