

# Joint Design and Pricing on a Network

**Luce Brotcorne**

LAMIH/ROI, Université de Valenciennes, 59313 Valenciennes Cedex 9, France,  
luce.brotcorne@univ-valenciennes.fr

**Martine Labbé**

SMG and ISRO, Université Libre de Bruxelles, 1050 Brussels, Belgium, mlabbe@ulb.ac.be

**Patrice Marcotte**

CIRRELT and Département d'Informatique et de Recherche Opérationnelle, Université de Montréal,  
Montreal, Quebec, Canada H3C 3J7, marcotte@iro.umontreal.ca

**Gilles Savard**

GERAD and Département de Mathématiques et de Génie Industriel, Ecole Polytechnique de Montréal,  
Montreal, Quebec, Canada H3C 3A7, gilles.savard@polymti.ca

To optimize revenue, service firms must integrate within their pricing policies the rational reaction of customers to their price schedules. In the airline or telecommunication industry, this process is all the more complex due to interactions resulting from the structure of the supply network. In this paper, we consider a streamlined version of this situation where a firm's decision variables involve both prices and investments. We model this situation as a joint design and pricing problem that we formulate as a mixed-integer bilevel program, and whose properties are investigated. In particular, we take advantage of a feature of the model that allows the development of an algorithmic framework based on Lagrangean relaxation. This approach is entirely novel, and numerical results show that it is capable of solving problems of significant sizes.

*Subject classifications:* economics: pricing; games; integer programming; heuristics; transportation; programming: nonlinear, nondifferentiable.

*Area of review:* Transportation.

*History:* Received June 2004; revisions received December 2005, August 2006; accepted September 2006.

## 1. Introduction

This paper is devoted to a model that captures the interaction between system design, price setting, and consumer choice over a transportation network, without assuming the a priori knowledge of demand functions. The problem involves two decision makers acting noncooperatively and in a sequential way. The upper level (leader) strives to maximize its revenue raised from tariffs imposed on a set of goods or services in its control, whereas the lower level (follower) optimizes its own objective, taking into account the tariff schedule set by the leader. The leader explicitly incorporates the reaction of the follower in his optimization process. In the field of economics, this fits the principal/agent paradigm (Van Ackere 1993) 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 with bilinear objectives at both levels of decision.

In the current context of deregulation, pricing decisions have become crucial for airline, trucking, telecommunication, and service industries where intense price competition and network modifications have occurred. Clearly,

a profit-maximizing firm must consider the trade-off between the cost of service and the revenue generated when designing its system and prices.

In the passenger or freight airline industry, a carrier (the leader) selects routing patterns, flight schedules, and fares. For instance, Budenbender et al. (2000) describe a system where freight providers such as express shipment companies operate or rent an aircraft fleet that must provide a high level of service. For consolidation purposes, the freight is first shipped to an airport. Next, it is flown nonstop to another airport, finally to be loaded on trucks and shipped to its final destination. The problem then consists of determining the terminal to operate, the take-off time, how to transport the freight to an airport, and the rate to charge. In passenger transportation, the introduction of new flights (direct or through a hub-and-spoke network) must take into account the supply over the entire network of flights, both from the leader airline and its competitors. The decisions are then taken with respect to the incurred costs, the quality of service, the possible influence on demand to other destinations and, most important, the revenues generated by the new services (Lederer 1993, Lederer and Nambimadon 1998).

In the surface freight transportation industry, important structural changes occur as shippers optimize the end-to-end supply through the implementation of Web-based portals. In that context, the costs incurred by a carrier is made up of two components: a fixed cost (including trade compliance, trade settlement with country-specific international trading portals, multimodal aspects, operating resources costs, global handling costs, etc.) and a unit transportation cost (Kerr 2001). Upon reception, a service carrier (the leader) has to decide whether or not to accept a request and, if accepted, to set a price. In reaction to those prices, the shippers (the follower) want their goods to be transhipped at minimal cost, hence the bilevel structure of the problem.

In the telecommunication area, a service provider (the leader) has to make network deployment decisions and to set prices for bandwidth usage. The response of users (the follower) to prices induces traffic on the network. In the current deregulated markets, pricing is a fundamental issue for communications carriers. Indeed, as new systems of ever-larger capacities are introduced, the marginal cost of data transmission is rapidly decreasing. Exploiting those cost savings and handling increased demand involves the optimization of technology acquisition and pricing processes (Lanning et al. 2000, Başar and Srikant 2002). A recent paper by Bienstock et al. (2006) addresses this issue within a dynamic framework, assuming a demand model involving constant elasticities and null cross elasticities. Several references to network design problems can be found therein, and also in the classic book by Ahuja et al. (1993).

Design and pricing are also challenging issues for business information service providers (Bashyam 2000). Information agencies such as Reuters and Bloomberg (foreign currency markets) and Aspect Development (component information services) are essentially intermediaries between firms that generate, and firms that use, content. Because information service providers (the leader) incur large fixed costs (data entry and updates, software development, database management systems, connections to commercial networks), their problem consists of specifying the size of the database they provide to subscribers (followers) as well as the price they will charge for subscriptions. At the lower level, the subscribers adapt their usage volume according to the level of service and tariffs of the service providers, or may select the self-service option whereby they collect and collate information directly from the sources.

Until now, design and pricing issues have mostly been treated separately. However, they are intrinsically linked and have to be addressed jointly. To our knowledge, the only papers addressing the joint design and pricing problem are those of Lederer (1993), Başar and Srikant (2002), and Bashyam (2000). Lederer (1993) proposes a Nash equilibrium model of air transport competition where firms select routes and prices. Competition is studied under two

different assumptions about consumer choice: Either consumers can spread their choice route using links belonging to different firms (“bundling” in the sense of Lederer), or they cannot. If bundling is forbidden, the author proves the existence of unique equilibrium prices. Otherwise, a price equilibrium may fail to be unique, or even to exist. At first glance, our work might seem to fit the framework analyzed by Lederer. However it differs in two main respects: bundling is an essential part of our model, and we look for a Stackelberg (leader-follower) equilibrium rather than a Nash equilibrium. Consequently, the focus of this paper is on algorithmic development rather than on economic considerations.

Başar and Srikant (2002) study the economics of providing large capacity from a telecommunication provider’s point of view. Design choices are not modelled using binary decisions, but through continuous-capacity variables. Each user is charged a fixed price per unit of bandwidth used, and this price is independent from congestion. The transmission rate of each user is assumed to be a function of network congestion and price per unit of bandwidth. The aim of the service provider is to maximize its revenue. The authors show that, as the number of users increases, the optimal price per unit of bandwidth charged by the service provider may increase or decrease depending upon the bandwidth of the link. However, for all values of the link capacity, the overall performance of each user improves and the service provider’s revenue per unit of bandwidth increases, thus providing an incentive for the service provider to increase the available bandwidth in proportion to traffic. Although this work provides some theoretical insight into the problem, no computational procedure is described for its solution.

Bashyam (2000) analyzes service design and pricing of business information services in a competitive environment, using game-theoretic concepts. The problem consists of determining the optimal size of the database, as well as the subscription price they will fix for subscriptions, taking into account the reaction of subscribers who want to minimize their cost. They consider two types of interactions: monopoly or duopoly, and two types of information delivery technologies: online service that allows subscribers to access information over online networks, and package service that delivers information using physical media such as CD-ROMs. Their analytical approach investigates the differences in price structure associated with the type of provided services. In the case of duopoly, they also analyze the class of consumers (high- or low-volume consumers) served depending on the size of the database and on prices.

In this paper, we focus on a joint design and pricing problem motivated by a telecommunication application. The problem involves two decision levels interacting non-cooperatively and in a sequential way: The upper level is concerned with maximizing profits raised from tariffs set on a subset of arcs, while users react by sending flow on shortest paths joining their respective origins and destinations.

The specificity of the problem lies in the simultaneous determination of the connections to be opened and the tariff schedule to be applied. In this respect, it differs from previous works (Brotcorne et al. 2000, 2001; Grigoriev et al. 2005; van Hoesel et al. 2003), where only the tariff issue was considered.

