A bilevel model of taxation and its application to optimal highway pricing.

Martine LABBÉ
Institut de Statistique et Recherche Opérationnelle
Université libre de Bruxelles

Patrice MARCOTTE
Centre de recherche sur les transports
and
Département d'informatique et recherche opérationnelle
Université de Montréal

Gilles SAVARD
GERAD and Département de mathématiques et génie industriel
Ecole Polytechnique

May 1998

Abstract: We consider a bilevel model where the leader wants to maximize revenues from a taxation scheme, while the follower rationnally reacts to those tax levels. We focus our attention on the special case of a toll-setting problem defined on a multicommodity transportation network. We show that the general problem is NP-complete, while particular instances are polynomially solvable. Numerical examples are given.

Keywords: Pricing. Networks. Bilevel.

1 Introduction

We consider a general taxation model involving two levels of decision-making. The upper level (leader) imposes taxes on a specified set of goods or services while the lower level (follower) optimizes its own objective, taking into account the taxation scheme devised by the leader. Since we do not force nonnegativity constraints on taxes, our model can implicitly deal with subsidies, which can be viewed as negative taxes. In the field of economics, this fits the principal/agent paradigm (see van Ackere [21]) where the principal, fully aware of the agent's rational behaviour, induces cooperation from the agent through an incentive scheme. In the field of mathematical programming, this problem belongs to the class of bilevel optimization problems where both objective functions are bilinear. Surprisingly, this class has never been studied from a theoretical point of view. The present paper is devoted to the theoretical and algorithmical properties of this taxation model. In particular we address the issues of computational complexity of both the general model and important subclasses of it.

In the first part of the paper, we introduce the general model and reformulate it as a standard linear-linear (or simply linear) bilevel program. Based on this reduction process, we derive an economic interpretation for an optimal taxation strategy. Linear bilevel programs have been studied quite extensively. They constitute the simplest instance of bilevel programs, where the two objective functions and the constraints are linear. Efficient implicit enumeration algorithms for their solution have been proposed by Bard and Moore [4], Faustino and Júdice [7] and Hansen, Jaumard and Savard [10], while exact penalty approaches have been investigated by Anandalingam and White [2, 23]. Nonlinear bilevel models have previously been applied to network problems such as the network design problem (see Marcotte [16]) and to the management of electric utility demand (Hobbs and Nelson [9]). A bilevel price control problem with linear upper level objective and bilinear lower level objective has been proposed, but not solved, by Bialas and Karwan [6].

The linear bilevel programming problem has been shown to be NP-complete by Jeroslow [12]. Penalty schemes have been investigated, dating back to Aiyoshi and Shimizu [1] in the nonlinear case. Recently, Marcotte and Zhu [17] have generalized this result to bilevel problems where the lower level is described by a variational inequality. An annotated bibliography containing more than one hundred references on bilevel programming has been compiled by Vicente and Calamai [22], while the books by Shimizu, Ishizuka and Bard [19] and Luo, Pang and Ralph [14] are devoted, in full or in part, to this subject.

In the second part of the paper, we focus our attention on the problem of setting tolls on a specified subset of arcs of a multicommodity transportation network. In this context the leader corresponds to the profit-maximizing owner of the network, and the follower to users travelling between nodes of the network. The users are assigned to shortest paths with respect to a generalized cost equal to the sum of the actual cost of travel plus a money equivalent of travel time. This model is formulated as a bilevel program with bilinear objectives at both levels of decision and network constraints at the lower level. This problem is shown to be strongly NP-hard. However its lower level network structure makes for very interesting algorithmic considerations. In particular, special cases of the problem can be solved in polynomial time, using shortest path and transshipment algorithms applied to a modified network. Furthermore, the general problem can be efficiently reformulated as a mixed integer program with a small number of integer (indeed binary) variables, large instances of which can be solved within reasonable time.

The outline of the paper is as follows. In Section 2, we introduce a very general nonlinear model and study in detail its specialization to bilinear objectives. Section 3 is devoted to the theoretical and algorithmical analysis of the Toll Optimization Problem (TOP). We prove that the general optimization problem is NP-hard, and that important special cases can be solved in polynomial time. In Section 4, we illustrate the features of the TOP model by means of medium-sized example. In Section 5, we propose a multiclass extension of the basic model and conclude the paper.

2 A general taxation problem

Let x and y be real vectors that specify the levels of taxed and untaxed activities (goods, services), and T be a tax vector attached to the activity vector x. For a given tax level vector T, in control of the leader, the follower strives to minimize its operating costs, while the leader seeks to maximize its revenues from taxes. Let F and f denote the leader's and follower's objective functions, respectively. This situation can be expressed as the bilevel mathematical program

$$
\begin{aligned}
\max_{T,x,y} \quad & F(x,y,T) \\
\text{subject to} \quad & T \in \Theta \\
& (x,y) \in \Pi \\
& (x,y) \in \arg\min_{(x',y')\in\Pi} f(x',y',T)
\end{aligned}
\tag{1}
$$

where Θ represents a set of constraints on taxes, and Π a set of constraints on the activities of the follower. If the lower level problem admits a unique solution denoted by $(x(T), y(T))$, implicitly defined by the inner optimization problem in (1), it is possible to rewrite the above program in the form

$$\max_{T\in\Theta} F(x(T), y(T), T). \tag{2}$$

Whenever the lower problem admits multiple solutions, ties are broken in favour of the leader. Alternative situations, where the leader expects the worse behaviour from the follower, have been analyzed by Loridan and Morgan [15]. According to usual bilevel programming practice (see [5]), we record the programs (1) or (2) as:

$$\begin{aligned} \max_{T\in\Theta,x,y} \quad & F(x, y, T) \\ \min_{x,y} \quad & f(x, y, T) \\ \text{subject to} \quad & (x, y) \in \Pi. \end{aligned}$$

This simple model can cover a wide variety of situations. For instance the vector T may embody subsidies as well as taxes, while the vectors x and y may represent either consumption or production levels. Furthermore, the lower level can be used to represent the group behavior of individuals competing among themselves. For instance, the lower level can involve several or infinitely many economic agents, corresponding to oligopolistic or perfect competition situations, respectively. The equilibrium state of the lower level is then characterized as a solution to a variational inequality parameterized in the leader's decision variables. This yields the generalized bilevel program of Marcotte and Zhu [17], or the equivalent MPEC (Mathematical Program with Equilibrium Constraints) of Luo et al. [14])

$$\begin{aligned} \max_{T\in\Theta,x,y} \quad & F(x, y, T) \\ \text{subject to} \quad & (x, y) \in \Pi \\ & \langle G(x, y, T), (x - y) - (x', y')\rangle \leq 0 \qquad \forall (x', y') \in \Pi \end{aligned}$$

where G is the mapping associated with the lower level variational inequality.

This program can be formulated as a standard bilevel program, if one substitutes for the lower level variational inequality the gap function defined as:

$$f(x,y,T) = \max_{(x',y')\in\Pi} \langle G(x,y,T),(x,y)-(x',y')\rangle.$$

The above formulation is valid only if the set Π is compact. If Π is not compact, one may alternatively use the regularized gap function (see Fukushima [8])

$$f(x,y,T) = \max_{(x',y')\in\Pi} \langle G(x,y,T),(x,y)-(x',y')\rangle - \frac{1}{2}\|(x,y)-(x',y')\|^2$$

as the lower level objective function.

