Decomposition Techniques for the Minimum Toll Revenue Problem

Lihui Bai, Donald W. Hearn, and Siriphong Lawphongpanich
Center for Applied Optimization, Industrial and Systems Engineering Department, University of Florida, Gainesville, Florida 32611-6595

The objective of the minimum toll revenue (MINREV) problem is to find tolls that simultaneously cause users to use the transportation network efficiently and minimize the total toll revenues that must be collected. This article investigates the Dantzig-Wolfe (DW) decomposition as an approach for solving the MINREV problem and establishes its relationships with a cutting plane algorithm and other proposed approaches. The article also identifies the variant of DW decomposition most suitable for implementation. Numerical experiments with real transportation networks suggest that DW decomposition is robust and should be used when the problems are too large for standard linear programming software. Although transportation planning is the application emphasized in this article, it should be noted that the MINREV problem also has applications in telecommunication network design and control. © 2004 Wiley Periodicals, Inc. NETWORKS, Vol. 44(2), 142–150 2004

Keywords: congestion pricing; Dantzig-Wolfe decomposition; cutting plane method

1. INTRODUCTION

Recently, toll pricing as a means to relieve traffic congestion has received significant attention by transportation planners as it is being implemented in large cities such as London [30]. This concept of controlling congestion has also received considerable attention in academic literature (e.g., [2, 14, 21, 26, 29, 33]) where it has long been endorsed. However, academic research often focuses on small examples that are suitable for detailed analysis. This article is therefore motivated by the need to determine toll prices in a more practical setting that generally involves large-scale networks.

The focus here is on solving the minimum toll revenue problem or MINREV. This problem, originally referred to as the minimum system or MINSYS problem, is one of several prototypical toll pricing models first introduced in Hearn and Ramana [16]. As a prototype, MINREV assumes that every road is tollable and the travel demand is fixed. Although these two assumptions may not be practical in all transportation networks, MINREV has since stimulated further studies in toll pricing. For example, Dial [9] uses MINREV as a model to illustrate basic concepts in toll pricing and develop new algorithms, Hearn and Yildirim [17] develop a toll pricing framework for models with elastic demand and show that the toll revenue in this case is constant (see also Larsson and Patriksson [22]), and Lawphongpanich and Hearn [25] use conditions similar to the constraints in MINREV to formulate second-best toll pricing problems (i.e., ones in which some roads are not tollable) as mathematical programs with equilibrium constraints or MPECs (see, e.g., Luo et al. [27]).

Although it may be hard to accept politically, the assumption that every road is tollable is not technologically far-fetched. It is certainly conceivable that cars can be equipped with global positioning systems and satellites can accurately track the location of each car at all times for the purpose of tolling. The MINREV problem also has applications in the internet or other telecommunication networks (see, e.g., [7, 19, 20]). In those applications, several forms of toll pricing have been considered, and some are already incorporated into the transmission control protocol or TCP for controlling the flow of information (or packets) on every link.

As an optimization problem, MINREV is a linear program and it is natural to use commercial software such as CPLEX [8], a code well known for its speed and ability to solve large linear programs. However, Hearn and Ramana [16] report that CPLEX does not perform well on large instances of MINREV. This is apparently due to one particularly dense constraint in MINREV that requires CPLEX to perform steps that are more expensive computationally. One of our two goals in this article is to investigate whether it is advantageous to use decomposition when solving large-scale MINREV problems, especially in light of the many optimization software packages currently competing in the operations research and management science community.

Our other goal for this article is of theoretical nature. In the literature, two alternative methods have been proposed for MINREV. One is a convergent cutting plane (CP) method

Received May 2003; accepted January 2004
Correspondence to: D. W. Hearn; e-mail: hearn@ise.ufl.edu
Contract grant sponsor: NSF; contract grant numbers: DMI-9978642 and DMI-0300316
DOI 10.1002/net.20024
Published online in Wiley InterScience (www.interscience.wiley.com).

proposed by Hearn et al. [18], and the other is a combinatorial algorithm for the single origin case by Dial [9]. In his second article, Dial [10] sketches a heuristic extension of his single origin method to the multiorigin case, but no convergence analysis is provided, and there is no computational implementation. Instead, Dial [10] notes that the problem can be addressed as a linear multicommodity network flow (LMCNF) problem, a problem for which the Dantzig-Wolfe (DW) decomposition is well suited (see, e.g., Larsson and Yuan [23]). From the duality relationship between DW and Benders decomposition (also referred to as the cutting plane algorithm) in linear programming, it is intuitive to assume that the CP method for MINREV in Hearn et al. [18] and DW decomposition of the LMCNF problem are dual of each of other. However, this is not immediately evident because these algorithms are derived from two different formulations of MINREV. The CP method uses the variational inequality formulation of the user equilibrium condition, and DW decomposition works with the dual of a version of MINREV in which a dense constraint has been disaggregated. Thus, some modifications to the two algorithms are necessary to establish the desired duality relationship, and consequently, unify the treatment of large-scale MINREV problems in the literature. When compared numerically, this article shows that DW decomposition is more efficient and robust at solving MINREV problems using real-world networks from the literature.

For the remainder, Section 2 formulates the MINREV problem. Section 3 reviews and describes variants of the CP algorithm. Section 4 presents the Dantzig-Wolfe algorithm and its variations for solving the dual of MINREV. Section 5 establishes the relationships between the CP algorithm and DW decomposition. Then, implementation issues and numerical results are discussed in Sections 6 and 7, respectively, and concluding remarks follow in Section 8.