The outline of this paper is as follows. In §2, we introduce a mixed-integer bilinear formulation for the joint design and pricing problem and discuss its properties. In §3, we prove a result that allows for the reduction of the size of the problem. In §4, we describe a solution approach based on the novel application of Lagrangean relaxation within the bilevel programming framework. Section 5 is concerned with numerical experiments on both randomly generated and real instances. Section 6 contains concluding remarks.

## 2. A Joint Design and Pricing Model

Let us consider a network defined over the underlying graph  $G = (\mathcal{N}, \mathcal{A})$ , with node set  $\mathcal{N}$  and arc set  $\mathcal{A}$ . A node represents either a supply site, a demand site, or the endpoints of an arc on which goods are carried. The set of arcs is partitioned into two subsets  $\mathcal{A}_1$  and  $\mathcal{A}_2$ , where  $\mathcal{A}_1$  denotes the set of links operated by the leader and  $\mathcal{A}_2$  the set of links operated by its competitors. With each arc  $a \in \mathcal{A}_1$ , we associate a tariff  $T_a$ , to be determined by the leader, a fixed opening cost  $f_a$ , and an operating cost  $c_a$  charged to the leader. Arcs in  $\mathcal{A}_2$  are tariff-free and bear only a unit cost  $d_a$ , which is outside the control of the leader. Demand is modelled by a set  $\mathcal{K}$  of commodities. These may represent distinct physical goods or identical physical goods associated with different points of origin and destination. Each commodity is associated with an origin-destination (OD) pair  $(p(k), q(k))$ . The demand vector  $b^k$  corresponding to commodity  $k$  is specified by

$$b_i^k = \begin{cases} n^k & \text{if } i = p(k), \\ -n^k & \text{if } i = q(k), \\ 0 & \text{otherwise,} \end{cases}$$

where  $n^k$  represents the amount of flow of commodity  $k$  to be shipped from  $p(k)$  to  $q(k)$ . The variable  $x_a^k$  (respectively,  $y_a^k$ ) denotes the flow of commodity  $k$  on arc  $a \in \mathcal{A}_1$  (respectively,  $a \in \mathcal{A}_2$ ). The binary variable  $v_a$ , associated with each arc  $a \in \mathcal{A}_1$ , indicates whether ( $v_a = 1$ ) or not ( $v_a = 0$ ) arc  $a$  is part of the network design.

The leader's variables are either discrete (design variables) or real valued (tariffs). Lower-level variables, i.e., flows, are real valued. In this nonatomic context, "bundling" occurs, and users maximize their utility on individual terms. Based on the above notation, the joint design and pricing problem can then be formulated as a mixed-integer bilevel program with bilinear objectives and linear constraints. The vector  $(x, y)$  of all link-commodity flows

corresponds to an optimal solution of the lower-level linear program parameterized by the upper-level tariffs  $T$ , which is solved on the subnetwork resulting from the binary variables  $v$ . Letting  $(A, B)$  denote the node-arc incidence matrix of the network, this leads to the bilevel programming formulation

$$\begin{aligned} \text{(JDP)} \quad & \max_{T, v, x, y} \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}_1} T_a x_a^k - \sum_{a \in \mathcal{A}_1} f_a v_a - \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}_1} c_a x_a^k \\ & \text{s.t. } v_a \in \{0, 1\} \quad \forall a \in \mathcal{A}_1, \end{aligned} \quad (1)$$

where  $(x, y)$  is an optimal solution of

$$\min_{x, y} \sum_{k \in \mathcal{K}} \left( \sum_{a \in \mathcal{A}_1} T_a x_a^k + \sum_{a \in \mathcal{A}_2} d_a y_a^k \right) \quad (3)$$

$$\text{s.t. } Ax^k + By^k = b^k \quad \forall k \in \mathcal{K}, \quad (4)$$

$$x_a^k \leq n^k v_a \quad \forall k \in \mathcal{K} \quad \forall a \in \mathcal{A}_1, \quad (5)$$

$$x^k, y^k \geq 0 \quad \forall k \in \mathcal{K}.$$

The upper-level objective (1) is to maximize total net revenue and is expressed as the difference between the sum of revenues arising from tariffs  $T_a$  and the sum of fixed opening costs and operating costs. The objective of the lower-level problem (7) is to minimize the total cost of the paths selected by network users. Constraints (8) represent the flow balance equations. Constraints (9) state that arcs can only be used if they are open.

For a given design vector  $v$ , let us perform the change of variable  $T' = T - c$ , which is tantamount to setting the usage cost of every tariff link  $a \in \mathcal{A}_1$  to  $c_a$  at the lower level. The resulting model coincides with that of Labbé et al. (1998), i.e.,

$$\text{(LMS)} \quad \max_{T', x, y} \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}_1} T'_a x_a^k, \quad (6)$$

where  $(x, y)$  is an optimal solution of

$$\min_{x, y} \sum_{k \in \mathcal{K}} \left( \sum_{a \in \mathcal{A}_1} (c_a + T'_a) x_a^k + \sum_{a \in \mathcal{A}_2} d_a y_a^k \right) \quad (7)$$

$$\text{s.t. } Ax^k + By^k = b^k \quad \forall k \in \mathcal{K}, \quad (8)$$

$$x_a^k \leq n^k v_a \quad \forall k \in \mathcal{K} \quad \forall a \in \mathcal{A}_1, \quad (9)$$

$$x^k, y^k \geq 0 \quad \forall k \in \mathcal{K}.$$

As in Labbé et al. (1998), we also assume that

- there do not exist tariffs that generate positive revenues and simultaneously create a negative cost cycle in the network (no "free lunch"),
- there exists at least one path composed of tariff-free arcs for each origin-destination pair (customers have tariff-free alternatives).

These assumptions imply that in the lower-level problem, flow is assigned to shortest (acyclic) paths linking

the customers' respective origins and destinations, on the subnetwork composed of tariff-free and tariff arcs that are available. Whenever ties occur among shortest paths, the paths selected are the ones yielding the highest profit for the leader. Note that a tariff schedule that induces unique paths and yields a revenue arbitrarily close to the optimal value can be achieved through a perturbation scheme. The next proposition provides an upper bound on the objective value of JDP, which can be evaluated by solving, respectively, shortest-path problems and a classical network design problem.

**PROPOSITION 1.** *A valid upper bound on the leader's profit is given by the difference between the follower's optimal objective corresponding to infinite tariffs (i.e., tariff arcs are unavailable to the follower) and the optimum value of the network design problem obtained by setting every tariff  $T_a$  at  $c_a$ .*

**PROOF.** For LMS, Labbé et al. (1998) derived the valid upper bound  $U_\infty - U_0$ , where  $U_\infty$  denotes the cost of a lower-level solution when access to tariff arcs is denied (infinite tariffs) and  $U_0$  is the cost of a shortest-path solution under null tariffs, i.e.,  $T' = 0$  or, equivalently, for fixed-design vector  $v$ ,  $T = c$  in JDP. It follows that a valid bound for the value of an optimal solution to JDP is given by

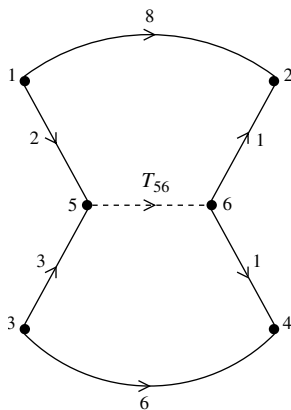
$$\max_v \left\{ U_\infty - U_0 - \sum_{a \in \mathcal{A}_1} f_a v_a \right\} = U_\infty - \min_v \left\{ U_0 + \sum_{a \in \mathcal{A}_1} f_a v_a \right\}, \quad (10)$$

where the term subtracted from  $U_\infty$  on the right-hand side of Equation (10) is precisely the optimal value of the network design problem obtained by setting the tariff vector  $T'$  at zero, i.e.,  $T = c$ , as claimed.  $\square$

### An Example

Note that the upper bound provided above need not be achieved. Indeed, consider the example of Figure 1, where

**Figure 1.** Upper bound on the profit not reached at the optimal solution.



demand is set to 2 on origin-destination pair 1-2 and to 4 on pair 3-4, whereas (5, 6) is the sole tariff link. If the opening and operating costs for the leader are set, respectively, to 1 and 0, the optimal solution is obtained by setting  $T_{5,6}$  to 2. This yields a profit of 11 units, whereas the upper bound provided is equal to  $40 - 23 = 17$ .