In this paper, we restrict our attention to the situation where the leader maximizes its revenues from taxes restricted to the polyhedron $\Theta = \{T : TC \geq e\}$ defined by a matrix C and a right-hand side vector e while, for given taxation levels, the lower level's reaction is obtained by solving a linear program over the polyhedral set $\Pi = \{(x,y) : Ax + By \geq b\}$. This yields the bilinear-bilinear bilevel program [1]

$$\text{(TAX)} \qquad \begin{array}{rl} \displaystyle\max_{T,x,y} & Tx \\ \text{subject to} & TC \geq e \\ \displaystyle\min_{x,y} & (c+T)x + dy \\ \text{subject to} & Ax + By \geq b. \end{array}$$

The vector c corresponds to the "before tax" cost of the activity x. In order to gain some insight into the above problem, we discuss the example depicted in Figure 1. For a given tax level T in R^1, the optimal solution of the lower level problem is achieved at a vertex of

[1] In order to avoid the use of the transpose operator, we adopt the convention that left (right) vectors are row (column) vectors, respectively.

the two-dimensional polyhedron Π. For instance, if $T = 0$, we have:

$$x(T) = x^0 \qquad y(T) = y^0.$$

The vertex (x^0, y^0) remains optimal as long as T does not exceed some threshold value T^0. For T less than or equal to T^0, the leader's objective is a linear function of T, with slope x^0. As T increases beyond T^0, the vertex (x^0, y^0) is no longer optimal for the lower level's program. Let (x^1, y^1) be the new optimal vertex; (x^1, y^1) remains optimal for T-values belonging to the interval $(T^0, T^1]$. Within this range, the slope of the leader's objective function is now x^1, with $x^1 < x^0$, obviously. The remainder of the process is self-explanatory. If T exceeds a sufficiently large value (T^6 in our example), the optimal vertex is (x^7, y^7), with $x^7 = 0$, and the leader's profit is driven to zero. The optimal solution is achieved at $T = T^2$. The upper envelope of the leader's objective, taking into account the optimal reaction of the follower, is illustrated in Figure 2.

Figure 1 goes here

Figure 2 goes here

Notice that the leader's objective $F(x(T), y(T), T)$ is neither a continuous nor a convex function of T (see Figure 2). However, it is not difficult to prove that it is upper semicontinuous. Hence there exists at least one optimal solution to the general problem, and to TAX in particular.

Throughout the paper we make the following assumptions, to avoid trivial situations:

Assumption 1 *The set* $\Pi = \{(x, y) : Ax + By \geq b\}$ *is bounded.*

Assumption 2 *The set* $\{y : By \geq b\}$ *is nonempty.*

Assumptions 1 and 2 imply that, for any given T, strong duality holds for the lower level program. According to Assumption 2, the lower level has a feasible solution that consists solely of untaxed activities. Consequently, the leader's profit is bounded from above. Our first results state that one can reduce TAX to either a single-level bilinear program with disjoint constraints, or to a linear bilevel program, two classes of problems for which reasonably efficient algorithms have been proposed. Our line of proof uses an exact penalty scheme similar to that introduced in [2].

Proposition 1 *Under Assumptions 1 and 2,* TAX *can be reformulated as a single-level bilinear program.*

Proof. Let us replace the lower level linear program in TAX by its primal-dual optimality constraints to yield the equivalent single-level problem

$$\begin{array}{lll} \max_{T,x,y,\lambda} & Tx & \\ \text{subject to} & TC \geq e & \\ & Ax + By \geq b & \text{(primal feasibility)} \\ & \lambda A = c + T & \\ & \lambda B = d & \text{(dual feasibility)} \\ & \lambda \geq 0 & \\ & \lambda(Ax + By - b) = 0. & \text{(complementarity slackness)} \end{array}$$

Let us penalize the complementarity slackness constraint $\lambda(Ax + By - b) = 0$, which is nonnegative whenever (x, y) and λ are feasible for the primal and the dual problems respectively.

This yields the bilinear program:

$$\text{(BILIN1)} \qquad \begin{array}{rl} \max_{T,x,y,\lambda} & Tx - K\lambda(Ax + By - b) \\ \text{subject to} & TC \geq e \\ & Ax + By \geq b \\ & \lambda A = c + T \\ & \lambda B = d \\ & \lambda \geq 0, \end{array}$$

or, after substituting $\lambda A - c$ for T:

$$\text{(BILIN2)} \qquad \begin{array}{rl} \max_{x,y,\lambda} & (\lambda A - c)x - K\lambda(Ax + By - b) \\ \text{subject to} & Ax + By \geq b \\ & (\lambda A - c)C \geq e \\ & \lambda B = d \\ & \lambda \geq 0. \end{array}$$

Since strong duality holds for the lower level problem, a dual optimal solution of the lower level problem is achieved at an extreme point of the polyhedron $\{\lambda : \lambda B = d, \lambda \geq 0\}$. Therefore the polyhedron $\Gamma = \{\lambda : \lambda B = d, (\lambda A - c)C \geq e, \lambda \geq 0\}$ may be assumed bounded. Denote by $\{\lambda^i, i \in I\}$ its extreme points and by $\{(x^j, y^j), j \in J\}$ the set of extreme points of the primal polyhedron Π. According to Assumption 1, we may also assume, without loss of generality, that $\Pi = \text{conv}_{j \in J}\{(x^j, y^j)\}$. The maximum of the disjoint bilinear program BILIN2 must be achieved at an extreme point (x^j, y^j, λ^i) of $\Pi \times \Gamma$ (see [20]). Let

$$K \geq K^* = \max_{i \in I, j \in J} \left\{ \frac{(\lambda^i A - c)x^j}{\lambda^i(Ax^j + By^j - b)} : \lambda^i(Ax^j + By^j - b) > 0 \right\}. \tag{3}$$

According to this choice of K, a triple (x^j, y^j, λ^i) qualifies for optimality if

$$\lambda^i(Ax^j + By^j - b) = 0.$$

Indeed, at any other extreme point, the leader's objective is negative, and therefore clearly non optimal. It follows that the term $\lambda(Ax + By - b)$ constitutes an exact penalty function for the mathematical program BILIN1, thus showing that TAX and BILIN1 are equivalent, whenever K is larger than K^*. $\square$

Proposition 2 *Under Assumptions 1 and 2,* TAX *can be reformulated as a linear bilevel program.*

Proof. Let us rewrite BILIN2 as:

$$\text{(BILIN3)}\quad \begin{aligned} \min_{x,y,\lambda} \quad & \lambda((K-1)Ax - Kb) + cx + Kdy \\ \text{subject to} \quad & Ax + By \geq b \\ & (\lambda A - c)C \geq e \\ & \lambda B = d \\ & \lambda \geq 0. \end{aligned}$$

For fixed x and y, BILIN3 is a linear program in the variable λ. If one substitutes its dual objective function $dz + (e + cC)u$ for its primal objective $\lambda((K-1)Ax - Kb)$, we obtain the linear-linear bilevel program

$$\begin{aligned} \min_{x,y} \quad & dz + (e + cC)u + cx + Kdy \\ \max_{z,u} \quad & dz + (e + cC)u \\ \text{subject to} \quad & Ax + By \geq b \\ & (K-1)Ax - Bz - ACu \geq Kb \\ & u \geq 0. \end{aligned} \tag{4}$$

$\square$

Note that the constraint $Ax + By \geq b$ in the above program is independent of z and can therefore be arbitrarily located before or after the lower level's objective. The optimal tax vector T is related to the optimal dual variable λ associated with the constraint

$$(K-1)Ax - Bz - ACu \geq Kb$$

through the equation $\lambda A = c + T$, and can be easily recovered.

This latter program can be given an interesting economic interpretation. Consider, for the sake of simplicity, that the taxation vector T is unrestricted, i.e., C and e are zero. Let us also assume, without loss of generality, that c is equal to zero; indeed, upon the introduction of an untaxed activity vector y', TAX is clearly equivalent to the mathematical program

$$
\begin{array}{rl}
\max\limits_{T,x,y,y'} & Tx \\[2ex]
\min\limits_{x,y,y'} & Tx + dy + cy' \\[2ex]
\text{subject to} & Ax + By \geq b \\[2ex]
 & x - y' \geq 0 \\[2ex]
 & -x + y' \geq 0,
\end{array}
$$

where the taxable vector x has now zero "before tax" cost.

With C and e set to zero, the next-to-last constraint of (4) can be rewritten as:

$$\frac{K-1}{K}Ax - \frac{Bz}{K} \geq b.$$

We now make the change of variables $z' = -z/K$ and set $\epsilon = 1/K$. This yields the equivalent linear bilevel program

$$
\begin{array}{rl}
\max\limits_{x,y,z'} & dz' - dy \\[2ex]
\text{subject to} & Ax + By \geq b \\[2ex]
\min\limits_{z'} & dz' \\[2ex]
 & Ax + Bz' \geq b + \epsilon Ax.
\end{array}
\tag{5}
$$

The leader positions himself by selecting an (x, y)-vector such that any marginal deviation from this proposed solution by the follower will result in a large deterioration for the follower.

This gap between the leader's proposal and the follower's second best alternative provides room for taxation. It is in the interest of the leader to maintain this gap as wide as possible.