2. MINIMUM REVENUE TOLL PRICING PROBLEM

To review the formulation of MINREV, let $\mathcal{G}(\mathcal{N}, \mathcal{A})$ denote a network, where $\mathcal{N}$ and $\mathcal{A}$ are the sets of nodes and arcs, respectively. Then, let d^{pq} denote the number of vehicles traveling from node p to node q during a given time period, that is, d^{pq} denotes the travel demand for the origin-destination (OD) pair (p, q). Define a commodity as the set of travel demands that originate from the same origin or, equivalently, the set of OD pairs with the same origin node. Let k be an index for commodity and $\mathcal{K}$ be the set of all commodities. Then, $k \in \mathcal{N}$ and $\mathcal{K} \subseteq \mathcal{N}$. For each k, let $D(k)$ be the set of destination nodes of all OD pairs with node k as the origin node, i.e., $D(k) = \{q : d^{kq} > 0, \forall q \in \mathcal{N}\}$. Associated with each k is a demand vector $b^k \in R^{|\mathcal{N}|}$, where

$$b_i^k = \begin{cases} \sum_{j \in D(k)} d^{kj} & \text{if } i = k \\ -d^{ki} & \text{if } i \in D(k) \\ 0 & \text{otherwise.} \end{cases}$$

Given the above parameters, a vector $x^k \in R^{|\mathcal{A}|}$ is a feasible flow vector for commodity k if it is nonnegative and satisfies $Ax^k = b^k$, where A is the node-arc incidence matrix associated with $\mathcal{G}(\mathcal{N}, \mathcal{A})$. Let $v = \sum_{k \in \mathcal{K}} x^k$, that is, v is an aggregation of $x^k \in R^{|\mathcal{A}|}$ or an aggregate flow vector, and the set of feasible aggregate flow vectors can be expressed mathematically as

$$V = \left\{ v \middle| v = \sum_{k \in \mathcal{K}} x^k,\ Ax^k = b^k,\ x^k \geq 0,\ \forall\, k \in \mathcal{K} \right\}.$$

To predict the travel behavior in the presence of tolls, let $s_a(v)$ and β_a be the travel cost (or time) function and the toll for arc a, respectively. For a given toll vector β, the aggregate flow vector $v^*(\beta)$ is in *tolled user equilibrium* if it satisfies the following inequality:

$$(s(v^*(\beta)) + \beta)^T (v - v^*(\beta)) \geq 0, \quad \forall\, v \in V.$$

In other words, $v^*(\beta)$ solves the variational inequality problem defined by the set V and the function $s(v) + \beta$, that is, $v^*(\beta)$ solves VI$[s(v) + \beta;\ V]$. When $\beta = 0$, the above VI problem reduces to the standard (untolled) user equilibrium model in the literature (see, e.g., Florian and Hearn [12]) and $v^*(0)$ is generally referred to as the "user equilibrium solution" or, more simply, the "user solution." Another traffic assignment model in the literature is the system problem stated below:

$$\bar{v} = \operatorname{argmin}\{s(v)^T v : v \in V\}.$$

Intuitively, $\bar{v}$ (or the system solution) is an aggregate flow vector with the minimum total delay.

In Hearn and Ramana [16], the goal in toll pricing is to find β so that $v^*(\beta) = \bar{v}$. They show that a toll vector β induces this equality, or is a valid toll vector, when there exists a ρ^k for each $k \in \mathcal{K}$ such that

$$s(\bar{v}) + \beta \geq A^T \rho^k, \quad \forall\, k \in \mathcal{K}, \tag{1a}$$

$$(s(\bar{v}) + \beta)^T \bar{v} = \sum_{k \in \mathcal{K}} (b^k)^T \rho^k. \tag{1b}$$

The above system of equations ensures that $\bar{v}$ satisfies the Karush-Kuhn-Tucker (KKT) conditions associated with VI$[s(v) + \beta;\ V]$. In particular, condition (1b) is an aggregation of the complementary slackness conditions associated with the nonnegativity constraint for x^k. As shown in [16], $\beta = \nabla s(\bar{v})^T \bar{v}$, the marginal social cost pricing toll vector (see, e.g., [2]), is always valid. Under some mild conditions, the set of all possible valid toll vectors is a polyhedron and the choice of a valid toll vector depends on the (secondary) goal of the traffic planner. (The first goal is to achieve the system solution, $\bar{v}$.) To minimize the financial impact on the transportation network users, it is logical to minimize the toll revenue collected. This minimum toll revenue problem can be formulated as MINREV-LP:

$$\begin{array}{lll} \min & \bar{v}^T\beta & \\ s.t. & s(\bar{v}) + \beta & \geq A^T\rho^k,\ \forall\, k \in \mathcal{K}, \\ & (s(\bar{v}) + \beta)^T\bar{v} & = \sum_{k\in\mathcal{K}} (b^k)^T\rho^k, \\ & \beta \geq 0. & \end{array}$$

The objective function of MINREV-LP is the sum of the toll times the flow on each arc and the constraints ensure that the toll vector is valid. As stated, MINREV-LP is a linear program and only allows nonnegative tolls. However, it is simple, computationally or otherwise, to allow negative tolls, and doing so would correspond to subsidizing usage of certain links. In this study, we consider only nonnegative tolls.

3. CUTTING PLANE ALGORITHM

In this section, we briefly describe the cutting plane algorithm presented in [18] and develop a variant motivated by the analysis of DW decomposition discussed in Section 4. As formulated above, MINREV-LP uses the KKT conditions for VI[$s(v) + \beta$; V]. In lieu of these conditions, the formulation below uses the definition of VI:

$$\begin{array}{ll} \min & \bar{v}^T\beta \\ \text{s.t.} & (s(\bar{v}) + \beta)^T(u - \bar{v}) \geq 0, \quad \forall\, u \in V, \\ & \beta \geq 0. \end{array}$$

The feasible region of the above problem can be tightened by disaggregating each constraint by commodity. This yields MINREV-VI:

$$\begin{array}{ll} \min & \bar{v}^T\beta \\ \text{s.t.} & (s(\bar{v}) + \beta)^T(u^k - \bar{x}^k) \geq 0, \qquad \forall\, k \in \mathcal{K}, u^k \in V^k, \\ & \beta \geq 0. \end{array}$$

where $V^k = \{x : Ax^k = b^k, x \geq 0\}$ and $\bar{v} = \sum_{k\in\mathcal{K}} \bar{x}^k$. Because each V^k is a bounded polyhedron, it can be represented as a convex combination of its extreme points, for which there are finitely many. In particular, MINREV-VI can be stated as follows:

$$\begin{array}{ll} \min & \bar{v}^T\beta \\ \text{s.t} & (s(\bar{v}) + \beta)^T(y^k[i] - \bar{x}^k) \geq 0, \quad \forall\, k \in \mathcal{K},\ i = 1, \ldots, L_k, \\ & \beta \geq 0, \end{array}$$

where, for each k, $y^k[i]$ is an extreme point of V^k and L_k is the number of such points. Generally, it is impractical to generate all L_k extreme points *a priori* and the algorithm below generates them one at a time instead.