Now, taking into account that the entire demand associated with a given OD pair can be assigned to a single shortest path, we may, without loss of generality, reformulate JDP as

$$\begin{aligned} \max_{T, v, x, y} \quad & \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k T_a x_a^k - \sum_{a \in \mathcal{A}_1} f_a v_a - \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k c_a x_a^k \\ \text{s.t.} \quad & v_a \in \{0, 1\} \quad \forall a \in \mathcal{A}_1, \end{aligned}$$

where  $(x, y)$  is an optimal solution of

$$\begin{aligned} \min_{x, y} \quad & \sum_{k \in \mathcal{K}} n^k \left( \sum_{a \in \mathcal{A}_1} T_a x_a^k + \sum_{a \in \mathcal{A}_2} d_a y_a^k \right) \\ \text{s.t.} \quad & Ax^k + By^k = e^k \quad \forall k \in \mathcal{K}, \\ & x_a^k \leq v_a \quad \forall a \in \mathcal{A}_1 \quad \forall k \in \mathcal{K}, \\ & x^k, y^k \geq 0 \quad \forall k \in \mathcal{K}, \end{aligned} \quad (11)$$

and

$$e_i^k = \begin{cases} 1 & \text{if } i = p(k), \\ -1 & \text{if } i = q(k), \\ 0 & \text{otherwise.} \end{cases}$$

For fixed-design vector  $v$ , the resulting problem reduces to a multicommodity toll optimization problem that can be reformulated as a mixed-integer program (see Labbé et al. 1998). This formulation readily extends to an MIP formulation for JDP through incorporation of the design variables  $v_a$ . It also allows us to solve small instances by commercial software, and may serve as a testbed to judge the performance, in terms of speed and quality of solution, of specialized algorithms.

We close this section by mentioning that in the case where there is only one OD pair, JDP reduces to the toll optimization problem analyzed by Brotcorne et al. (2000, 2001). Indeed, the binary flow variables  $x_a$  can then replace the design variables  $v_a$ , and the problem formulation becomes

$$\begin{aligned} \max_{T, x, y} \quad & \sum_{a \in \mathcal{A}_1} n T_a x_a - \sum_{a \in \mathcal{A}_1} f_a x_a - \sum_{a \in \mathcal{A}_1} n c_a x_a \\ \text{s.t.} \quad & x_a \in \{0, 1\} \quad \forall a \in \mathcal{A}_1, \end{aligned}$$

where  $(x, y)$  is an optimal solution of

$$\begin{aligned} \min_{x, y} \quad & n \left( \sum_{a \in \mathcal{A}_1} T_a x_a + \sum_{a \in \mathcal{A}_2} d_a y_a \right) \\ \text{s.t.} \quad & Ax + By = e, \\ & x, y \geq 0. \end{aligned}$$

After performing the change of variables  $\tilde{T}_a = T_a - (1/n)f_a - c_a$ , the toll optimization problem becomes

$$\max_{\tilde{T}, x, y} \sum_{a \in \mathcal{A}_1} n\tilde{T}_a x_a,$$

where  $(x, y)$  is an optimal solution of

$$\begin{aligned} \min_{x, y} \quad & \sum_{a \in \mathcal{A}_1} (\tilde{T}_a + c_a + f_a/n)x_a + \sum_{a \in \mathcal{A}_2} d_a y_a \\ \text{s.t.} \quad & Ax + By = e, \\ & x, y \geq 0. \end{aligned}$$

The dropping of the flow integrality constraints at the upper level is then justified by the fact that the lower-level constraints are totally unimodular, and that it is not in the leader’s interest to induce fractional flows.

### 3. Moving Constraints to the Upper Level

For general bilevel programs, constraints involving both upper- and lower-level variables cannot be moved freely from one level to the other without altering both the feasible set and the optimal solution of the bilevel program. For one, upper-level constraints are transparent to the follower, and can only be induced through a proper choice of the leader’s tariffs. On the other hand, lower-level constraints must be satisfied by the follower. Actually, even in the simple case of linear bilevel programming, the feasible set corresponding to joint upper-level constraints may be disconnected. This explains why the presence of such constraints is perceived as a nuisance from the algorithmic point of view. However, in the case of JPD, the opposite might be true. Indeed, if the capacity constraints are moved upwards, the resulting *restricted* problem assumes the simple structure of a many-to-many shortest-path problem. It is a remarkable feature of JDP, and the wider class of bilinear bilevel programs to which it belongs as well, that one can perform this operation without affecting the optimal solution. This property will be central to the development of an efficient solution algorithm.

**PROPOSITION 2.** *Assume that P1 admits an optimal solution and that the matrix G has nonnegative entries. Then the sets of optimal solutions of the mathematical programs P1 and P2, displayed below, are nonempty and coincide:*

$$\begin{aligned} \text{(P1)} \quad & \max_{T, x, y} Tx - cx & \text{(P2)} \quad & \max_{T, x, y} Tx - cx \\ & \text{where } (x, y) \text{ solves} & & Gx \leq b^2, \\ & \min_{x, y} Tx + dy & & \text{where } (x, y) \text{ solves} \\ & \text{s.t. } Ex + Fy = b^1, & & \min_{x, y} Tx + dy \\ & Gx \leq b^2, & & \text{s.t. } Ex + Fy = b^1, \\ & x, y \geq 0, & & x, y \geq 0. \end{aligned}$$

**PROOF.** Under the assumptions, P1 is feasible and its objective function is bounded from above. It follows that one may replace the lower-level problem of P1 by its primal-dual optimality conditions. Because P2 is a *restriction* of P1 (the constraint  $Gx \leq b^2$  must be induced by the leader without being binding for the follower), the optimality conditions must also hold for P2. This allows the introduction of problems P1’ and P2’, respectively, equivalent to P1 and P2:

$$\begin{aligned} \text{(P1')} \quad & \max_{T, x, y, \lambda, \delta} Tx - cx \\ & \text{s.t. } Ex + Fy = b^1, \\ & Gx \leq b^2, \\ & \lambda E + \delta G \leq T - c, \\ & \lambda F \leq d, \\ & (d - \lambda F)y = 0, \\ & (T - c - \lambda E - \delta G)x = 0, \\ & \delta(b^2 - Gx) = 0, \\ & x, y \geq 0, \\ & \delta \leq 0. \end{aligned}$$

$$\begin{aligned} \text{(P2')} \quad & \max_{T, x, y, \lambda} Tx - cx \\ & \text{s.t. } Ex + Fy = b, \\ & Gx \leq b^2, \\ & \lambda E \leq T - c, \\ & \lambda F \leq d, \\ & (d - \lambda F)y = 0, \\ & (T - c - \lambda E)x = 0, \\ & x, y \geq 0. \end{aligned}$$

Let  $(T^*, x^*, y^*)$  be an optimal solution of P2 and  $\lambda^*$  the dual vector associated with its lower-level equality constraint. By setting  $\delta^* = 0$ , one obtains a solution  $(T^*, x^*, y^*, \lambda^*, 0)$  of P1’ with the same objective value, so that the optimal value of P1 (or P1’) is at least as large as that of P2. This is in agreement with our earlier observation about P2 being a restriction of P1. Therefore, it is sufficient to show that any optimal solution of P1 can be matched with a solution of P2 with the same objective value.

Indeed, let  $s^* = (T^*, x^*, y^*, \lambda^*, \delta^*)$  be an optimal solution of P1’, and consider the alternate solution  $s' = (T^* - \delta^*G, x^*, y^*, \lambda^*, 0)$ . By construction,  $s'$  is feasible both for P1’ and P2’. For the latter, we discard the last (or zero) component of  $s'$  because  $\delta$  is not part of P2’. The value of the objective function with  $s'$  is  $(T^* - \delta^*G)x^*$  not less than  $T^*x^*$ . Because P1’ is a relaxation of P2’,  $(T^* - \delta^*G)x^*$  cannot exceed  $T^*x^*$ . Therefore, we have that both objectives

are equal,  $\delta^* Gx^* = 0$ ,  $s$  is optimal for P2', and  $(T^*, x^*, y^*)$  is optimal for P1, as required.  $\square$

Proposition 1 has an interesting economic interpretation. It states that certain constraints imposed on the users can actually be enforced through a suitable and *finite* tariff schedule, and that this can be achieved without negatively affecting the leader's revenue. From the algorithmic point of view, this facilitates the development of efficient procedures that fully exploit the fact that the lower-level problem reduces to a set of independent shortest-path problems.