This is exactly what the above program (5) achieves. Indeed, the lower level in (5) strives to satisfy a marginal increase in demand ϵAx at the least possible cost, while setting x at the level prescribed by the leader.

If ϵ is set to zero, a value of dz' that is less or equal to that of dy can clearly be achieved by the follower, since the value dy corresponds to the feasible lower level program $z' = y$. This means that, at best, the leader's objective is zero. The leader may achieve this optimal value zero by setting y to z', where z' is an optimal response to any feasible vector x selected by the leader. If ϵ is positive, the follower optimally adjusts himself to the increased demand ϵAx by adopting a recourse involving the sole vector z'. In the bilevel formulation (5), the leader wants to maximize the added cost of this recourse to the follower.

If the last inequality of (5) is equivalently expressed as

$$A[(1-\epsilon)x] + Bz' \geq b,$$

the aim of the lower level is simply to substitute z'-variables for a marginal decrease ϵ in the upper level vector x, while meeting the demand vector b. This alternative constitutes the follower's marginal best recourse. Since the penalty scheme is exact, there exists a value ϵ^* such that the above marginal analysis is exact, i.e., there exists an optimal recourse that corresponds to an extremal solution of the constraint polyhedron and this extremal solution remains optimal when ϵ stays within the range $(0, \epsilon^*]$. As stated earlier, the optimal tax vector T can easily be recovered from the dual vector associated with the constraint of the lower level program through the equation

$$T = \lambda A.$$

For fixed x, the optimal solution of the mathematical program (5) is easily obtained by solving two linear programs parameterized in x. It follows that the objective of (5) is continuous, as the difference of two convex, continuous functions of the vector x. This is to be contrasted with the situation illustrated in Figure 2, where the profit function is a piecewise linear but discontinuous function of the decision variable T.

3 A ROAD PRICING MODEL

In this section we focus on an application of the preceding model where an authority sets tolls on a specified subset of arcs of a transportation network, and the users of the network travel on shortest routes between their relative origins and destinations. While the aim pursued by the authority could be to improve the network performance through its toll policy, we assume that it simply seeks to maximize revenues. Such a model corresponds for instance to the setting of profit maximizing tolls on a privately owned highway system. An optimal toll policy is such that toll levels are sufficiently low not to deter the users from taking toll arcs rather than alternative routes, simultaneously generating high revenues. The route choice model that we adopt in this section is simplistic: we assume that the users travel on shortest origin-destination routes, and that congestion is not an influential factor.

In this section we first introduce the model and the notation, together with an integer programming formulation and a proof of NP-completeness; next we show that, under additional assumptions, the problem becomes polynomially solvable.

3.1 The model: formulations and computational complexity

Consider a multicommodity transportation network characterized by an underlying graph, a cost structure and demand for travel between vertices, given by an origin-destination matrix. The leader is allowed to set taxes on a subset of the arcs of the network, while the remaining arcs bear only fixed costs. For given tax levels, the network users minimize their generalized travel costs, which include two components, the first being the tolls and the second the travel costs, exclusive of tolls. The cost of a route is defined as the sum of the generalized costs of its arcs. The only constraints on tolls are lower bound constraints. Most of our results would extend easily to upper bounded tolls.

If one adopts the notation of Table 1, the toll setting problem can be expressed as a bilevel program with bilinear objectives and linear constraints:

$$
\begin{array}{rll}
\text{(TOP)} & \displaystyle\max_{T,x} & \displaystyle\sum_{a\in A_1} T_a x_a \\
 & \displaystyle\min_{x} & \displaystyle\sum_{a\in A_1} (c_a + T_a)x_a + \sum_{a\in A_2} c_a x_a \\
 & \forall i,k,l \in N\times\Omega\times\Delta: & \displaystyle\sum_{a\in i^+} x_a^{kl} - \sum_{a\in i^-} x_a^{kl} = \begin{cases} 1 & \text{if } i=k \\ -1 & \text{if } i=l \\ 0 & \text{otherwise} \end{cases} \\
 & \forall a\in A: & \displaystyle x_a = \sum_{k,l} d^{kl} x_a^{kl} \\
 & \forall a,k,l \in A\times\Omega\times\Delta: & x_a^{kl} \geq 0 \\
 & \forall a \in A_1: & T_a \geq l_a.
\end{array}
$$

In this formulation, the vector $\{x_a\}_{a\in A_1}$ plays the role of x and $\{x_a\}_{a\in A_2}$ plays the role of y in TAX.

In the remainder of the paper we assume that there cannot exist a toll setting scheme

Table 1: Notation for the toll setting problem

N:	set of nodes of the network
Ω:	set of origin nodes of the network
Δ:	set of destination nodes of the network
A:	set of arcs of the network
$a = (i, j)$:	an element of A
A_1:	set of toll arcs
A_2:	set of untolled (free) arcs; $A_2 = A - A_1$
i^-:	Backward star associated with node i: $i^- = \{(k, i) \in A : k \in N\}$
i^+:	Forward star associated with node i: $i^+ = \{(i, k) \in A : k \in N\}$
c_a:	travel cost on arc $a \in A$, exclusive of toll
T_a:	toll on arc $a \in A_1$
l_a:	lower bound on T_a
d^{kl}:	proportion of flow demand between origin node k and destination node l
x_a^{kl}:	flow from origin k to destination l on arc a
x_a:	total flow on arc a,

that generates profits and creates a negative cost cycle in the network, and that there exists at least one path composed solely of untolled arcs, for each origin-destination pair. These assumptions imply that the lower level optimal solution corresponds to a set of shortest paths and that the upper level profit is bounded from above.

Using the ideas of Section 2, TOP can be reduced to a linear bilevel program. Furthermore, the characterization of lower level solutions as origin-destination paths carrying either no flow or the total origin-destination flow allows us to obtain an integer programming formulation of TOP that involves a small number of binary variables, namely

$$\text{(TOP-MIP)} \qquad \max_T \quad \sum_{k,l} \sum_{a \in A_1} d^{kl} T_a^{kl}$$

$$\forall a = (i, j) \in A_1, k, l \in \Omega \times \Delta : \qquad \lambda_i^{kl} - \lambda_j^{kl} \le c_a + T_a$$

$$\forall a = (i, j) \in A_2, k, l \in \Omega \times \Delta : \qquad \lambda_i^{kl} - \lambda_j^{kl} \le c_a$$

$$\forall k,l \in \Omega \times \Delta : \quad \sum_{a \in A_1} (c_a x_a^{kl} + T_a^{kl}) + \sum_{a \in A_2} c_a x_a^{kl} = \lambda_l^{kl} - \lambda_k^{kl}$$

$$\forall i,k,l \in N \times \Omega \times \Delta : \quad \sum_{a \in i^+} x_a^{kl} - \sum_{a \in i^-} x_a^{kl} = \begin{cases} 1 & \text{if } i = k \\ -1 & \text{if } i = l \\ 0 & \text{otherwise} \end{cases}$$

$$\forall a,k,l \in A_1 \times \Omega \times \Delta : \quad -Mx_a^{kl} \leq T_a^{kl} \leq Mx_a^{kl}$$

$$\forall a,k,l \in A_1 \times \Omega \times \Delta : \quad -M(1 - x_a^{kl}) \leq T_a^{kl} - T_a \leq M(1 - x_a^{kl})$$

$$\forall a,k,l \in A_1 \times \Omega \times \Delta : \quad x_a^{kl} \in \{0,1\}$$

$$\forall a,k,l \in A_2 \times \Omega \times \Delta : \quad x_a^{kl} \geq 0$$

$$\forall a \in A_1 : \quad T_a \geq l_a.$$

This reformulation is obtained by first replacing the lower level problem by its optimality conditions. The only nonlinear constraint of the resulting program is

$$\forall k,l \in \Omega \times \Delta : \sum_{a \in A_1} (c_a x_a^{kl} + T_a x_a^{kl}) + \sum_{a \in A_2} c_a x_a^{kl} = \lambda_l^{kl} - \lambda_k^{kl}$$

and states that the primal and dual objective values must be equal. We linearize this constraint replacing $T_a x_a^{kl}$ by T_a^{kl} and adding the constraints

$$\forall a,k,l \in A \times \Omega \times \Delta : \quad -Mx_a^{kl} \leq T_a^{kl} \leq Mx_a^{kl}$$

$$\forall a,k,l \in A \times \Omega \times \Delta : \quad -M(1 - x_a^{kl}) \leq T_a^{kl} - T_a \leq M(1 - x_a^{kl})$$

$$\forall a,k,l \in A_1 \times \Omega \times \Delta : \quad x_a^{kl} \in \{0,1\}.$$

Note that M can be polynomially related to the problem data, using the same approach as the one described for the constant K in equation (3).