Cutting Plane Algorithm

STEP 0. For each $k \in \mathcal{K}$, let $y^k[0]$ be an extreme point of V^k and set $l = 1$.

STEP 1. Let β^l solve the master problem:

$$\begin{array}{ll} \text{argmin} & \bar{v}^T\beta \\ \text{s.t.} & (s(\bar{v}) + \beta)^T(y^k[i] - \bar{x}^k) \geq 0, \\ & \qquad \forall\, k \in \mathcal{K}, \quad i = 0, \ldots, (l-1), \\ & \beta \geq 0. \end{array}$$

STEP 2. For each $k \in \mathcal{K}$, solve the following subproblem:

$$y^k[l] = \text{argmin}\{(s(\bar{v}) + \beta^l)^T y | Ay = b^k, y \geq 0\}.$$

STEP 3. If $(s(\bar{v}) + \beta^l)^T(y^k[l] - \bar{x}^k) \geq 0$, $\forall k \in \mathcal{K}$, stop and β^l is an optimal toll vector. Otherwise, set $l = l + 1$ and go to Step 1.

One method for finding an extreme point in Step 0 is to solve, for each $k \in \mathcal{K}$, the subproblem in Step 2 with $\beta^l = 0$, i.e.,

$$y^k[0] = \text{argmin}\{s(\bar{v})^T y | Ay = b^k, y \geq 0\}.$$

Note that the above problem (and the one in Step 2) is a minimum cost flow problem (see, e.g., [1]) without arc capacities. Thus, it is solvable as a shortest path problem.

In Step 1, $\bar{v}^T\beta^l$ is a lower bound for the optimal toll revenue. In addition, the size of the master problem can be reduced by aggregating the constraints as follows:

$$\begin{array}{ll} \text{argmin} & \bar{v}^T\beta \\ \text{s.t.} & (s(\bar{v}) + \beta)^T(y[i] - \bar{v}) \geq 0, \quad \forall\, i = 0, \ldots, (l-1), \\ & \beta \geq 0. \end{array}$$

where $y[i] = \sum_{k\in\mathcal{K}} y^k[i]$ and $\bar{v} = \sum_{k\in\mathcal{K}} \bar{x}^k$ as before. However, the numerical results in [18] suggest that this type of constraint aggregation generates weak lower bounds for the optimal toll revenue and makes the CP algorithm converge slower than the one in Step 1.

In the constraints of the master problem in Step 1, $w^k[i] = y^k[i] - \bar{x}^k$ is a feasible direction with respect to the set V^k at the point $\bar{x}^k$. Mathematically, the set of all feasible directions at $\bar{x}^k$ is

$$\Omega^k = \{w^k : Aw^k = 0 \text{ where } w_a^k \geq 0 \text{ if } \bar{x}_a^k = 0 \text{ and } w_a^k \text{ unrestricted if } \bar{x}_a^k > 0\}.$$

Because Ω^k is unbounded, it is more convenient (see, e.g., [4]) to consider the following set instead:

$$\Omega_1^k = \{w^k : Aw^k = 0 \text{ where } 0 \leq w_a^k \leq 1 \text{ if } \bar{x}_a^k = 0 \text{ and } -1 \leq w_a^k \leq 1 \text{ if } \bar{x}_a^k > 0\}.$$

Except for the fact that the ℓ_∞ norm of the direction in Ω_1^k is at most one, the two sets of feasible directions are essentially the same. In particular, if w^k belongs to Ω^k, then $1/(\|w^k\|_\infty)\ w^k$ belongs to Ω_1^k. Conversely, if $\hat{w}^k$ belongs to Ω_1^k, then $\hat{w}^k$ also belongs to Ω^k.

Using Ω_1^k, MINREV-VI can also be stated as follows:

$$\begin{array}{ll} \min & \bar{v}^T\beta \\ \text{s.t} & (s(\bar{v}) + \beta)^T w^k \geq 0, \quad \forall\, k \in \mathcal{K}, w^k \in \Omega_1^k, \\ & \beta \geq 0. \end{array}$$

Because it is a bounded polyhedron, Ω_1^k can be represented as a convex combination of its extreme points (denoted below as θ^k) that can be generated as the algorithm progresses. This leads to the following modification of Step 2:

STEP 2′. Solve the following subproblem for each commodity k:

$$\theta^k[l] = \text{argmin}\{(s(\bar{v}) + \beta^l)^T w : w \in \Omega_1^k\}$$

Let $\alpha^k[l] = \min_{a \in \mathcal{A}} \left\{ \dfrac{-\bar{x}_a^k}{\theta_a^k[l]} : \theta_a^k[l] < 0 \right\}$ and set

$$y^k[l] = \bar{x}^k + \alpha^k[l]\theta^k[l].$$

As constructed above, $y^k[l]$ is feasible to V^k because $\bar{x}^k$ is feasible, $\theta^k[l]$ is in the null space of A, and $\alpha^k[l]$ is the largest step size without making any component of $y^k[l]$ negative. The next section shows that the above subproblem is a minimum cost circulation problem (see, e.g., [1]).