**COROLLARY 1.** *Constraints (9) of JDP can be moved to the upper level.*

**PROOF.** To prove the result, we will show that for fixed-design vector  $v$ , the resulting pricing problem can be cast in the format P1. To this aim, we introduce total arc flow variables

$$x_a = \sum_{k \in \mathcal{K}} x_a^k$$

and replace the individual commodity constraints  $x_a^k \leq n^k v_a$  by the equivalent global constraint

$$x_a \leq \left( \sum_{k \in \mathcal{K}} n^k \right) v_a.$$

The resulting bilevel program is

$$\max_{T, x, y} Tx - cx,$$

where  $(x, y)$  is an optimal solution of

$$\begin{aligned} \min_{x, y} \quad & \sum_{k \in \mathcal{K}} Tx^k + \sum_{k \in \mathcal{K}} dy^k \\ \text{s.t.} \quad & Ax^k + By^k = b^k \quad \forall k \in \mathcal{H}, \\ & x \leq \left( \sum_{k \in \mathcal{K}} n^k \right) v, \\ & x = \sum_{k \in \mathcal{K}} x^k, \\ & y = \sum_{k \in \mathcal{K}} y^k, \\ & x^k, y^k \geq 0 \quad \forall k \in \mathcal{H}. \end{aligned}$$

Now, by making the correspondences

$$\begin{aligned} x &\equiv x, \\ y &\equiv (y, (x^k)_{k \in \mathcal{K}}, (y^k)_{k \in \mathcal{K}}), \\ G &\equiv I, \\ b^2 &\equiv \left( \sum_{k \in \mathcal{K}} n^k \right) v, \end{aligned}$$

and

$$Ex + Fy = b^1 \equiv \begin{pmatrix} I & 0 & -I & \cdots & -I & 0 & \cdots & 0 \\ 0 & I & 0 & \cdots & 0 & -I & \cdots & -I \\ 0 & 0 & A & \cdots & 0 & B & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & A & 0 & \cdots & B \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ (x^k)_{k \in \mathcal{K}} \\ (y^k)_{k \in \mathcal{K}} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ (b^1) \\ \vdots \\ (b^{|\mathcal{K}|}) \end{pmatrix},$$

one recovers the generic form of P1, and the result follows.  $\square$

Finally, we note that the assumptions underlying Corollary 1 could be slightly relaxed to care for situations where P1 and P2 are allowed to be infeasible or unbounded. To avoid technicalities, the proofs have been omitted.

## 4. A Solution Procedure for JDP

Most bilevel programs that have been proposed in the literature are “intractable,” due mainly to the complementarity constraints that arise when one replaces the lower-level problem by its optimality conditions to achieve a standard mathematical program. Actually, the JDP subsumes the model of Labbé et al. (1998), which has been shown by Roch et al. (2005) to be strongly NP-hard, even when restricted to a single origin-destination pair. JDP also generalizes the network design problem, another notoriously difficult combinatorial problem. It follows that the JDP compounds the difficulties of two NP-hard problems, and thus some interesting algorithmic challenge.

In this section, we propose an iterative algorithm that adapts the Lagrangean relaxation framework to bilevel programs. In this framework, we treat constraints (11) as the “complicating” ones, and append them to the objective to form a Lagrangean function. As in classical Lagrangean procedures, the algorithm alternates between an outer phase, where Lagrange multipliers are updated, and an inner phase, where the concave dual function (Lagrangean) is evaluated.

To evaluate the dual function, one needs to solve the Lagrangean subproblem, itself an NP-hard toll optimization problem. This latter problem is solved using a variant of the primal-dual algorithm proposed in Brotcorne et al. (2000), where the subproblem is reformulated as a single-level problem through the use of an exact penalty function applied to the lower-level complementarity term.

The resulting bilinear program is solved iteratively, à la Gauss-Seidel, by optimizing sequentially with respect to the upper-level (tariffs) and lower-level (flows) variables. Once a local optimum has been reached, the penalty factor is increased and the process repeated until no progress is achieved. The baton is then passed to the outer phase.

Let us now turn our attention to a formal description of the procedure. After having moved the capacity constraints from the lower to the upper level in accordance with Corollary 1, we construct the dual function  $L(u)$ :

(LSP( $u$ ))

$$L(u) = \max_{T, v, x, y} \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k T_a x_a^k - \sum_{a \in \mathcal{A}_1} f_a v_a - \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k c_a x_a^k + \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} u_a^k (v_a - x_a^k)$$

s.t.  $v_a \in \{0, 1\} \quad \forall a \in \mathcal{A}_1,$

where  $(x, y)$  is an optimal solution of

$$\min_{x, y} \sum_{k \in \mathcal{K}} n^k \left( \sum_{a \in \mathcal{A}_1} T_a x_a^k + \sum_{a \in \mathcal{A}_2} d_a y_a^k \right)$$

s.t.  $Ax^k + By^k = e^k \quad \forall k \in \mathcal{K},$   
 $x^k, y^k \geq 0 \quad \forall k \in \mathcal{K},$

whose evaluation requires the solution of a bilevel program. Because, for each  $u \geq 0$  LSP( $u$ ) is a relaxation of JDP, the solution  $L(u)$  to LSP( $u$ ) provides an upper bound on the optimal value of JDP; the best Lagrangean bound is obtained by solving the dual problem:

$$(DL) \quad \min\{L(u) : u \geq 0\}. \quad (12)$$

In our implementation, the dual problem is maximized by an algorithm inspired from subgradient optimization. At a given iteration, a predetermined step is taken along a direction specified by an approximate solution of LSP( $u$ ) rather than based on the *true* subgradient  $v(u) - x(u)$  that would correspond to any optimal  $(v(u), x(u))$  solution of LSP( $u$ ). In the resulting algorithm (outer phase) outlined below,  $|K|$  denotes the number of commodities,  $\gamma_j$  the step size at iteration  $j$ , and  $Z^*$  the best upper-level value achieved so far.

### Algorithm JDP (Outer Phase)

Step 0 (initialization)

- $u_a^0 \leftarrow f_a/|K| + \epsilon$ ;  $Z^* \leftarrow -\infty$ ;  $T^0 \leftarrow 0$
- $(x^0, y^0) \leftarrow$  an optimal lower-level solution consistent with  $T^0$
- $j \leftarrow 1$

Step  $j$

- $(T^j, v^j, x^j, y^j) \leftarrow$  an approximate solution of LSP( $u^{j-1}$ )
- **if** solution improved **then** update  $Z^*$
- $u^j \leftarrow \max\{0, u^{j-1} - \gamma^j(v^j - x^j)\}$
- **if** stopping criterion is met **then** halt
- **else**  $j \leftarrow j + 1$  and repeat Step  $j$

The efficiency of the above algorithm rests on a procedure able to provide a near-subgradient within a reasonable amount of time. To this aim, we adapted the primal-dual scheme initially proposed by Brotcorne et al. (2000, 2001), which addresses by a penalty method the single-level reformulation of JDP, where the lower-level problem is replaced by its Kuhn-Tucker optimality conditions, i.e.,

$$Z(u) = \max_{T, v, x, y, \lambda} \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k T_a x_a^k - \sum_{a \in \mathcal{A}_1} f_a v_a - \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k c_a x_a^k + \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} u_a^k (v_a - x_a^k)$$

s.t.  $v_a \in \{0, 1\} \quad \forall a \in \mathcal{A}_1,$   
 $Ax^k + By^k = e^k \quad \forall k \in \mathcal{K},$  (13)  
 $x^k, y^k \geq 0 \quad \forall k \in \mathcal{K},$   
 $\lambda^k A \leq T \quad \forall k \in \mathcal{K},$   
 $\lambda^k B \leq d \quad \forall k \in \mathcal{K},$   
 $\sum_{k \in \mathcal{K}} n^k (Tx^k + dy^k - \lambda^k e^k) = 0,$