Even in this very restrictive setting of a single origin and a single destination, the next proposition shows that the toll setting problem is NP complete.

Theorem 1 *TOP is strongly NP-complete.*

Proof. Let us consider the decision problem defined by the oracle:

> Given a TOP instance and a rational profit level τ, does there exist a toll vector T such that Tx exceeds τ and (x, y) is an optimal lower-level reaction to T?

First note that TOP can be formulated as the penalized bilinear program BILIN1 with constraints disjoint in (x, y) and λ. One of its optimal solution must occur at a vertex of the primal-dual polyhedron. Since there is a representation of vertices which is polynomially related to the problem data, and K is polynomially related to the extremal solutions (see (3)), we have that TOP is in NP.

We show that TOP is NP-complete by a reduction from the Directed Hamiltonian Path problem (DHP, see Garey and Johnson [11]) to TOP. Let $G = (N, A)$ be a directed graph. We seek an Hamiltonian path between two distinguished vertices s and t of N and show that this can be achieved by finding a polynomially related toll setting problem. Consider the toll problem defined on the graph $\bar{G} = (N, \bar{A})$ where $\bar{A} = A \cup \{(s, t)\}$, all arcs of A are toll arcs and arc (s, t) (or its parallel copy) is an untolled arc. We set the initial costs to -1 on toll arcs and to $|N| - 1$ on the untolled arc (s, t). The lower bounds on tolls are set to 2. Demand from node s to node t is set to 1. We show that the existence of a toll strategy with revenue at least $\tau = 2(|N| - 1)$ implies the existence of an s-t Hamiltonian path in G. Let l denote the length of any s-t path. Since the maximum achievable profit compatible with this path is $|N| - 1 + l$, it is clearly optimal to set tolls onto the longest path of the network (with respect to the number of arcs), i.e. a Hamiltonian path, if such a path exists. The toll values on toll arcs that do not lie on the selected longest path are set to $|N| + 1$, assuring that any alternative path has a cost higher than that of the selected longest path, and hence that users do not have cheaper alternatives. □

Remark: If there exists an s-t Hamiltonian path $P \subseteq A$, then an optimal solution of the toll setting problem is obtained by setting $T_{ij} = 2$ if (i,j) is in P and $T_{ij} = |N|+1$ otherwise. The associated profit is equal to $2|N|-2$. Any policy that does not set tolls on the arcs of a Hamiltonian path results in a lower profit. Hence, the solution of TOP on the modified graph yields a Hamiltonian path on the original graph, if such a path exists.

As an example, the graph of Figure 3 possesses two Hamiltonian paths from 1 to 4: 1-2-3-4 and 1-3-2-4. An optimal solution, corresponding to the path 1-2-3-4, consists in setting $T_{1,2}$, $T_{2,3}$, $T_{3,4}$ to 2 and $T_{1,3}$, $T_{3,2}$, $T_{2,4}$ to 4. The maximum profit for this example is equal to 6.

Figure 3 goes here

3.2 The case of a single toll arc

An easily solvable instance occurs when there is but a single taxable arc, say a, by tailoring the parametric linear programming technique to this situation. We define $\gamma_{kl}(T_a)$ as the cost of a shortest path for the origin-destination couple (k,l), for a given value of T_a, and we set $\pi_{kl} = \gamma_{kl}(\infty) - \gamma_{kl}(0)$. Assume that the quantities π_{kl}'s are sorted in nonincreasing order:

$$\pi_{k_1,l_1} \geq \pi_{k_2,l_2} \geq \ldots \geq \pi_{k_t,l_t}$$

and let

$$i^* \in \arg\max_i \{\pi_{k_i,l_i} \sum_{j \leq i} d^{k_j l_j} : \pi_{k_i,l_i} \geq l_a\}.$$

An optimal toll is then directly obtained as

$$T_a = \pi_{k_{i^*},l_{i^*}}.$$

If π_{kl} is less than l_a for all origin-destination couples (k, l), the profit will be zero for all values of T_a larger than l_a.

3.3 The lower level path is known

The current and the next subsections are devoted to the single origin, single destination case. Without loss of generality, we assume that demand is equal to one. We denote by p_0 a shortest path from the origin s to the destination t obtained by setting all tolls to zero, and by p_∞ the shortest s-t path obtained by setting all tolls to $+\infty$. Furthermore, let π denote the difference between the cost of p_∞ and the cost of p_0. Clearly, π is an upper bound on the leader's profit, and one might expect that, corresponding to an optimal toll vector, the flow be assigned to path p_0. However, this need not be the case, as can be readily verified on the network of Figure 4, where an optimal solution is given by $T_{23} = T_{45} = T_{15} = 7$ and where flows are assigned to the path 1–5 rather than to the optimal path with respect to null tolls, i.e., $p_0 = 1–2–3–4–5$. Moreover, the upper bound $\pi = 12 - 4 = 8$ is not reached.

Figure 4 goes here

In this subsection, we develop an algorithm that determines optimal toll levels, given that the shortest path taken by the users is known a priori. In this case, optimal tolls can be obtained by solving for a shortest path in some modified graph G', which is defined in the statement of the next proposition.

Let p^* denote the selected shortest path. Without loss of generality, one may delete the toll arcs that are not part of path p^*; this is equivalent to setting their toll to an arbitrary large value, thus ensuring that the flow on these arcs is zero. A common value T_∞ suitable for all toll arcs not on p^* is given by

$$T_\infty = \gamma_{st}(\infty) - \gamma_{st}(0) + |N| \max\{0, -\min_{(i,j)\in A} c_{ij}\}. \tag{6}$$

If all costs c_{ij} are nonnegative, T_∞ can be simply set to the difference $\gamma_{st}(\infty) - \gamma_{st}(0)$ of the shortest path costs with infinite and zero costs, respectively.

Proposition 3 *Let $G = (N, A)$ be a graph without negative cycles with respect to the cost vector c and p^* be a simple s–t path. Define $G' = (N, A')$ as the graph where $a = (i, j)$ is in A' if either (i, j) is in A_2 or (j, i) lies on p^*. Set $c'_{ij} = c_{ij}$ if (i, j) is in A_2, $c'_{ij} = -c_{ji}$ if (j, i) is in $A_2 \cap p^*$ and $c'_{ij} = -c_{ji} - l_{ji}$ if (j, i) is in $A_1 \cap p^*$. Then optimal tax levels are given by $T^*_{ij} = \infty$ if (i, j) is in $A_1 - p^*$ and $T^*_{ij} = \lambda_j - \lambda_i - c_{ij}$ if (i, j) is in $A_1 \cap p^*$, where λ_k is the length of a shortest path from the origin node s to node k in the graph G'. If no shortest s–t path is detected, then the problem is infeasible with respect to the lower bound vector l.*

Proof. The single origin-destination problem is formulated as

$$\begin{array}{ll}
\max\limits_{T\geq l, x} & \sum\limits_{(i,j)\in A_1} T_{ij}x_{ij} \\
\min\limits_{x\in\Phi} & \sum\limits_{(i,j)\in A_1} (c_{ij} + T_{ij})x_{ij} + \sum\limits_{(i,j)\in A_2} c_{ij}x_{ij},
\end{array}$$

where Φ, the polyhedron of feasible flows, is defined by the flow conservation and nonnegativity constraints

$$\begin{array}{l}
\sum\limits_{a\in i^-} x_a - \sum\limits_{a\in i^+} x_a = \left\{\begin{array}{rl} -1 & i = s \\ +1 & i = t \\ 0 & \text{else} \end{array}\right. \qquad \forall i \in N \\
x_{ij} \geq 0 \quad \forall (i, j) \in N \times N.
\end{array}$$