In Step 3, it is more practical to stop when $(s(\bar{v}) + \beta^l)^T(y^k[l] - \bar{x}^k) \geq -\epsilon_k$, $\forall k \in \mathcal{K}$, where ϵ_k is a small positive constant. As explained in [18], the cutting plane algorithm with or without the above modifications must converge to an optimal solution after a finite number of iterations because the algorithm generates distinct extreme points, of which there are finitely many.

4. DANTZIG-WOLFE DECOMPOSITION

As mentioned in Section 2, the second constraint in MINREV-LP is an aggregation of the complementary slackness conditions associated with the nonnegativity constraints on x^k. Thus, by disaggregating the second constraint and combining the resulting constraints with the first set, MINREV-LP can be equivalently written as P-MINIREV:

$$\begin{array}{ll} \min & \bar{v}^T\beta \\ \text{s.t} & \beta_a - \rho_j^k + \rho_i^k \geq -s_a(\bar{v}), \\ & \qquad \forall\, k \in \mathcal{K}, \quad a \in \{(i,j) : \bar{x}_{ij}^k = 0\}, \\ & \beta_a - \rho_j^k + \rho_i^k = -s_a(\bar{v}), \\ & \qquad \forall\, k \in \mathcal{K}, \quad a \in \{(i,j) : \bar{x}_{ij}^k > 0\}, \\ & \beta_a \geq 0, \quad \forall\, a. \end{array}$$

Letting w_a^k denote the dual variable associated with each constraint and using the fact that maximizing $\sum_{k,a} - s_a(\bar{v})w_a^k$ is equivalent to minimizing $\sum_{k,a} s_a(\bar{v})w_a^k$, the dual of the above problem can be written as D-MINREV:

$$\begin{array}{lll} \min & \sum_{k \in \mathcal{K}} s(\bar{v})^T w^k & \\ \text{s.t.} & \sum_{k \in \mathcal{K}} w_a^k \leq \bar{v}_a, & \forall\, a, \\ & Aw^k = 0, & \forall\, k \in \mathcal{K}, \\ & w_a^k \geq 0, & \forall\, k \in \mathcal{K}, a \in \{a : \bar{x}_a^k = 0\}, \\ & w_a^k \text{ unrestricted} & \forall\, k \in \mathcal{K}, a \in \{a : \bar{x}_a^k > 0\}. \end{array}$$

As formulated above, D-MINREV is a linear multicommodity network flow problem in which the underlying network structure may be different for different commodities. In particular, the last constraint in D-MINREV allows some arcs to be bidirectional for certain commodities. When considered together, the last three constraints in D-MINREV describe Ω^k, the set of feasible directions at $\bar{x}^k$ with respect to the set V^k. As in the previous section, consider Ω_1^k, instead of Ω^k, with extreme points denoted as $\theta^k[i]$ for $i = 1, \ldots, M_k$. Thus, for any $w^k \in \Omega^k$, there must exist an $\alpha \geq 0$ and a $\hat{w}^k \in \Omega_1^k$ (e.g., $\alpha = \|w^k\|_\infty$ and $\hat{w}^k = (1/\alpha)w^k$) such that $w^k = \alpha\hat{w}^k = \alpha \sum_{i=1}^{M_k} \gamma_i\theta^k[i]$ where $\sum_{i=1}^{M_k} \gamma_i = 1$ and $\gamma_i \geq 0$. Letting $\delta_i = \alpha\gamma_i$ implies that $w^k \in \Omega^k$ can be represented as a nonnegative combination of $\theta^k[i]$, that is, $w^k = \sum_{i=1}^{M_k} \delta_i\theta^k[i]$, where $\delta_i \geq 0$, and D-MINREV can be written as:

$$\begin{array}{ll} \min\limits_{\lambda} & \sum_{k \in \mathcal{K}} s(\bar{v})^T \left(\sum_{i=1}^{M_k} \lambda_i^k \theta^k[i] \right) \\ s.t. & \sum_{k \in \mathcal{K}} \left(\sum_{i=1}^{M_k} \lambda_i^k \theta^k[i] \right) \leq \bar{v}, \\ & \lambda_i^k \geq 0, \quad \forall\, k \in \mathcal{K}, \quad i = 1, \ldots, M_k. \end{array}$$

A feasible primal and dual pair, (λ, ζ), to D-MINREV is optimal when all reduced costs are nonnegative, that is, $(s(\bar{v}) - \zeta)^T\theta^k[i] \geq 0$, $\forall k \in \mathcal{K}$, $i = 1, \ldots, M_k$. Instead of evaluating the reduced cost of each extreme point of Ω_1^k, it is also possible to solve the following (sub)problem:

$$\min\{(s(\bar{v}) - \zeta^l)^T w : w \in \Omega_1^k\}.$$

If the optimal objective value of the subproblem is nonnegative, then the current solution (λ, ζ) is optimal and $\beta = -\zeta$ is an optimal toll vector.

As in the cutting plane algorithm, the number of extreme directions, M_k, is extremely large and it is impractical to generate them *a priori*. Instead of doing so, the DW decomposition below generates these directions one at a time.

STEP 0. Let $\theta^k[0]$ be an extreme point of Ω_1^k for all $k \in \mathcal{K}$ and set $l = 1$.

STEP 1. Solve the following master problem DW-M(l):

$$\begin{aligned} \min_{\lambda} \quad & \sum_{k\in\mathcal{K}} s(\bar{v})^T \left(\sum_{i=0}^{l-1} \lambda_i^k \theta^k[i] \right) \\ \text{s.t.} \quad & \sum_{k\in\mathcal{K}} \left(\sum_{i=0}^{l-1} \lambda_i^k \theta^k[i] \right) \le \bar{v}, \\ & \lambda_i^k \ge 0, \quad \forall\, k \in \mathcal{K}, \quad i = 0, 1, \ldots, l-1, \end{aligned}$$

and let ζ^l denote the optimal dual multiplier associated with the first constraint.