where the last constraint, always nonnegative for feasible triple  $(x^k, y^k, \lambda^k)$ , specifies the equality of the primal and dual lower-level objectives. Penalizing this constraint yields the disjoint<sup>1</sup> bilinear program:

$$(PEN) \quad \max_{T, v, x, y, \lambda} \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k T_a x_a^k - \sum_{a \in \mathcal{A}_1} f_a v_a - \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k c_a x_a^k + \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} u_a^k (v_a - x_a^k) - M_1 \sum_{k \in \mathcal{K}} n^k \left( \sum_{a \in \mathcal{A}_1} T_a x_a^k + \sum_{a \in \mathcal{A}_2} d_a y_a^k - \lambda^k e^k \right)$$

s.t.  $v_a \in \{0, 1\} \quad \forall a \in \mathcal{A}_1,$   
 $Ax^k + By^k = e^k \quad \forall k \in \mathcal{K},$   
 $x^k, y^k \geq 0 \quad \forall k \in \mathcal{K},$   
 $\lambda^k A \leq T \quad \forall k \in \mathcal{K},$   
 $\lambda^k B \leq d \quad \forall k \in \mathcal{K},$

where  $M_1$  is some positive number, and whose objective can be rewritten as

$$\sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} ((1 - M_1)n^k T_a - u_a^k - n^k c_a) x_a^k - M_1 \sum_{k \in \mathcal{K}} n^k \left( \sum_{a \in \mathcal{A}_2} d_a y_a^k - \lambda^k e^k \right) - \sum_{a \in \mathcal{A}_1} \left( f_a - \sum_{k \in \mathcal{K}} u_a^k \right) v_a.$$

This latter problem is separable in  $v$  and  $T, x, y, \lambda$ ; the binary variables  $v_a$  ( $a \in \mathcal{A}_1$ ) are set to one whenever the corresponding term

$$\sum_{k \in \mathcal{K}} u_a^k - f_a$$

is positive, and to zero otherwise. The procedure for solving PEN then iterates between the leader's tariff vector and the

follower's commodity flows  $x^k$  and  $y^k$ . For fixed penalty factor  $M_1$ , it consists of three main steps:

- Given a fixed  $(x, y)$  flow vector, solve PEN for  $T$ .
- Given a fixed  $T$  vector, solve PEN for  $(x, y)$ .
- For fixed  $(x, y)$  flow vector, find a compatible revenue-maximizing  $T$ .

The grand strategy underlying algorithm PEN is to induce modifications (basis changes) in the lower-level solution. In this process, extremal flow assignments corresponding to distinct values of the tariff vector  $T$  are generated, and we expect one of these combinations to be of high quality for JDP.

At a given iteration, the tariff vector  $T$  solves the penalized problem PEN for fixed-flow vectors  $x^k, y^k$  (Step 1). Next, one solves for an optimal assignment with respect to  $T$  (Step 2); this is achieved by computing shortest paths for all OD pairs. The revenue-maximizing tariff vector consistent with  $(x, y)$  is then the solution of the linear program (Step 3). The main components of the primal-dual algorithm are made explicit below. At Step 0, arc flows are initialized to values that achieved the best leader profit at one of the previous main iterations, whereas the design vector  $v$  is set to the optimal solution of the problem:

$$\begin{aligned} \text{(PEN1}(v)) \quad & \max_v \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} (u_a^k - f_a) v_a \\ \text{s.t.} \quad & v_a \in \{0, 1\} \quad \forall a \in \mathcal{A}_1. \end{aligned}$$

At Step 1 of the algorithm, and for fixed-commodity flows  $x^k$ , let  $T$  and  $\lambda$  be optimal solutions of the problem:

$$\begin{aligned} \text{(PEN2}(T, \lambda)) \quad & \max_{T, \lambda} (1 - M_1) \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k x_a^k T_a + M_1 \sum_{k \in \mathcal{K}} n^k \lambda^k e^k \\ \text{s.t.} \quad & \lambda^k A \leq T \quad \forall k \in \mathcal{K}, \\ & \lambda^k B \leq d \quad \forall k \in \mathcal{K}, \end{aligned}$$

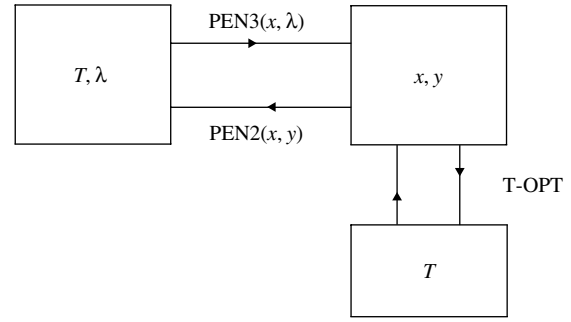
whose dual is a multicommodity flow problem. At Step 2, the commodity flows  $x^k$  and  $y^k$  solve, for fixed-tariff vector  $T$ , the linear program

$$\begin{aligned} \text{(PEN3}(x, y)) \quad & \max_{x, y} \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} ((1 - M_1) n^k T_a - u_a^k - n^k c_a) x_a^k \\ & - M_1 \sum_{a \in \mathcal{A}_2} \sum_{k \in \mathcal{K}} n^k d_a y_a^k \\ \text{s.t.} \quad & Ax^k + By^k = e^k \quad \forall k \in \mathcal{K}, \\ & x^k, y^k \geq 0 \quad \forall k \in \mathcal{K}. \end{aligned}$$

At Step 3, the algorithm computes a common tariff vector that maximizes the total profit of the leader while maintaining the lower-level optimality of the current commodity flows. The structure of this “inverse” problem is that of the uncapacitated multicommodity network flow problem

$$\begin{aligned} \text{(T-OPT)} \quad & \max_{T, \lambda} \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k T_a x_a^k - \sum_{a \in \mathcal{A}_1} \sum_{k \in \mathcal{K}} n^k c_a x_a^k \\ \text{s.t.} \quad & \lambda^k A \leq T \quad \forall k \in \mathcal{K}, \\ & \lambda^k B \leq d \quad \forall k \in \mathcal{K}, \\ & n^k (Tx^k + dy^k - \lambda^k e^k) = 0 \quad \forall k \in \mathcal{K}. \end{aligned}$$

**Figure 2.** The algorithmic framework.



The structure of the algorithm is illustrated in Figure 2 and summarized below, where  $\alpha \in (0, 1)$  denotes a relaxation factor and  $Z^*$  represents the current best objective value.

### Primal-Dual Algorithm (Inner Iteration)

- Step 0:* (initialization)
- $j$  (major iteration index);  $Z^* \leftarrow -\infty$
  - **if**  $\sum_{k \in \mathcal{K}} (u_a^k - f_a) \geq 0$  **then**  $v_a^j \leftarrow 1$  **else**  $v_a^j \leftarrow 0$
  - $l \leftarrow 1$  (minor iteration index)
- Step 1:* (computation of  $T_l$  and  $\lambda_l$ )
- for fixed  $x_{l-1}^k$  and  $y_{l-1}^k$ ,  $(T_l, \lambda_l) \leftarrow$  solution of PEN2( $T, \lambda$ )
- Step 2:* (computation of  $x_l$  and  $y_l$ )
- solve (PEN3( $x, y$ )) for the tariff vector  $(1 - \alpha)T_l + \alpha T_{l-1}$
- Step 3:* (computation of optimal tariffs for given flows)
- if** flows are identical to those of some previous iteration
  - then** go to Step 4
  - else**
  - $\tilde{T} \leftarrow$  optimal solution of T-OPT
  - **if**  $x_{lg} = 1$  **then**  $\tilde{v}_a \leftarrow 1$  **else**  $\tilde{v}_a \leftarrow 0$
  - let  $\tilde{Z}(\tilde{T}, \tilde{v}) = \tilde{T} x_l - f \tilde{v} - c x_l$ .
  - **if**  $\tilde{Z} > Z^*$  **then**  $Z^* \leftarrow \tilde{Z}$  and  $(T^*, v^*, x^*, y^*, \lambda^*) \leftarrow (\tilde{T}, \tilde{v}, x_l, y_l, \lambda_l)$
- Step 4:* (stopping criterion)
- if** stopping criterion met **then**  $(T^j, v^j, x^j, y^j, \lambda^j) \leftarrow (T^*, v^*, x^*, y^*, \lambda^*)$
  - else**  $l \leftarrow l + 1$ , increase the penalty factor  $M_1$  and return to Step 1.

