AN EXACT ALGORITHM FOR TOLL OPTIMIZATION

Sophie Dewez
Service d'optimisation, Université libre de Bruxelles
E-mail : sdewez@smg.ulb.ac.be

Martine Labbé
Service d'optimisation, Université libre de Bruxelles

Patrice Marcotte
Centre de recherche sur les transports (CRT), Université de Montréal

Gilles Savard
Groupe d'études et de recherche en analyse des décisions (GERAD), Université de Montréal

ETENDED ABSTRACT

In this paper we are interested in the taxation of highway and telecommunication networks. We are looking for the optimal taxes to put on the arcs belonging to the company or government. An optimal taxation method consists in determining level of taxes low enough to favor the use of taxed arcs, and high enough to get important profits, see e.g. [1],[2],[3]. In order to take into account this interaction, this problem is formulated as a bilevel bilinear program. So there are two decisions levels : the leader one and the follower one. Both levels want to optimize its economic function (maximize the profit or minimize the cost).

There does not exist any exact algorithm neither for the general bilevel bilinear problem, neither for this particular problem.

The problem can be formulated as follow :

$$\max_t tx$$

$$\min_{x,y} (c + t)x + dy$$

$$s.t. Ax + By = b$$

$$x, y \geq 0,$$
where $x$ and $y$ are the vectors of taxed and untaxed arcs respectively, $t$ is the vector of taxes associated to vector $x$, $c$ and $d$ are cost vectors associated to vectors $x$ and $y$ respectively, and $A$ and $B$ are the incidence matrices of the arcs.

A first step consists in rewriting the program in order to obtain a one-level bilinear program. For a fixed vector $t$, the second level is linear. So we can replace it by its dual-primal optimality conditions:

$$\begin{align*}
\max \, & t^\top x \\
\text{s.t.} \, & Ax + By = b \\
& x, y \geq 0 \\
& \lambda A \leq c + t \\
& \lambda B \leq d \\
& (c + t - \lambda A) x = 0 \\
& (d - \lambda B) y = 0.
\end{align*}$$

Adding the two last constraints and including a new variable $t' = tx$ in order to linear the program, give:

$$\begin{align*}
\max \, & \sum_{k \in K} \sum_{(i,j) \in A} f^{i,j,k} t_{ij}' \\
\text{s.t.} \, & -\sum_{i:(i,j) \in A} x^k_{ij} - \sum_{i:(i,j) \in B} y^k_{ij} + \sum_{l:(j,l) \in A} x^k_{jl} + \sum_{l:(j,l) \in B} y^k_{jl} = \begin{cases} -1 & \text{if } j = O(k), \\ 1 & \text{if } j = D(k), \\ 0 & \text{otherwise} \end{cases} \forall k \in K, j \in N \\
& \lambda^k_j - \lambda^k_i \leq c_{ij} + t_{ij} \forall k \in K, (i,j) \in A, \\
& \lambda^k_j - \lambda^k_i \leq d_{ij} \forall k \in K, (i,j) \in B, \\
& \sum_{(i,j) \in A} (c_{ij} x^k_{ij} + t^k_{ij}) + \sum_{(i,j) \in B} d_{ij} y^k_{ij} = \lambda^k_{D(k)} - \lambda^k_{O(k)} \forall k \in K, \\
& t^k_{ij} \leq M^k_{ij} x^k_{ij} \forall k \in K, (i,j) \in A, \\
& t_{ij} - t^k_{ij} \leq N_{ij} (1 - x^k_{ij}) \forall k \in K, (i,j) \in A, \\
& x^k_{ij}, y^k_{ij}, t^k_{ij} \text{ and } t_{ij} \geq 0 \forall k \in K, (i,j) \in A \cup B.
\end{align*}$$
Notations are explained in the following table.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>set of nodes</td>
</tr>
<tr>
<td>$A$</td>
<td>set of taxed arcs</td>
</tr>
<tr>
<td>$B$</td>
<td>set of untaxed arcs</td>
</tr>
<tr>
<td>$K$</td>
<td>set of commodities</td>
</tr>
<tr>
<td>$k \in K$</td>
<td>a set of users having the same origin and the same destination</td>
</tr>
<tr>
<td>$O(k)$</td>
<td>origin of commodity $k$</td>
</tr>
<tr>
<td>$D(k)$</td>
<td>destination of commodity $k$</td>
</tr>
<tr>
<td>$f^k$</td>
<td>demand of the commodity $k$</td>
</tr>
<tr>
<td>$x^k_{ij}$</td>
<td>flow on the taxed arc $(i, j)$ for the commodity $k$</td>
</tr>
<tr>
<td>$y^k_{ij}$</td>
<td>flow on the untaxed arc $(i, j)$ for the commodity $k$</td>
</tr>
<tr>
<td>$t_{ij}$</td>
<td>tax on the arc $(i, j)$</td>
</tr>
<tr>
<td>$c_{ij}$</td>
<td>cost of the taxed arc $(i, j)$</td>
</tr>
<tr>
<td>$d_{ij}$</td>
<td>cost of the untaxed arc $(i, j)$</td>
</tr>
<tr>
<td>$M^k_{ij}$</td>
<td>constant depending on the taxed arc $(i, j)$ for the commodity $k$</td>
</tr>
<tr>
<td>$N^k_{ij}$</td>
<td>constant depending on the taxed arc $(i, j)$ over all commodities</td>
</tr>
</tbody>
</table>

We use this formulation to develop an exact algorithm. We first look for good bounds on parameters $M^k_{ij}$ and $N^k_{ij}$. All of the bounds we propose can be computed in polynomial time by applying some shortest path algorithms. Further these bounds do not dominate each other. Detailed computational experiments show that the inclusion of those bounds in the linear program improves a lot the value at the relaxation, the number of nodes used in the branch-and-bound algorithm (we use CPLEX 7.00) and the time of resolution. Then we add cuts depending on the optimality constraints and on the network structure.

**REFERENCES**