STEP 2. Solve the following subproblem for each commodity k:

$$\theta^k[l] = \operatorname{argmin}\{(s(\bar{v}) - \zeta^l)^T w : w \in \Omega_1^k\}.$$

STEP 3. If $(s(\bar{v}) - \zeta^l)^T\theta^k[l] \ge 0$, $\forall k \in \mathcal{K}$, stop and $\beta = -\zeta^l$ is an optimal toll vector. Otherwise, set $l = l + 1$ and go to Step 1.

In Step 0, $\theta^k[0]$ can be set to zero for all k. For each arc $a = (i, j)$ such that $x_a^k > 0$ in the subproblem of Step 2, w_a^k can be replaced by $[w_a^k]^+ - [w_a^k]^-$, where $0 \le [w_a^k]^+ \le 1$ and $0 \le [w_a^k]^- \le 1$. Structurally, this corresponds to adding an arc (j, i) with a capacity of one and a cost of $-(s_a(\bar{v}) - \zeta_a^l)$ to the original network. By letting $\hat{\mathcal{A}}$ represents the arc set for the expanded network and $\hat{A}^k$, d^k, and $(\hat{s}(\bar{v}) - \hat{\zeta}^l)$ represent the expanded node-arc incidence matrix, direction, and cost vectors, respectively, the subproblem in Step 2 is equivalent to the following:

$$\begin{aligned} \text{argmin} \quad & (\hat{s}(\bar{v}) - \hat{\zeta}^l)^T d \\ \text{s.t} \quad & \hat{A}^k d = 0, \\ & 0 \le d_a \le 1, \quad \forall\, a \in \hat{\mathcal{A}}. \end{aligned}$$

This problem is a minimum cost circulation problem (see, e.g., [1]) and can be solved efficiently by the network simplex algorithm.

As noted earlier, Ω^k and Ω_1^k are sets of feasible directions of V^k at the point $\bar{x}^k$. The optimal solution, $\theta^k[l]$, in Step 2 is a feasible direction with negative reduced cost, that is, an improving feasible direction. An alternative method for obtaining an improving feasible direction is to first solve the shortest path problem described in the previous section, that is, solve the following problem:

$$y^k[l] = \operatorname{argmin}\{(s(\bar{v}) - \zeta^l)^T y | Ay = b^k, y \ge 0\}.$$

Then, $\theta^k[l] = y^k[l] - \bar{x}^k$ is an improving feasible direction if $(s(\bar{v}) - \zeta^l)^T(y^k[l] - \bar{x}^k) < 0$. This leads to the following modification of Step 2 for DW decomposition:

STEP 2′. Solve the following subproblem for each commodity k:

$$y^k[l] = \operatorname{argmin}\{(s(\bar{v}) - \zeta^l)^T y | Ay = b^k, y \ge 0.\}$$

and set $\theta^k[l] = y^k[l] - \bar{x}^k$.

As before, a more practical stopping criterion in Step 3 is $(s(\bar{v}) - \zeta^l)^T\theta^k[l] \ge -\epsilon_k$, $\forall k \in \mathcal{K}$ and standard convergent arguments apply to the DW decomposition and its variant above.

An alternative to the above approach is to consider the penalty function method for MINREV-LP. In theory, the solution to the following problem solves MINREV-LP when the penalty parameter μ is sufficiently large.

$$\begin{aligned} \min \quad & \bar{v}^T\beta + \mu\left[(s(\bar{v}) + \beta)^T\bar{v} - \sum_{k\in\mathcal{K}} (b^k)^T\rho^k \right] \\ s.t. \quad & s(\bar{v}) + \beta \ge A^T\rho^k, \quad \forall\, k \in \mathcal{K}, \\ & \beta \ge 0. \end{aligned}$$

After a constant term has been factored out, the dual of the above problem reduces to the linear multicommodity network flow problem with joint arc capacities found in Marcotte and Savard [28]. The DW decomposition of this new LMCNF problem with joint arc capacities yields a subproblem that generates extreme points instead of extreme directions by solving shortest path instead of minimum circulation problems. In addition, the master problem of this new LMCNF problem minimizes a linear objective function over all possible convex combinations of generated extreme points, instead of nonnegative combinations of generated extreme directions. Despite the simpler subproblem and a slightly more complicated master problem, our implementation of the DW decomposition for the new LMCNF problem using CPLEX callable libraries ([8]) did not perform as well as the original DW decomposition. Perhaps a more specialized implementation with careful fine tuning of parameters may make the new DW decomposition more competitive. However, to show the advantages of decomposition, the original DW decomposition suffices.

5. DUALITY RESULTS

The results below demonstrate the relationships between DW decomposition and the cutting plane algorithm.

Theorem 5.1. *When Step 2′ is used, the dual of master problem in the CP algorithm is the master problem in DW decomposition.*

Proof. The dual of the CP master problem in Step 1 can be written as

$$\begin{aligned} \max \quad & \sum_{i=0}^{l-1} \left(s(\bar{v})^T \sum_{k\in\mathcal{K}} \lambda_i^k (\bar{x}^k - y^k[i]) \right) \\ s.t. \quad & \sum_{i=0}^{l-1} \left(\sum_{k\in\mathcal{K}} \lambda_i^k (y^k[i] - \bar{x}^k) \right) \le \bar{v}, \\ & \lambda_i^k \ge 0, \quad \forall\, k \in \mathcal{K}, \quad i = 0, 1, \ldots, l-1. \end{aligned}$$

When Step 2′ is used in the CP algorithm, $\alpha^k[l]\theta^k[i] = y^k[i] - \bar{x}^k$, where $\alpha^k[l] > 0$. Thus, replacing $y^k[i] - \bar{x}^k$ with $\alpha^k[l]\theta^k[i]$ and letting $\hat{\lambda}_i^k = \alpha^k[l]\lambda_i^k$ yield the following problem:

$$\begin{aligned} \min \quad & \sum_{i=0}^{l-1} \left(s(\bar{v})^T \sum_{k\in\mathcal{K}} \hat{\lambda}_i^k \theta^k[i] \right) \\ s.t. \quad & \sum_{i=0}^{l-1} \left(\sum_{k\in\mathcal{K}} \hat{\lambda}_i^k \theta^k[i] \right) \le \bar{v}, \\ & \hat{\lambda}_i^k \ge 0, \quad i = 0, 1, \ldots, l-1, \end{aligned}$$

Because the above problem is the master problem of DW decomposition, the proof is complete. ∎

Using a similar argument, the following result also holds.