As before, one replaces the lower level program by its optimality conditions. This yields the single-level mathematical program

$$
\begin{array}{lll}
\max_{x,T,\lambda} & \sum_{(i,j)\in A_1} T_{ij}x_{ij} & \\
\text{subject to} & \lambda_j - \lambda_i \leq c_{ij} + T_{ij} & (i,j) \in A_1 \\
 & \lambda_j - \lambda_i \leq c_{ij} & (i,j) \in A_2 \\
 & x_{ij}(c_{ij} + T_{ij} - \lambda_j + \lambda_i) = 0 & (i,j) \in A_1 \\
 & x_{ij}(c_{ij} - \lambda_j + \lambda_i) = 0 & (i,j) \in A_2 \\
 & T_{ij} \geq l_{ij} & (i,j) \in A_1 \\
 & x \in \Phi. &
\end{array} \tag{7}
$$

Since the lower level shortest path p^* is a priori determined, there exists an optimal solution x that satisfies

$$
x_{ij} = \begin{cases} 1 & \text{if } (i,j) \in p^* \\ 0 & \text{else.} \end{cases}
$$

This allows us to substitute $\lambda_j - \lambda_i - c_{ij}$ for T_{ij} in the objective function and to get rid of the complementarity slackness terms in (7). These simplifications lead to the more compact form

$$
\begin{array}{lll}
\max_{\lambda} & \sum_{(i,j)\in A_1\cap p^*} (\lambda_j - \lambda_i - c_{ij}) & \\
\text{subject to} & \lambda_j - \lambda_i \leq c_{ij} & (i,j) \in A_2 - p^* \\
 & \lambda_j - \lambda_i = c_{ij} & (i,j) \in A_2 \cap p^* \\
 & \lambda_i - \lambda_j \leq -c_{ij} - l_{ij} & (i,j) \in A_1 \cap p^*.
\end{array} \tag{8}
$$

The objective function of the above program can be expressed as

$$
\begin{aligned}
\sum_{(i,j)\in A_1\cap p^*} (\lambda_j - \lambda_i - c_{ij}) &= \sum_{(i,j)\in p^*} (\lambda_j - \lambda_i) - \sum_{(i,j)\in A_1\cap p^*} c_{ij} - \sum_{(i,j)\in A_2\cap p^*} (\lambda_j - \lambda_i) \\
&= \lambda_t - \lambda_s - \sum_{(i,j)\in A_1\cap p^*} c_{ij} - \sum_{(i,j)\in A_2\cap p^*} c_{ij} \\
&= \lambda_t - \lambda_s - \sum_{(i,j)\in p^*} c_{ij}
\end{aligned}
$$

and we can thus rewrite our optimization problem as

$$
\begin{array}{lll}
\max_{\lambda} & \lambda_t - \lambda_s & \\
\text{subject to} & \lambda_j - \lambda_i \leq c_{ij} & (i,j) \in A_2 - p^* \\
 & \lambda_j - \lambda_i \leq c_{ij} & (i,j) \in A_2 \cap p^* \\
 & \lambda_j - \lambda_i \leq -c_{ji} & (j,i) \in A_2 \cap p^* \\
 & \lambda_j - \lambda_i \leq -c_{ji} - l_{ji} & (j,i) \in A_1 \cap p^*.
\end{array}
$$

This program is nothing but the dual of a shortest s-t path problem defined on the graph G'. Its optimal solution yields optimal toll values $T^*_{ij} = \lambda_i - \lambda_j - c_{ij}$.

If no shortest path exists, i.e., there exists a negative cycle in G', then this implies that the program (6) is infeasible and we conclude that no vector of tolls can both satisfy the lower bound constraints and be compatible with p^* being a shortest s-t path. □

We illustrate the solution procedure on the example of Figure 5a, where $s = 1$, $t = 5$ and all lower bounds l_{ij} are set to $-\infty$. Return arcs of toll arcs are assigned an infinite cost and can therefore be ignored. The numbers next to the arcs correspond to the (initial) arc costs c_{ij}. An upper bound on the profit is given by $\pi = \gamma(\infty) - \gamma(0) = 22 - 6 = 16$. The path p^* is selected as {(1,2), (2,3), (3,4), (4,5)} and corresponds to p_0. The resulting graph G' is shown in Figure 5b. The shortest path from node 1 to node 5 in G' is {(1,2), (2,4), (4,3), (3,5)}. The node labels are: $\lambda_1 = 0$, $\lambda_2 = 2$, $\lambda_3 = 9$, $\lambda_4 = 11$, $\lambda_5 = 21$, and the associated optimal solution is

$$\begin{aligned} T_{23} &= \lambda_3 - \lambda_2 - c_{23} = 5 \\ T_{45} &= \lambda_5 - \lambda_4 - c_{45} = 10. \end{aligned}$$

Figure 5 goes here

Notice that the upper bound π defined earlier is not reached, although the current path and associated profit are optimal.

It is instructive to interpret the solution of this example within the framework of the linear bilevel program (5). In order to conform with the requirement that the initial cost of the toll arcs be zero, we first replace the toll arc (2,3) by an untolled arc $(2, 2')$ with cost 2 and a toll arc $(2', 3)$ with zero cost (see Figure 5). Next we reduce the flow on toll arcs by ϵ. Since flow conservation is no longer satisfied, we must reroute this flow along untolled arcs, at lowest cost. This is achieved by solving the transshipment problem illustrated in Figure 5c, where a supply of ϵ, indicated by $+\epsilon$ is issued at nodes $2'$ and 4, and a demand of ϵ, indicated by $-\epsilon$, is issued at nodes 3 and 5. An optimal solution to this problem is obtained by sending a flow of ϵ along paths $2'$–2–4–3 and 4–3–5, respectively. The added cost of this rerouting is equal to ϵ times $(-2+9-2)$ (cost of first path) plus $(-2+12)$ (cost of second path), i.e., 15ϵ, and the optimal profit 15 is obtained by dividing by ϵ. The dual variables, or node potentials, corresponding to this solution are

$$\lambda_1 = 0 \quad \lambda_2 = 2 \quad \lambda_{2'} = 4 \quad \lambda_3 = 9 \quad \lambda_4 = 11 \quad \lambda_5 = 21,$$

where the dual variable of node 1 has been arbitrarily set to zero. Using the notation of the TAX problem (see Proposition 2 for instance), the optimal tax levels are given by the expression

$$T = \lambda A - c = \lambda A,$$

since $c = 0$, i.e.,

$$T_{23} = T_{2'3} = \lambda_3 - \lambda_{2'} = 9 - 4 = 5 \quad T_{45} = \lambda_5 - \lambda_4 = 21 - 11 = 10.$$

This matches the total profit $15\epsilon/\epsilon$ obtained previously.

Of course, the above solution of the transshipment problem is only valid insofar as flows remain nonnegative. In the above example, this requirement is satisfied if and only if ϵ is less than or equal to $1/2$, which is consistant with our marginal analysis. Therefore, for this problem, the exact penalty factor K in BILIN1 can be set to $1/\epsilon = 2$. We conjecture that this upper bound holds for all toll-setting problems involving a single origin-destination pair.

Some remarks are in order:

1. The complexity of the solution procedure is $O(|N|^3)$, if one implements Ford-Bellman-Moore's algorithm for finding the shortest path from s to t. If there are no lower bound constraints, and p^* is composed solely of toll arcs, then there are no reverse arcs, and Dijkstra's algorithm can be implemented, to yield an $O(n^2)$ worst-case procedure.

2. It is easy to construct examples where the solution, contrary to this example, is not unique and (or) requires negative toll values. This is illustrated in Figure 6 where the lower bound values are set to $-\infty$. An optimal solution is given by

$$T_{12} = 4 + \alpha \quad T_{23} = -2 - \alpha \quad T_{3,4} = 4 + \alpha$$

is optimal for all nonnegative values of α.

Figure 6 goes here

3.4 The toll arcs with positive flows are known

We now solve a variant of the problem where the set of taxable arcs carrying positive flow is known a priori. Specifically we show how to determine a path passing through the selected toll arcs that is used by the lower level, i.e., a shortest path with respect to total arc costs. Once this path is identified, the problem reduces to the one resolved in subsection 3.3.

To this aim, let us partition the set of toll arcs as

$$A_1 = A_1' \cup A_1'',$$

where A_1' denotes the set of toll arcs carrying positive flow.