## 5. Numerical Results

The algorithm has been tested on both randomly generated and real data. The random problems consist of complete grid networks that promote interaction between commodities, and makes for problems that are combinatorially challenging. The real-life instances correspond, respectively, to a freight transportation network in the Nord-Pas-de-Calais region of northern France, and to a doctored version of France Telecom's backbone network. Before addressing the large instances, the algorithm's parameters have been calibrated with respect to random instances for which the optimal solution was computable by a mixed-integer approach.



All computations have been performed on a Sun Enterprise 10,000 workstation.

The generation of random instances is adapted from techniques introduced in Brotcorne et al. (2001) for a pure pricing problem and where there exists at least one toll-free path for each commodity. The main parameters of the problem are: cost symmetry (yes or no), size of the network (number of commodities, nodes, arcs), proportion of design arcs; all toll arcs involve a design variable. Note that this is without loss of generality because one may assign a null cost to a tariff arc that is part of the network. All experiments are based on networks of 60 nodes ( $5 \times 12$  grid) and 208 arcs. Even on such medium-sized networks, finding a good solution can be extremely difficult, depending on the choice of key parameters such as the percentage of design arcs and the number of commodities. In our experiment, these parameters belonged to the ranges [5%, 20%] and [10, 40], respectively.

As stated previously, the training set consisted of instances that could be solved for their global optimum. These involved networks with 10% or 15% of design arcs and 10 commodities. The parameters that were adjusted through sensitivity analysis were the following:

- $u_a^0$  (initial value of the Lagrange multiplier);
- $l$  (fixed step size along the subgradient direction);
- $m$  (number of inner primal-dual iterations);
- $\alpha$  (relaxation factor);
- $\delta$  (parameter involved in the stopping criterion).

Tables 1 through 5 are concerned with the calibration of the algorithm’s various parameters, with the exception of the penalty factor  $M_1$  involved in the Lagrangean phase, which has been initialized to 1.3, and incremented by 0.05 at the end of each primal-dual iteration.

Starting from the initial setting ( $u_a^0 = 0, l = 1, m = 1, \alpha = 0, \delta = 300$ ), the parameters have been calibrated in a sequential (left-to-right) manner. When treating one parameter, the “left” parameters are fixed at their updated value,

**Table 1.** Sensitivity analysis on Lagrange multiplier  $u_0$  ( $u_0, l = 1, m = 1, \alpha = 0, \delta = 300$ ).

$u_0$	Heuristic						CPLEX
	%TA	#TA	DI	#BAS	H/M	CPU	CPU
0	10	5.5	69.7	36.8	0.85	22.2	28.3
0	15	7.3	67.9	46.5	0.71	25.0	29.1
( $\min f_a$ ) $- 0.001/ K $	10	5.5	71.6	36.7	0.91	22.4	28.3
( $\min f_a$ ) $- 0.001/ K $	15	6.2	68.2	41.0	0.83	24.1	29.1
$f_a/ K  - 0.01$	10	5.3	57.8	33.1	0.93	21.5	28.3
$f_a/ K  - 0.01$	15	6.5	68.5	38.9	0.80	24.8	29.1
$f_a/ K  + 0.01$	10	5.4	56.2	30.1	0.92	21.9	28.3
$f_a/ K  + 0.01$	15	7.1	56.0	37.0	0.82	24.3	29.1
( $\max f_a$ ) $/ K $	10	5.2	66.3	33.2	0.93	21.8	28.3
( $\max f_a$ ) $/ K $	15	7.1	68.6	37.4	0.76	24.4	29.1
$1,000/ K $	10	5.3	55.5	23.0	0.80	21.4	28.3
$1,000/ K $	15	5.3	41.7	28.2	0.70	22.8	29.1

**Table 2.** Sensitivity analysis on step lengths  $l$  ( $u_0 = f_a/|K| + 0.01, l, m = 1, \alpha = 0, \delta = 300$ ).

$l$	Heuristic						CPLEX
	%TA	#TA	DI	#BAS	H/M	CPU	CPU
1	10	5.4	56.2	30.1	0.92	21.9	28.3
1	15	7.1	56.0	37.0	0.82	24.3	29.1
5	10	5.4	18.5	22.8	0.94	21.4	28.3
5	15	6.4	40.8	25.2	0.93	23.8	29.1
10	10	5.1	14.2	17.1	0.94	22.0	28.3
10	15	6.5	18.7	20.2	0.92	20.2	29.1
15	10	5.1	29.8	18.7	0.95	21.6	28.3
15	15	6.8	30.2	18.1	0.93	22.5	29.1
20	10	5.4	39.3	16.8	0.95	20.9	28.3
20	15	6.4	33.0	17.7	0.92	22.6	29.1

and the “right” parameters at their initial value. The meaning of column headings is the following: “%TA” refers to the percentage of design arcs, “#TA” to the number of design arcs actually opened in the solution, “DI” to the index of the subgradient iteration at which the solution was reached, “#BAS” to the number of follower basis encountered during the iterative process, and “H/M” to the ratio of the heuristic objective over the optimal solution achieved by the mixed integer optimizer Cplex 8.1. The “CPU” labels refer to running times expressed in seconds. Each line provides CPU times averaged over 10 randomly generated instances.

Table 1 presents the results for different values of the initial Lagrange multiplier  $u_a^0$ . Very small (respectively very high) values of the multiplier force the design arcs to be initially closed (respectively, open). A sufficient and necessary condition for an arc  $a$  to be opened, at a given iteration, is that  $\sum_{k \in \mathcal{K}} u_a^k - f_a$  be positive. A compromise is reached by setting, for every commodity  $k$ , the Lagrange multiplier  $u_a^{0,k}$  to the value  $(f_a/|\mathcal{K}|) + \epsilon$ , where  $\epsilon$  is some small positive number. Accordingly, all design arcs are open before entering the subgradient procedure.

**Table 3.** Sensitivity analysis on the number of primal-dual iterations  $m$  ( $u_0 = f_a/|K| + 0.01, l = 5, m, \alpha = 0, \delta = 300$ ).

$m$	Heuristic						CPLEX
	%TA	#TA	DI	#BAS	H/M	CPU	CPU
1	10	5.4	18.5	22.8	0.94	21.0	28.3
1	15	6.4	40.8	25.2	0.93	24.0	29.1
5	10	5.0	19.1	54.8	0.96	83.5	28.3
5	15	6.5	26.6	46.0	0.94	88.9	29.1
10	10	4.9	20.5	68.1	0.96	161.3	28.3
10	15	6.6	26.7	59.3	0.97	168.1	29.1
15	10	4.7	23.0	82.5	0.98	230.0	28.3
15	15	6.2	27.1	69.1	0.98	246.0	29.1
20	10	4.6	24.6	102.7	0.97	311.9	28.3
20	15	6.3	28.4	79.9	1.00	334.2	29.1

**Table 4.** Sensitivity analysis on the overrelaxation factor  $\alpha$  ( $u_0 = f_a/|K| + 0.01, l = 5, m = 20, \alpha, \delta = 300$ ).

$\alpha$	Heuristic						CPLEX
	%TA	#TA	DI	#BAS	H/M	CPU	CPU
0	10	4.6	24.6	102.7	0.97	311.9	28.3
0	15	6.3	28.4	79.9	1.00	334.2	29.1
0.3	10	4.5	24.8	105.7	0.98	279.6	28.3
0.3	15	6.3	26.1	88.3	0.99	282.1	29.1
0.5	10	4.6	23.4	114.3	0.98	282.5	28.3
0.5	15	6.3	27.1	87.3	1.00	282.3	29.1
0.7	10	4.7	20.7	96.3	0.97	279.3	28.3
0.7	15	6.2	27.8	87.8	0.99	284.4	29.1

The impact of the step size  $l$  taken along the subgradient direction is assessed in Table 2. Our choice of the value  $l = 5$  has been motivated by the fact that it produced solutions of high quality and allowed the algorithm to proceed through a high number of basic solutions. This is akin to a diversification phase, where each basic solution is probed using inverse optimization.