Theorem 5.2. *When Step 2′ is used, the dual of master problem in DW decomposition is the master of the CP algorithm.*

Based on the earlier discussion and the above results, the CP algorithm with its Step 2′ is equivalent to DW decomposition, in that the master of one is the dual of the other and their respective subproblems are the same. For the same reason, DW decomposition with its Step 2′ is also equivalent to the CP algorithm.

6. IMPLEMENTATION ISSUES

The previous section shows that certain variations of the CP algorithm and DW decomposition are equivalent. However, there are several reasons why DW decomposition is preferable, and they are described below.

The feasibility of the CP master problem is sensitive to $\bar{v}$ (and $\bar{x}^k$). Unless $\bar{v}$ solves the system problem exactly, $(s(\bar{v}) + \beta)^T(y^k[i] - \bar{x}^k)$ may be negative for all $\beta \ge 0$, thereby making the master problem infeasible. To avoid this, the master problem is often implemented with relaxed cuts as follows:

$$\begin{aligned} \beta^l = \text{argmin} \quad & \bar{v}^T\beta \\ \text{s.t.} \quad & (s(\bar{v}) + \beta)^T(y^k[i] - \bar{x}^k) \ge -\epsilon_k, \\ & \forall\, k \in \mathcal{K}, \quad i = 0, \ldots, (l-1), \\ & \beta \ge 0 \end{aligned}$$

where ϵ_k is a small positive constant. However, the proper choice of ϵ_k is difficult to determine. An ϵ_k that is too small may still cause the master problem to be infeasible. Large ϵ_k would enlarge the feasible region of the master problem which, in turn, leads to a weaker lower bound for the MINREV-LP problem. On the other hand, the DW master problem is always feasible, that is, $\lambda_i^k = 0$ is always a feasible solution. In addition, DW decomposition is derived from D-MINREV, a problem that only depends on whether $\bar{x}_a^k$ is positive and not its magnitude. This makes DW decomposition rather robust, for it works well even when $\bar{x}_a^k$ is only approximately feasible to the system problem.

Moreover, our experiments indicate that the CP master problem is numerically unstable. On some relatively large transportation networks we tested, CPLEX terminated prematurely due to numerical difficulties. An attempt to manually scale the constraint matrix of the CP master problem by dividing each (relaxed) cut by $\|s(\bar{v})^T(y^k[i] - \bar{x}^k)\|$ did not improve the situation. For the network from Hull, Canada, the condition number of the basis matrix of the CP master problem with the manual scaling grew to 10^7 after only 10 iterations. On the other hand, the DW master problem is relatively well scaled. Each component of its master column, $\theta^k[i]$, is either 0, 1, or -1, and the condition number of the basis matrix for the same Hull network is at most 10^3.

7. COMPUTATIONAL RESULTS

Given the above discussion, we report below the results from our implementation of DW decomposition (with the original Step 2). We implemented the algorithm in C++ and used routines from the CPLEX Callable Library (Version 7.0) [8] to solve the master and subproblems. In particular, we used the network simplex routine to solve the subproblems and the primal simplex routines to solve the master problem. For the transportation networks listed in Table 1, we solved the D-MINREV problem using DW decomposition and, as a benchmark, the monolithic P-MINREV problem using CPLEX. In addition to the usual network attributes, Table 1 also lists the numbers of variables and constraints in the P-MINREV problem associated with each network. These numbers provide a more accurate measure of problem size.

In our implementation, we assumed in Step 2 of DW decomposition that $0 \le w_a^k \le 1$ in the set Ω_1^k when $\bar{x}_a^k \le 0.0001$ instead of $\bar{x}_a^k = 0$ and added $\theta^k[l]$ to the master problem in the next iteration only if $(s(\bar{v}) - \zeta^l)^T\theta^k[l] < -0.0001$. We terminated the algorithm when the reduced cost relative to the current objective function value is no smaller than -10^{-5}, i.e., $(\sum_{k\in\mathcal{K}} (s(\bar{v}) - \zeta^l)^T\theta^k[l])/$

TABLE 1. Network attributes.

Network	Links	Nodes	Commodities	Variables	Constraints
Sioux Falls [24]	76	24	24	652	1,824
Hull [13, 15]	798	501	23	12,321	18,354
Stockholm [16]	962	416	46	20,098	44,252
Winnipeg [13, 15]	2836	1052	147	157,480	416,892

$(-\bar{v}^T \zeta^l) \geq -10^{-5}$. To conserve memory when dealing with the Winnipeg network (our largest network), we did not use the advanced basis facility in CPLEX when solving either the master or subproblems.

Table 2 displays the results for DW decomposition and solving the P-MINREV problem with CPLEX using the dual simplex method, the default solver for linear programs. For each method, the table lists the objective value (toll revenue) obtained, the number of iterations, the required CPU time (in seconds) on an IBM SP2, a 300-MHz workstation with 256 MB of RAM, and the quality of the toll vector produced.

More specifically, the number of iterations for DW decomposition in Table 2 refers to the number of master iterations required to achieve the desired solution. For CPLEX, it is the number of iterations required by the dual simplex algorithm. It is also interesting to note that approximately 50% of the CPU time for DW decomposition reported in Table 2 is spent on solving the master and subproblems. The remaining 50% is used mainly by "bookkeeping" routines that, for example, obtain an optimal primal solution (CPXsolution) from the subproblem, add columns (CPXaddcols) to and obtain a dual solution (CPXgetpi) from the master problem.