A lower level path is obtained by solving for the shortest $s-t$ path going through the imposed arcs. This problem is equivalent to the minimum cost flow problem

$$\begin{array}{lll} \min\limits_{x\in\Phi} & \sum\limits_{a\in A} c_a x_a & \\ \text{subject to} & x_a = 1 & a \in A'_1 \\ & x_a = 0 & a \in A''_1. \end{array}$$

The validity of this construction is based on the fact that the ranking of the paths (with respect to their lengths) going through the imposed toll arcs is not influenced by the actual values of the tolls. Hence the shortest such path can be determined without reference to the toll values, to be determined subsequently. Going back to the example illustrated in Figure 5a, we find that the shortest path (actually the unique path) going through the toll arcs (2,3) and (4,5) is p^*=1–2–3–4–5. In this case, the above minimum cost flow problem has a trivial solution. This need not be always the case.

4 A Numerical example

In this section we illustrate our model by means of the instance of a network toll setting problem shown in Figure 7. The problem involves four origin-destination pairs, 11 nodes and 25 arcs, seven of the latter being toll arcs (boldfaced arcs on the figure).

Two solution techniques have been tested. The first is based on the reformulation of TOP as a linear bilevel program, along the lines of Section 2. The resulting bilevel program has 44 upper level flow conservation constraints, 44 lower level flow conservation constraints, 100, i.e., 4×25, upper level variables, 72 ($4 \times (25-7)$) lower level variables, and opposite objectives. By today's standards this is a hard, medium-sized bilevel program, which we solved using a state-of-the-art algorithm of Hansen, Jaumard and Savard [10]. The difficulty of the problem is reflected in the large number of nodes of the implicit enumeration tree that had to be explored before reaching an optimal solution *and* getting a certificate of optimality. We observed that the algorithm was somewhat insensitive to the value of the penalty parameter K. Indeed, feasible solutions were reported for quite small values of K, and we could report no numerical instability problems. Whatever the foreseeable improvements in bilevel algorithms, we do not believe that this approach could address much larger instances of TOP.

The mixed integer reformulation, on the other hand, efficiently exploits the combinatorial structure of the problem, i.e., the fact that the lower level's extremal solution are paths and can be represented by binary variables. Using this approach, problems based on the network of Figure 7 could be solved in less than 1 second on a Sun Sparc 10, using GAMS–CPLEX 2.1. This leads us to believe that much larger instances could be routinely solved using this formulation.

Figure 7 goes here

Three scenarios were considered. The first is the base scenario. In the second scenario, demand is increased on origin-destination pair 1–3. In the third scenario, nonnegativity constraints on the tolls are incorporated into the model. The optimal solutions are shown in Tables 2, 3 and 4. For Scenario I, no profits are achieved for travellers going from origin 1 to destination 3. As the demand on that pair is increased from 12 to 20 in Scenario II, tolls, even negative ones, are introduced (see Table 3). When nonnegativity constraints on tolls are imposed (Scenario III), a new solution, with lower overall profit, is obtained (see Table 4). It is interesting to notice that flows are not always assigned to the shortest path with respect to null tolls. For instance, in Scenario I, the final path for the origin-destination pair 1–3 is 1–4–3 (with initial and final cost 20) while the initial shortest path is 1–10–9–11–3 (with initial cost 12).

Table 5 analyzes the effect of varying the parameter M. The fastest execution times were obtained for small values of M, as expected. The execution time is highly correlated to the value of the initial upper bound, which stabilizes very early, as M increases. Within the Branch-and-Bound scheme of CPLEX, the "Best First" searched strategy performed much better than the "Depth First" strategy. When M is too small, the solution becomes suboptimal, although the algorithm still yields a solution that is feasible with respect to lower level optimality. This need not be the case for the linear bilevel reformulation, where small values of the penalty parameter K might result in solutions that are "overoptimal" and nonoptimal for the lower level. If tolls are bounded from below, a valid estimate for M is the difference between the shortest path with infinite tolls and the shortest path with tolls set at their respective lower bounds.

Finally, in order to assess the potential of the mixed integer reformulation, we applied it to 12 larger instances of TOP. Specifically, the second test network consists in a 5×12 grid of bidirectional arcs (60 nodes, 206 arcs). Costs were generated according to independent uniform random variables over the interval [2,20]. The probability α that an arc be tollable was set to 0.05, 0.10, 0.15, and 0.20. Three instances were generated for each value of α.

OD	demand	path	path toll	path fixed cost	total cost	profit
1–3	12	1–4–3	0	20	240	0
1–2	8	1–10–2	11	12	184	88
2–4	16	2–6–5–8–4	6	14	320	96
2–3	10	2–11–3	14	160	300	140

TOTAL PROFIT: 324

arc:	(1,10)	(5,8)	(6,9)	(7,3)	(9,7)	(10,9)	(11,3)
toll:	11	6	4	16	0	0	14

Table 2: Scenario I: The optimal solution

OD	demand	path	path toll	path fixed cost	total cost	profit
1–3	20	1–10–9–7–3	7	13	400	140
1–2	8	1–10–2	−4	12	64	−32
2–4	16	2–6–5–8–4	6	14	320	96
2–3	10	2–6–9–7–3	15	15	300	150

TOTAL PROFIT: 354

arc:	(1,10)	(5,8)	(6,9)	(7,3)	(9,7)	(10,9)	(11,3)
toll:	−4	6	4	13	−2	0	14

Table 3: Scenario II: The optimal solution

The number of origin-destination pairs was fixed to 20. All problems with α less than or equal to 0.10 were solved to optimality with CPU times ranging from 155 to 3969 seconds. The number of explored nodes in the branch-and-bound tree varied from 106 to 2114. For α 0.15 or 0.20, an optimal solution was not reached for two out of six instances within 24 CPU hours. This shows the limitations of the approach: it solves medium-sized instances.

5 Conclusion and extensions

The model introduced in this paper constitutes a first attempt at understanding, both from the theoretical and algorithmical points of view, an important class of decision problems. At this step, it is clear that we had no intention of incorporating into our model all the features of a real-life application. In particular, our representation of user behaviour is overly simplistic

OD	demand	path	path toll	path fixed cost	total cost	profit
1–3	20	1–10–9–11–3	8	12	400	160
1–2	8	1–10–2	0	12	96	0
2–4	16	2–6–5–8–4	6	14	320	96
2–3	10	2–11–3	8	16	240	80

TOTAL PROFIT: 336

arc:	(1,10)	(5,8)	(6,9)	(7,3)	(9,7)	(10,9)	(11,3)
toll:	0	6	4	7	0	0	8

Table 4: Scenario III: The optimal solution

strategy	M	nodes	cpu	pivots	upper bound
BFS	15	21	.90	240	416.6
DFS		24	.97	264	
BFS	25	77	2.88	671	437.0
DFS		124	4.18	680	
BFS	35	53	1.89	443	437.8
DFS		180	5.70	755	
BFS	45	74	2.31	471	438.3
DFS		260	7.79	991	
BFS	55	85	2.66	532	438.6
DFS		242	7.00	923	
BFS	95	83	2.72	553	439.2
DFS		454	12.92	1277	
BFS	500	89	2.79	481	439.9
DFS		262	7.76	741	

Table 5: Scenario I: Varying parameter M.

in that it assumes no dispersion of traffic along the routes of the network. A dispersion effect could be achieved simply by incorporating volume-delay curves. A more irksome assumption of our toll model is that the value-of-time parameter is uniform throughout the user population, and that, given the choice between two paths of equal costs, the users always select the one with the highest toll. Both these drawbacks could be eliminated by introducing a range of value-of-time parameters across users. For instance (see Leurent [13]), let α denote the value-of-time parameter associated with a cross-section α of the user population, and assume that α is distributed according to a function $H(\alpha)$ whose derivative $h(\alpha)$ is the associated density function, i.e.:

$$H'(\alpha) = h(\alpha) \qquad h(\alpha) \geq 0 \qquad \int_0^\infty h(\alpha)\, d\alpha = 1.$$