The maximum number of primal-dual steps  $m$  has been set to 20, a value that produced a high number of optimal solutions for the largest instances (see Table 4). Note that:

- The number of basis visited increases with  $m$ , and so does the quality of the solution (almost always);
- The computation time grows linearly with  $m$ .

The parameter  $\alpha$  has been set to 0.5 (see Table 4). Note that this parameter does not have a significant impact on either CPU time or the quality of the solution. Finally, the parameter  $\delta$  that controls the stopping of the algorithm has been set to 30 (see Table 5). Because higher values do not improve the solution and do result in higher CPU times, they have not been considered. From now on, the parameter vector is set to ( $u_0 = f_a/|K| + 0.01, l = 5, m = 20, \alpha = 0.5, \delta = 30$ ).

The numerical results on randomly generated networks are summarized in Tables 6 to 8. Each line corresponds to an average taken over five problem instances. In addition to previous column headings, label “Type” refers to

**Table 5.** Sensitivity analysis on the stopping criterion ( $u_0 = f_a/|K| + 0.01, l = 5, m = 20, \alpha = 0.5, \delta$ ).

$\delta$	Heuristic						CPLEX
	%TA	#TA	DI	#BAS	H/M	CPU	CPU
10	10	5.0	17.9	87.2	0.94	33.4	28.3
10	15	7.2	18.1	71.1	0.90	34.4	29.1
20	10	4.7	19.3	91.4	0.97	40.3	28.3
20	15	6.6	18.9	79.6	0.96	42.5	29.1
30	10	4.6	23.4	112.6	0.99	60.6	28.3
30	15	6.3	27.1	86.7	1.00	63.6	29.1
40	10	4.6	23.4	112.9	0.99	69.4	28.3
40	15	6.3	27.1	86.8	1.00	73.3	29.1

**Table 6.** Fixed cost sensitivity on asymmetric networks (20 commodities).

$f_a$	Heuristic								CPLEX
	%TA	NOPT	#TA	DI	#BAS	BOPT	H/M	CPU	CPU
0	5	5	9.0	0.6	16.0	12.0	0.99	28.4	1,959.6
0	10	3	15.6	0.4	17.6	8.4	1.03	31.8	8,682.0
0	15	0	27.0	16.0	18.2	17.0	1.03	53.4	15,974.0
0	20	0	34.4	0.8	9.8	7.4	1.05	45.6	16,453.4
30	5	5	4.8	19.8	159.6	133.60	0.98	6.0	251.4
30	10	5	5.4	27.0	187.4	171.2	0.97	84.8	879.6
30	15	4	9.8	32.8	196.0	181.2	1.05	118.4	14,480.0
30	20	1	12.6	33.4	348.2	236.4	1.02	268.6	23,446.8
60	5	5	2.0	29.2	177.0	165.6	1.00	67.8	29.6
60	10	5	3.0	33.2	232.4	187.8	0.97	86.6	35.8
60	15	5	5.6	37.2	317.6	263.6	1.00	140.8	3,500.2
60	20	4	7.0	49.4	466.8	426.4	1.02	314.6	6,561.0

the arc cost structure, either symmetric (S) or asymmetric (AS); label “ $|\mathcal{K}|$ ” represents the number of commodities; “BOPT” refers to the position (rank) of the basis within the sequence of basis generated by the heuristic procedure; the label “H/M” now refers to the ratio of the heuristic objective over the best (not necessarily optimal) solution achieved by the mixed-integer programming code CPLEX 8.1. The latter was halted whenever one of these conditions was fulfilled: CPU time exceeded eight hours, the number of nodes explored in the enumeration process exceeded 400,000, or memory requirements exceeded one gigabyte. Finally, label “NOPT” refers to the number of problems solved to optimality. On the larger instances presented in Table 8, CPLEX was unable to find feasible solutions, hence the absence of a “CPLEX” column.

Tables 6 and 7 show the results of sensitivity analyses performed with respect to the opening cost  $f_a$ . The sensitivity analyses confirm some intuitive results. For instance, when the ratio of opening to operating costs is high, most

**Table 7.** Fixed cost sensitivity on symmetric networks (20 commodities).

$f_a$	Heuristic								CPLEX
	%TA	NOPT	#TA	DI	#BAS	BOPT	H/M	CPU	CPU
0	5	4	9.6	0.0	9.2	5.6	0.99	27.6	375.0
0	10	1	18.6	3.4	10.8	7.4	1.03	3.0	11,941.8
0	15	0	25.4	8.2	14.8	13.6	1.19	47.6	20,458.2
0	20	1	35.8	1.6	19.6	15.8	1.15	52.0	21,470.4
30	5	5	6.8	8.8	41.8	29.0	0.98	47.5	2,488.0
30	10	3	6.6	19.0	129.0	82.4	1.27	92.0	13,982.4
30	15	1	8.6	41.0	288.6	277.4	1.07	195.5	21,866.8
30	20	1	7.8	37.6	339.4	299.2	1.14	241.9	23,553.2
60	5	5	4.2	22.4	86.0	82.8	1.00	63.6	158.6
60	10	5	3.8	27.0	168.4	154.8	0.98	79.4	543.8
60	15	3	5.4	38.6	356.2	271.0	0.96	176.0	12,750.8
60	20	4	4.8	46.2	335.6	295.4	1.00	184.0	6,695.6

**Table 8.** Asymmetric and symmetric networks (40 commodities).

Type	Heuristic						
	%TA	#TA	DI	#BAS	BOPT	OPT	CPU
AS	5	3.6	22.2	246.8	225.8	401.0	189.8
AS	10	8.2	30.0	426.8	270.6	719.2	290.5
AS	15	7.2	52.4	817.0	667.8	1,078.4	966.8
AS	20	9.4	60.0	1,052.4	778.6	1,089.8	1,682.8
S	5	4.2	25.2	326.8	283.0	434.0	212.6
S	10	4.8	36.4	569.8	444.4	463.6	531.0
S	15	5.2	42.6	686.4	585.4	402.4	818.0
S	20	11.0	49.4	910.2	684.0	1,437.6	2,312.6

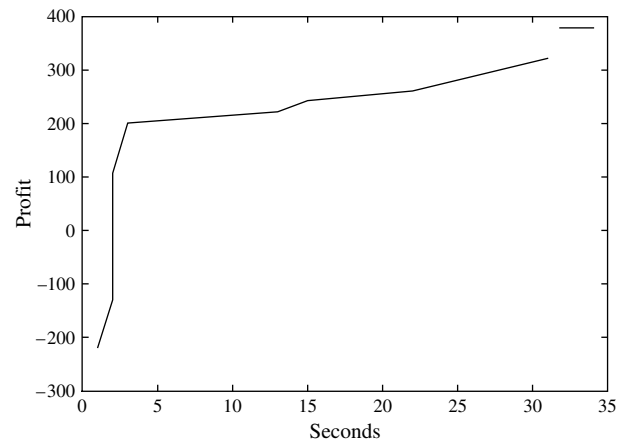
tariff arcs are left closed. In this case, the combinatorial structure is “weak” and it is not surprising to observe that CPLEX can easily solve this class of problems. The converse conclusion holds when this ratio is low. We also remark that symmetric problems tend to be more difficult to solve.<sup>2</sup>

As a general rule, we may infer that Lagrangean relaxation produces high-quality solutions rapidly and consistently. Typically, the solutions are greatly superior to those obtained by CPLEX. With the exception of the smallest instances (10 commodities, 5% or 10% tariff arcs) the proposed heuristic is also much faster than the exact MIP algorithm. It has been observed that even if the CPU time required by the heuristic increases with the percentage of tariff arcs and the number of commodities, this increase is more modest than for CPLEX.

All 10-commodity instances, with the exception of instances with null fixed cost and 20% of tariff arcs, could be solved by CPLEX. However, running times rise steeply and in an unstable fashion as the number of tariff arcs is increased from 5% to 20%. In contrast, Table 7 shows moderate CPU times for the Lagrangean algorithm, for which both symmetric and asymmetric 20-commodity instances could be solved, with no significant decrease in solution quality. As observed, the symmetric instances proved more difficult. Indeed, beyond 20 commodities, these problems could not be solved to optimality by CPLEX. This remark is compatible with the fact that, as indicated in columns “BOPT” and “#BAS,” the best bases are uncovered toward the end of the enumeration process.