The toll quality in Table 2 is a measure of the ability of the toll vector to replicate the system solution. Let β be a toll vector and $v^*(\beta)$ solve VI$[s(v) + \beta; V]$. If β is a valid toll, then $\|v^*(\beta) - \bar{v}\|$ must be zero theoretically. In transportation modeling, travel demands are "loaded" on to the network via artificial or connector arcs. In addition, arcs with low traffic volume are not critical. In our measure of toll quality, we consider only those arcs that correspond to actual roads and highways with significant traffic volume. In particular, we define for our quality measure a reference arc set, $\mathcal{A}'$, as the set $\{a | v_a^*(\beta) \geq 0.25C_a \text{ or } \bar{v}_a \geq 0.25C_a\}$, where C_a is the capacity of arc a. (In our networks, every connector arc has an infinity capacity and is excluded from $\mathcal{A}'$ by the specified conditions.) Then, the quality of toll vector β is the percentage of arcs in the reference set that have flows within 10% of the system solution, that is, the percentage of arcs such that $|v_a^*(\beta) - \bar{v}_a|/\bar{v}_a \leq 0.10$.

The results in Table 2 indicate that the solutions from DW decomposition and CPLEX are equally good. For three smaller networks [Sioux Falls (see Fig. 1), Hull and Stockholm], solving the monolithic MINREV problems by CPLEX is more efficient. However, for larger problems, solving the monolith may not be feasible and the result from the Winnipeg network illustrates this. We attempted to solve the monolithic Winnipeg problem using all three routines (primal simplex, dual simplex, and barrier method) for solving linear programs in CPLEX without success on our computer. On the other hand, DW decomposition is able to provide a good solution with a toll revenue of 85,186.70 after 9,491.41 CPU seconds, approximately 2.6 hours.

Of course the specific results given are platform dependent. To illustrate, we modeled the monolithic P-MINREV problem for Winnipeg using GAMS [11] and solved it with a newer CPLEX (Version 8.0) on a computer with two times as much memory. This implementation yielded a solution with a toll revenue of 90,094.71 after 12,035.78 CPU seconds (or approximately 3.3 hours). This illustrates that as computers and software improve (see Bixby [6]) the problem size for which decomposition will be necessary will increase. At the same time, as network planners seek greater detail, the size of models will continue to increase. Some

TABLE 2. Numerical results for the four networks.

	Dantzig-Wolfe decomposition				CPLEX			
Networks	Toll rev.	Num. iter.	CPU (sec.)	Toll qual.	Opt. rev.	Num. iter.	CPU (sec.)	Toll qual.
Sioux Falls	20.67	7	0.49	100%	20.67	77	0.12	100%
Hull	3,462.82	15	5.98	100%	3,464.67	2298	0.97	98.7%
Stockholm	1,850,974.05	40	116.02	98.7%	1,859,824.54	8090	25.72	99.3%
Winnipeg	85,186.70[a]	68	9,491.41	94.1%	N/A[b]			

[a] Due to its size, we terminated DW decomposition for Winnipeg when the relative reduced cost is at least -5.0×10^{-4} instead of -10^{-5}.

[b] The primal simplex, dual simplex, and barrier routines in CPLEX terminate without obtaining an initial feasible solution because of insufficient memory.

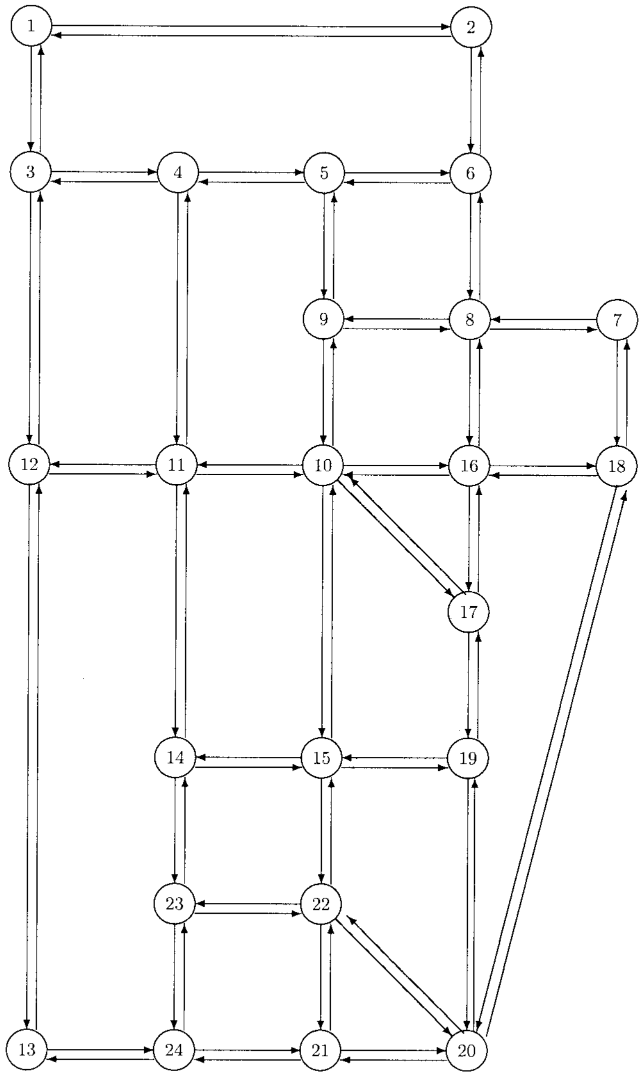

FIG. 1. Sioux Falls network.

urban planning models already have tens of thousands of links (e.g., the Chicago regional network in Bar-Gera [3] has approximately 40,000 links) and telecommunication networks are even larger (see Beinstock [5]). Our results show that DW decomposition is an effective approach once the model size reaches the point where the monolithic problem cannot be solved on the chosen computer platform.

