On the Stop Number Minimization Problem

Mourad Baïou, Rafael Colares, Hervé Kerivin

LIMOS, CNRS and Université Clermont II, Clermont-Ferrand, France
{baiou,colares,kerivin}@isima.fr

Keywords : *operations research, combinatorial optimization, branch-and-cut, exact methods*

1 Introduction

We study the Stop Number Minimization Problem (SNMP) first introduced in [1]. Consider a fleet of capacitated vehicles traveling along a closed circuit network in a clockwise direction, and a set of dial-a-ride demands. The SNMP consists of assigning to each demand a vehicle such that no vehicle is overloaded at any point of the circuit and the total number of vehicle's stops is minimized. Notice that a vehicle is allowed to make several tours before serving a demand.

Our efforts focus on the constrained variant of the SNMP where all demands have a unit load and vehicles are obliged to serve all demands in only one tour. This constrained problem is called Unit SNMP and whether or not it is hard to solve remains an open question, while the general SNMP is known to be NP-Hard.

Let K be the set of available vehicles each with capacity $Cap \in \mathbb{Z}_+$, $V = \{0, \dots, n\}$ be the set of stations sequentially ordered as they appear in the circuit network and $E = \{0, \dots, m\}$ be a set of unit load dial-a-ride demands such that each demand $e \in E$ has an origin station $o_e \in V$ and a destination station $d_e \in V$. The binary variable z_e^k equals 1 if demand $e \in E$ is assigned to vehicle $k \in K$, or 0 otherwise, and the binary variable s_v^k equals 1 if vehicle $k \in K$ stops at station $v \in V$, or 0 otherwise. The ILP presented in [1] is the following:

$$\min \quad \sum_{v \in V} \sum_{k \in K} s_v^k$$

s.t.

$$\sum_{k \in K} z_e^k = 1 \qquad \forall e \in E, \tag{1}$$

$$\sum_{e \in Int_v} z_e^k \leq Cap \qquad \forall k \in K, v \in V, \tag{2}$$

$$z_e^k \leq s_v^k \qquad \forall k \in K, e \in E, v \in \{o_e, d_e\} \tag{3}$$

$$z_e^k \in \{0,1\} \qquad \forall k \in K, e \in E, \tag{4}$$

$$s_v^k \in \{0,1\} \qquad \forall k \in K, v \in V, \tag{5}$$

where $Int_v = \{e \in E : o_e \leq v \text{ and } d_e > v\}$ is the set of demands that intersect (have to be on a vehicle at) station $v \in V$.

However, given its weak linear relaxation, traditional Branch-and-Bound approaches have failed to solve this model efficiently. Thus, we search for families of valid inequalities capable of strengthening the linear relaxation. Simultaneously, we look for cases where the problem can be easily solved.

2 Our Contribution

2.1 Special polynomial cases

Consider the graph $G = (V, E)$. Notice that this graph may have multiple edges. We treat the special case where all demands intersect each other at some station $v \in V$ (i.e., $Int_v = E$) and the vehicle's capacity is restricted to 2. In this case, each vehicle can serve at most 2 demands.

First of all, we pack together (put in the same vehicle) pairs of multiple edges, then pairs of isolated edges (edges that do not have any adjacent edges). Let G' be the graph G after deleting the packed edges. We construct the line graph $L(G')$ and search for a maximum matching. Since [2] gives a polynomial time algorithm to solve the maximum matching problem on line graphs, and our packing phase is also polynomial, the whole procedure is done in polynomial time.

2.2 Valid Inequalities

For each $v \in V$, let O_v and D_v be the sets of demands that have station v as their origin, and as their destination, respectively.

The following family of *Strong Capacity inequalities* is valid:

$$\sum_{e \in O_v} z_e^k \leq Cap.s_v^k \qquad \forall k \in K, v \in V, \tag{6}$$

$$\sum_{e \in D_v} z_e^k \leq Cap.s_v^k \qquad \forall k \in K, v \in V. \tag{7}$$

From the *Strong Capacity inequalities* (6)-(7) we are able to derive the following Gomory cuts:

$$\sum_{k \in K} s_v^k \geq \left\lceil \frac{\max\{|O_v|, |D_v|\}}{Cap} \right\rceil \qquad \forall k \in K, v \in V. \tag{8}$$

Let $C \subseteq E$ be a subset of edges and $G[C]$ be the graph induced by the edges in C. Then, a set of edges C is called a connected cover if and only if $|C| > Cap$ and the graph $G[C]$ is connected.

Let $d_G(v)$ denote the degree of node $v \in V$ on graph G and let G_v be the subgraph induced by edges in Int_v. By definition, G_v is bipartite. Then, the following family of *Connected Cover inequalities* is valid :

$$\sum_{e \in C} z_e^k \leq \sum_{u \in V(G[C])} ((d_{G[C]}(u) - 1) s_u^k) \qquad \forall k \in K, C \subseteq E(G_v), v \in V, \tag{9}$$

$$\text{such that } C \text{ is a connected cover.}$$

When $G[C]$ is a tree, the *Connected Cover inequalities* above are capable of cutting fractional extreme points of the polyhedra defined by (1)-(3). On the other hand, if $G[C]$ contains a cycle, the inequality remains valid but do not cut any point. In this case however, we know how to lift the inequality in order to so.

As these inequalities may appear in exponential number, we treat them in a Branch-and-Cut algorithm.

References

[1] V. Pimenta, A. Quilliot, H. Toussaint and D. Vigo, Models and algorithms for reliability-oriented Dial-a-Ride with autonomous electric vehicles. *European Journal of Operational Research 000* (2016) 1–13.

[2] S. Shiraishi, A remark on maximum matching of line graphs. *Discrete Mathematics 179* (1998) 289–291.