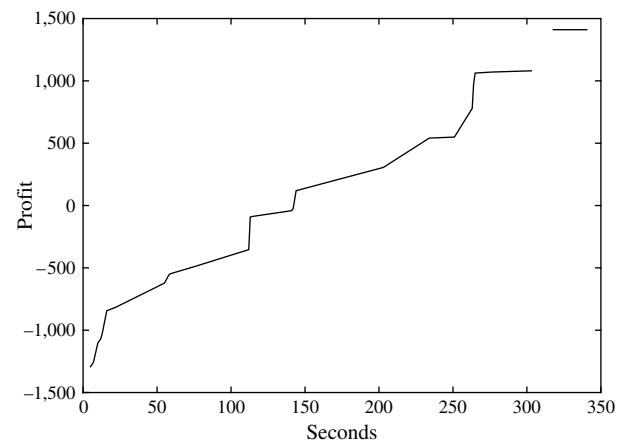
In Table 8 are presented the results pertaining to the largest instances, with the parameters  $f_a$  set to 30. It is interesting to have a look at the upper envelopes on profit function, which are given in Figures 3 and 4 for two typical instances involving, respectively, 20 commodities and 40 commodities. These functions exhibit similar behavior: They increase sharply at the beginning of the process, and flatten out in the middle and at the end of the algorithmic process. Note that although the initial iterates produce negative revenues, they still participate in the “open-close” process that eventually yields near-optimal solutions.

**Figure 3.** Network with 20 commodities.



The real-life instances come from the freight transportation and telecommunication areas. First we consider an aggregate transportation network of the Nord-Pas-de-Calais region (France) consisting of 78 nodes, 180 links, 42 taxable arcs, and 40 OD pairs. The transportation network supports a large amount of freight flows, mainly due to the locations of three main harbours (Dunkerque, Calais, and Boulogne), a large metropolitan area (Lille), and a multimodal transportation platform in Dourges (road, railway, waterway network). Moreover, the region is crossed from north to south and east to west by highways. The transportation data come from the Conseil Régional du Nord-Pas-de-Calais. Three scenarios have been considered, the first (NPDC1) corresponding to the base scenario. The second scenario is characterized by the existence of a direct link between each OD pair (NPDC2), and the third by the increase of freight to and from Dunkerque (NPDC3). Whether or not to open a taxable arc corresponds to the decision of a service carrier (the leader) to accept a request or not and, whenever the request is accepted, to set a price. The cost of the arcs are related to the distance and the type of associated road.

**Figure 4.** Network with 40 commodities.



**Table 9.** Freight transportation instances.

Instance	Heuristic					CPLEX	
	#TA	DI	#BAS	H/M	CPU	GAP	CPU
NPDC1*	28	45	330	1.05	295	58.11	7,093
NPDC2*	32	49	280	0.98	321	39.92	7,888
NPDC1*	28	36	239	0.98	274	39.12	7,108

\*Could not be solved to optimality by CPLEX.

Second, we consider telecommunication instances. In this area, the leader is a service provider who has to make network deployment decisions and set usage prices. The network represents the French telecommunication backbone network (doctored for confidentiality reasons) that consists of 20 nodes, 100 links, and 65 taxable links. Three instances were considered, involving number 40, 60, and 80 origin-destination pairs, respectively. The results, which are reported in Tables 9 and 10, are self-explanatory. In all but one instance (TELE1), an optimal solution of JDP could not be achieved by CPLEX. Worse, no feasible (integral) solution could be found for TELE3, within the allocated time limit. Empirically, the results in Tables 9 and 10 illustrate that the heuristic uses moderate CPU time to produce a high-quality solution. From a qualitative point of view, it has been observed that the cost of “tariff” paths frequently matches those of the shortest tariff-free paths. However, in certain instances, tariffs have been set to smaller values to induce customers to travel on tariff paths and maximize revenue.

## 6. Conclusion

In this paper, we introduced a mixed continuous-discrete design problem that arises naturally when modelling pricing decisions over transportation networks, and proposed for its solution an algorithm based on the application of Lagrangean relaxation within a bilevel programming framework. In view of the algorithm’s encouraging performance on random problems involving more than 4,000 variables, we are looking forward to applying the methodology more general problems involving link capacities and elastic demand. Finally, while some applications would require to take into account congestion, we believe that the resulting nonlinear models would call for an entirely different kind of approach.

**Table 10.** Telecommunication instances.

Instance	Heuristic					CPLEX	
	#TA	DI	#BAS	H/M	CPU	GAP	CPU
TELE1	16	24	28	1.00	73	0.00	166
TELE2*	20	117	2,333	1.00	632	2.66	7,397
TELE3*	56	20	391	—	759	inf	7,019

\*Could not be solved to optimality by CPLEX.

## Endnotes

1. A bilinear program with objective  $uQv + cu + dv$  is disjoint if none of its linear constraints involve both vectors  $u$  and  $v$ .
2. Interestingly, such a result is a “folk theorem” in the case of the traveling salesman problem.

## Acknowledgments

This work was partially supported by NSERC (Canada), FQRNT (Québec), France Telecom Research and Development, contract number 001B852, under the supervision of Mustapha Bouhtou and Jean-Luc Lutton. Thanks are also due to Amélie Forget for her programming help, and to two referees for their careful reading and constructive comments.

## References

- Ahuja, R. K., T. L. Magnanti, J. B. Orlin. 1993. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, Englewood Cliffs, NJ.
- Başar, T., R. Srikant. 2002. Revenue-maximizing pricing and capacity expansion in a many-users regime. Presented at IEEE Infocom, New York (June).
- Bashyam, T. 2000. Service design and price competition in business information service. *Oper. Res.* **48** 362–375.
- Bienstock, D., O. Raskina, I. Saniee, Q. Wang. 2006. Combined network design and multi-period pricing: Modeling, solution techniques, and computation. *Oper. Res.* **54** 261–276.
- Brotcorne, L., M. Labbé, P. Marcotte, G. Savard. 2000. A bilevel model and solution algorithm for a freight tariff-setting problem. *Transportation Sci.* **34** 289–302.
- Brotcorne, L., M. Labbé, P. Marcotte, G. Savard. 2001. A bilevel model for toll optimization on a multicommodity transportation network. *Transportation Sci.* **35** 1–14.
- Budenbender, K., T. Grunert, H. Sebastian. 2000. A hybrid tabu search/branch and bound algorithm for the direct flight network design problem. *Transportation Sci.* **34** 364–380.
- Grigoriev, A., S. van Hoesel, A. F. van der Kraaij, M. Uetz, M. Bouhtou. 2005. Pricing network edges to cross a river. G. Persiano, R. Solis-Oba, eds. *Proc. 2nd Workshop on Approximation and Online Algorithms (WAOA 2004), Lecture Notes in Computer Science*, Vol. 3351. Springer, Berlin, 140–153.
- Kerr, F. 2001. Technology utilization & collaborative strategies—An analysis of the leading freight transportation companies report. First Conferences Ltd. eyefortransport. <http://www.eyefortransport.com>.
- Labbé, M., P. Marcotte, G. Savard. 1998. A bilevel model of taxation and its application to optimal highway pricing. *Management Sci.* **44** 1608–1622.
- Lanning, S., D. Mitra, Q. Wang, M. Wright. 2000. Optimal planning for optical transport networks. *Philosophical Trans.: Math., Physical Engrg. Sci.* **358** 2183–2219.
- Lederer, P. J. 1993. A competitive network design problem with pricing. *Transportation Sci.* **27** 25–38.
- Lederer, P. J., R. S. Nambimadon. 1998. Airline network design. *Oper. Res.* **46** 785–804.
- Roch, S., G. Savard, P. Marcotte. 2005. Design and analysis of an algorithm for Stackelberg network pricing. *Networks* **46** 57–67.
- Van Ackere, A. 1993. The principal/agent paradigm: Its relevance to various functional fields. *Eur. J. Oper. Res.* **70** 83–103.
- van Hoesel, S., A. F. van der Kraaij, G. Mannino, G. Oriolo, M. Bouhtou. 2003. Polynomial cases of the tariffication problem. RM03053, Maastricht Economic Research School on Technology and Organisation, Maastricht, The Netherlands.