8. CONCLUSIONS

In this article, we investigate the theoretical and numerical aspects of applying DW decomposition to the dual formulation of the MINREV problem. We develop variants of DW decomposition and the CP algorithm, and show that some of these variants are equivalent. We also examine the DW master and subproblems and identify the variant most suitable for implementation. Depending on the computer hardware and software chosen, solving the monolithic MINREV problem using CPLEX is more efficient for small- to medium-sized problems. Once the monolithic problem cannot be solved or takes excessive time, our experiments with real transportation networks indicate that, between the two methods presented in this article, DW decomposition is a more robust and effective method to employ.

REFERENCES

[1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin, Network flows: Theory, algorithm and applications, Prentice Hall, Englewood Cliffs, NJ, 1993.

[2] R. Arnott and K. Small, The economics of traffic congestion, Am Sci 82 (1994), 446–455.

[3] H. Bar-Gera, Origin-based algorithm for the traffic assignment problem, Transport Sci 36 (2002), 398–417.

[4] M.S. Bazaraa, J.J. Jarvis, and H.D. Sherali, Linear programming and network flows, Wiley, New York, 1990.

[5] D. Beinstock, Potential function methods for approximately solving linear programming problems: Theory and practice, CORE Lecture Series, 2001.

[6] R. Bixby, Solving real-world linear programs: A decade and more of progresses, Operat Res 50 (2002), 3–15.

[7] R. Cole, Y. Dodis, and T. Roughgarden, Pricing network edges for heterogeneous selfish users, STOC'03, June 9–11, San Diego, CA, 2003.

[8] CPLEX, CPLEX Optimization, Inc., Incline Village, NV, 1996.

[9] R. Dial, Minimal revenue congestion pricing part I: A fast algorithm for the single-origin case, Transport Res B 33 (1999), 189–202.

[10] R. Dial, Minimal revenue congestion pricing part II: An efficient algorithm for the general case, Transport Res B 34 (2000), 645–665.

[11] GAMS, General Algebraic Modeling System, GAMS Development Corporation, 1995.

[12] M. Florian and D. Hearn, "Network equilibrium and pricing," Handbook of transportation science, 2nd edition, R.W. Hall (Editor), Kluwer Academic Publisher, Norwell, MA, 2003, pp. 373–412.

[13] M. Florian, J. Guélat, and H. Spiess, An efficient implementation of the PARTAN variant of the linear approximation method for the network equilibrium problem, Networks 17 (1987), 319–339.

[14] P. Ferrari, Road pricing and network equilibrium, Transport Res B 29 (1995), 357–372.

[15] D. Hearn, S. Lawphongpanich, and J. Ventura, Restricted simplicial decomposition: Computation and extensions, Math Program Study 31 (1987), 99–118.

[16] D. Hearn and M. Ramana, "Solving congestion toll pricing models," Equilibrium and advanced transportation modeling, P. Marcotte and S. Nguyen (Editors), Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998, pp. 109–124.

[17] D. Hearn and M. Yildirim, "A toll pricing framework for traffic assignment problems with elastic demand," Current trends in transportation and network analysis papers in honor of Michael Florian, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2001.

[18] D. Hearn, M. Yildirim, M. Ramana, and L. Bai, Computational methods for congestion toll pricing models, IEEE Intell Transport Syst Proc, 2001, pp. 257–262.

[19] R. Johari, Mathematical modeling and control of internet congestion, SIAM News 33 (2000), March.

[20] F.P. Kelly, A.K. Maulloo, and D.H.K. Tan, Rate control in communication networks: Shadow prices, proportional fairness, and stability, J Operat Res Soc 49 (1998), 237–252.

[21] M. Labbe, P. Marcotte, and G. Sarvard, A bilevel model of taxation and its application to optimal highway pricing, Manage Sci 44 (1998), 1608–1622.

[22] T. Larsson and M. Patriksson, "Side constrained traffic equilibrium models—Traffic management through link tolls," Equilibrium and advanced transportation modeling, P. Marcotte and S. Nguyen (Editors), Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998, pp. 125–151.

[23] T. Larsson and D. Yuan, An augmented lagrangian algorithm for large scale multicommodity routing, Technical Report, LiTH-MAT-R-2000-12, Linkopings University, Linkopings, Sweden, 2000.

[24] L.J. LeBlanc, E.K. Morlok, and W.P. Pierskalla, An efficient approach to solving the road network equilibrium traffic assignment problem, Transport Res 9 (1975), 309–318.

[25] S. Lawphongpanich and D. Hearn, On the second-best toll pricing problem, Mathematical Programming B, (to appear).

[26] L.N. Liu and J.F. McDonald, Economic efficiency of second-best congestion pricing schemes in urban highway systems, Transport Res B 33 (1999), 157–188.

[27] Z.-Q. Luo, J.-S. Pang, and D. Ralph, Mathematical programs with equilibrium constraints, Cambridge University Press, New York, 1996.

[28] P. Marcotte and G. Sarvard, "A bilevel programming approach to optimal price setting," Decision and control in management sciences: Essays in honor of Alain Haurie, G. Zaccour (Editor), Kluwer Academic Publishers, Dordrecht, The Netherlands, 2001.

[29] J.F. McDonald, Urban highway congestion: An analysis of second-best tolls, Transportation 22 (1995), 353–369.

[30] The Review of Charging Options for London (ROCOL) Working Group (T. Eugenie, Chairman), Road charging options for London: A technical assessment, The Stationery Office, London, England, 2000.

[31] E. Verhoef, Second-best congestion pricing in general networks: Algorithms for finding second-best optimal toll levels and toll points, Transport Res B 36 (2002), 707–729.

[32] E. Verhoef, Second-best congestion pricing in general static transportation networks with elastic demands, Regional Sci and Urban Econom 32 (2002), 281–310.

[33] E. Verhoef, P. Nijkamp, and P. Rietveld, Second-best congestion pricing: The case of an untolled alternative, J Urban Econom 40 (1996), 279–302.