The lower level assignment now takes the form of the control problem

$$\begin{array}{rl}
\min\limits_{x(\cdot)} & \sum\limits_{a\in A_1} c_a x_a + \sum\limits_{a\in A_2} d_a x_a + \sum\limits_{a\in A_1} T_a \int_0^\infty \alpha x_a(\alpha) \\
\text{subject to} & \sum\limits_{a\in i^+} x_a^{kl}(\alpha) - \sum\limits_{a\in i^-} x_a^{kl}(\alpha) = \begin{cases} h(\alpha)d^{kl} & \text{if } i = k \\ -h(\alpha)d^{kl} & \text{if } i = l \\ 0 & \text{else} \end{cases} \\
\forall a \in A: & x_a(\alpha) = \sum\limits_{k,l} x_a^{kl}(\alpha) \\
\forall a \in A: & x_a = \int_0^\infty x_a(\alpha)\, d\alpha \\
\forall a, k, l \in A \times \Omega \times \Delta: & x_a^{kl}(\alpha) \geq 0.
\end{array}$$

While this infinite-dimensional linear program is more complex than the shortest path problem it replaces, not everything is gloomy. Indeed, as indicated in Marcotte and Zhu [18], an efficient solution to this problem can be obtained by solving a parametric shortest path problem with respect to the parameter α. The general taxation problem TAX could be enhanced in a similar fashion. The introduction of infinitely many nonatomic customer classes has a smoothing effect on the model, in the sense that the lower level problem is likely to have, for a given tax vector T, a unique and continuously differentiable solution $x(T)$. This opens up the possibility of addressing the toll setting problem as a smooth program, allowing the implementation of nonlinear programming algorithms, as proposed in [14]. Note that this does not make the problem convex, i.e., we stay in the realm of global optimization. However, this could facilitate the search for 'good' local solutions.

References

[1] Aiyoshi, E. and Shimizu, K., "A solution method for the static constrained Stackelberg problem via penalty methods", *IEEE Transactions on Automatic Control* **29** (1984) 1111–1114.

[2] Anandalingam, and White, D. J., "A solution method for the linear static Stackelberg problem using penalty functions", *IEEE Transactions on Automatic Control* **35** (1990) 1170–1173.

[3] Bard, J., "An investigation of the linear three-level programming problem", *IEEE Transactions on Systems, Man and Cybernetics* **14** (1984) 711–717.

[4] Bard, J.F. and Moore J.T., "A branch and bound algorithm for the bilevel linear programming problem," *SIAM Journal on Scientific and Statistical Computing* **11** (1990) 281–292.

[5] Ben-Ayed O., "Bilevel linear programming," *Computers and Operations Research* **20** (1993) 485–501.

[6] Bialas, W. and Karwan, M., "Two-level linear programming", *Management Science* **30** (1984) 1004–1020.

[7] Júdice J.J. and Faustino A.M, "The solution of the linear bilevel programming problem by using the linear complementary problem," *Investigação Operacional* **8** (1988) 77–95.

[8] Fukushima, M., "Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems", *Mathematical Programming* **53** (1992) 99–110.

[9] Hobbs, B. and Nelson, S., "A nonlinear bilevel model for analysis of electric utility demand-side planning issues", *Annals of Operations Research* **34** (1992) 255–274.

[10] Hansen, P., Jaumard, B. and Savard, G., "New branch-and-bound rules for linear bilevel programming", *SIAM Journal on Scientific and Statistical Computing* **13** (1992) 1194–1217.

[11] Garey, M.R. and Johnson, D.S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman, New York (1979).

[12] Jeroslow, R. G., "The polynomial hierarchy and a simple model for competitive analysis" *Mathematical Programming* **32** (1985) 146–164.

[13] Leurent, F., *Cost versus time equilibrium over a network*, European Journal of Operational Research 71 (1993) 205–221.

[14] Luo, Z.Q., Pang, J.S. and Ralph, D., *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, Cambridge (1996).

[15] Loridan, P. and Morgan, J., "ϵ-regularized two-level optimization problems: approximation and existence results", *Fifth French-German Conference on Optimization*, Lecture Notes in Mathematics 1405, Springer-Verlag (1989) 99–113.

[16] Marcotte, P., "Network design problem with congestion effects: A case of bilevel programming", *Mathematical Programming* **34** (1986) 142–162.

[17] Marcotte, P. and Zhu, D., "Exact and inexact penalty methods for the generalized bilevel programming problem", *Mathematical Programming* **74** (1996) 141–157.

[18] Marcotte, P. and Zhu, D., "Equilibria with infinitely many differentiated classes of customers", in *Complementarity and variational problems. State of the art*, M.C. Ferris and J.S. Pang eds., SIAM, Philadelphia (1997).

[19] Shimizu, K., Ishizuka, Y, and Bard, J.F., "Nondifferentiable and Two-Level Mathematical Programming", Kluwer, Boston (1997).

[20] Vaish, H. and Shetty C.M., "The bilinear programming problem," *Naval Research Logistics Quarterly* **23** (1976) 303–309.

[21] van Ackere, A., "The principal/agent paradigm: its relevance to various functional fields", *EJOR* **70** (1993) 83–103.

[22] Vicente, L. N. and Calamai, P. H., "Bilevel and multilevel programming: a bibliography review", *Journal of Global Optimization* **5** (1994) 291–306.

[23] White, D. J. and Anandalingam, "A penalty function approach for solving bilevel linear programs", *Journal of Global Optimization* **3** (1993) 397–419.

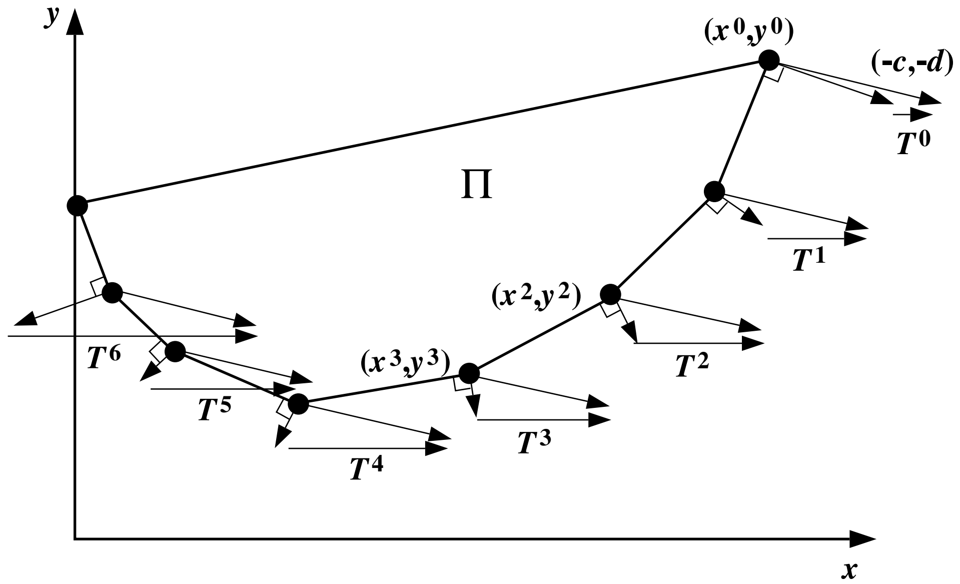

Figure 1: A two-dimensional example

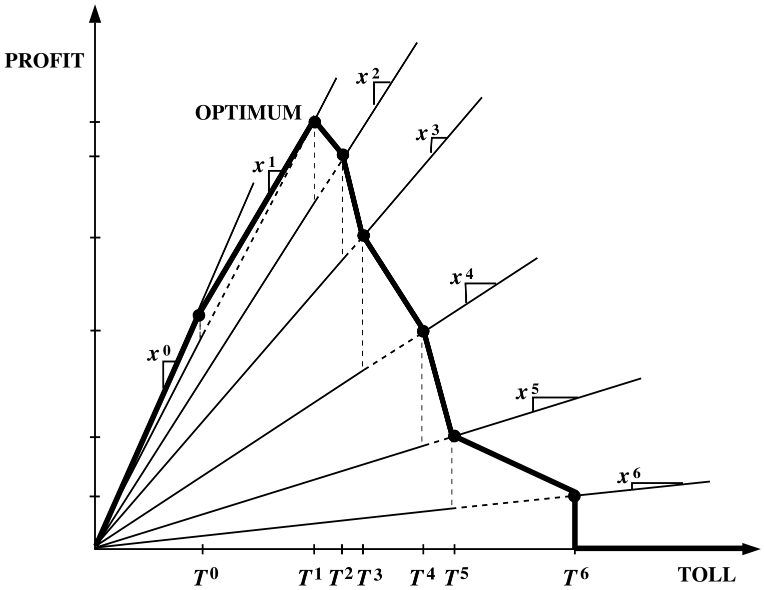

Figure 2: Upper envelope of the function $F(x(T), y(T), T)$

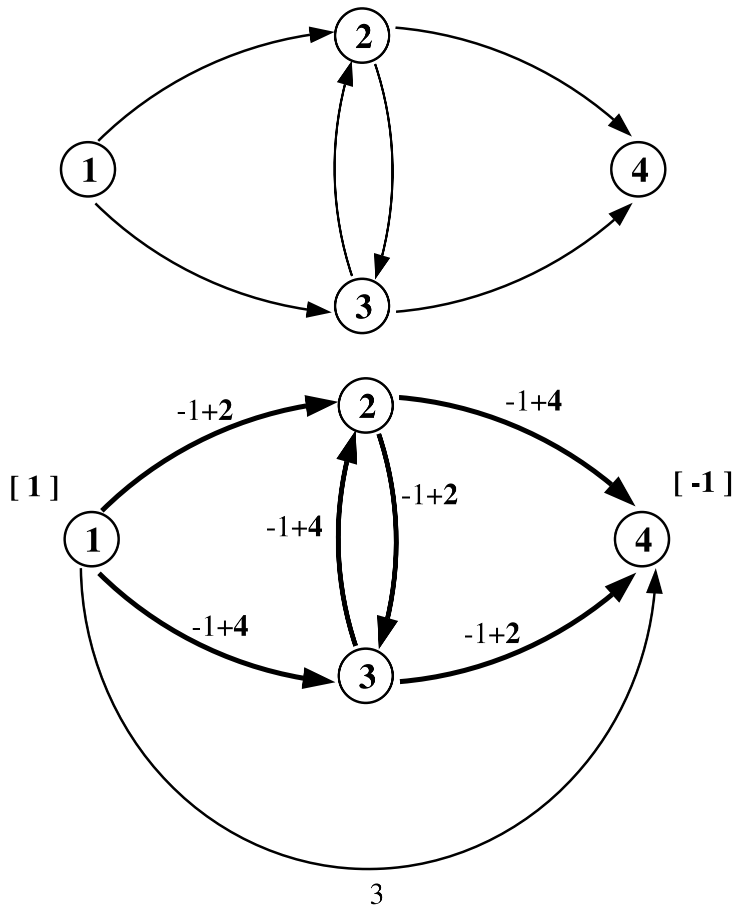

Figure 3: The graph transformation

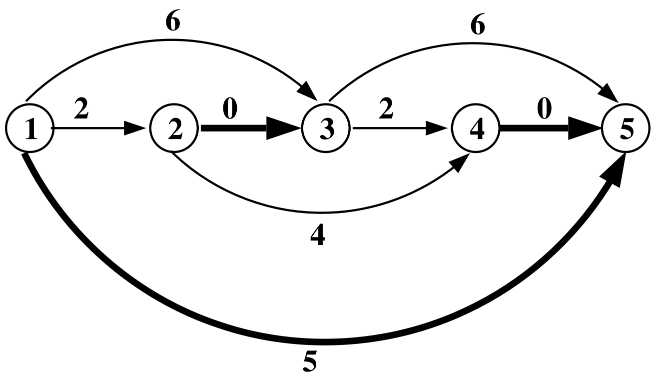

Figure 4: The upper bound is not reached.

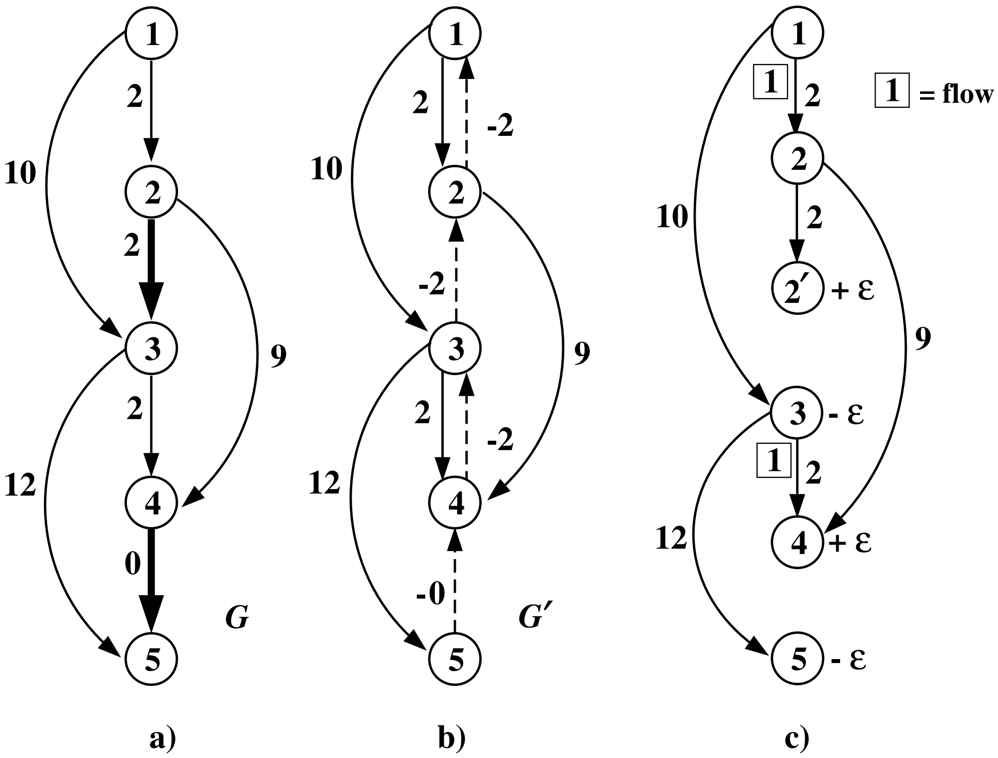

Figure 5: The input graph G and the modified graph G'

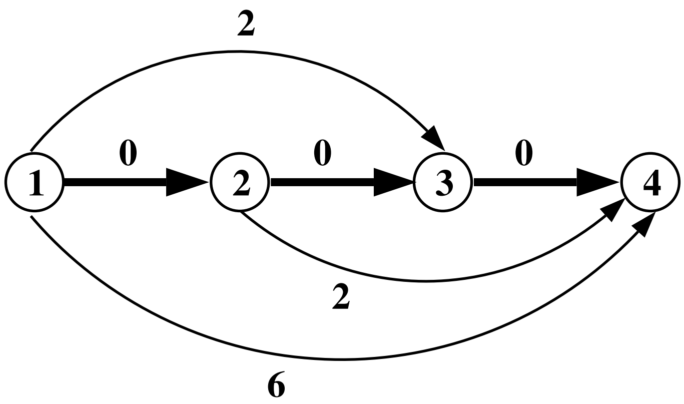

Figure 6: An optimal solution with a negative toll value.

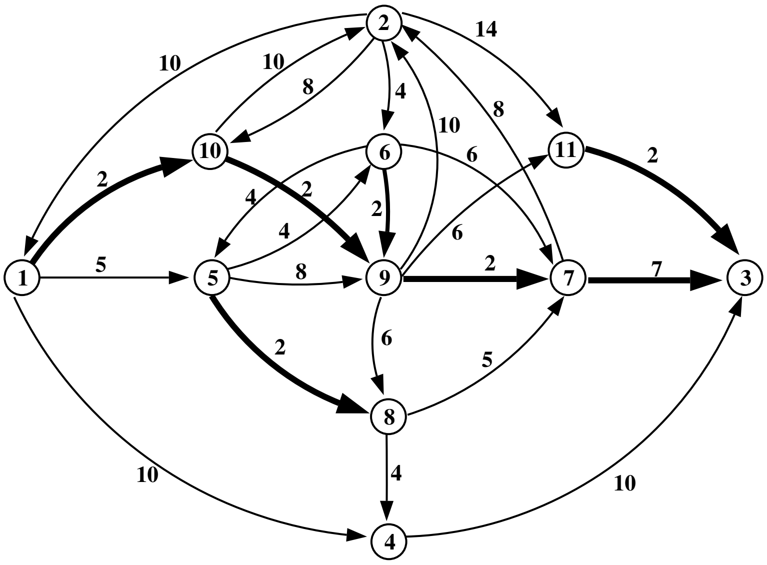

Figure 7: Network used for the numerical tests