

A New Look at the Multiclass Network Equilibrium Problem

Patrice Marcotte

DIRO, Université de Montréal, C.P. 6128, Succursale Centre-Ville, Montréal, Canada H3C 3J7, patrice.marcotte@umontreal.ca

Laura Wynter

IBM Research, P.O. Box 704, Yorktown Heights, New York 10598, lwynter@us.ibm.com

The multiclass network equilibrium problem is expressed in general as a nonmonotone, asymmetric, variational inequality problem. We show that in spite of the nonmonotonicity of the cost operator, the problem may actually satisfy a weaker property, induced by the hierarchical nature of the travel cost interactions. This property allows a natural decomposition approach, not otherwise available, that admits provably convergent algorithms. We present one such algorithm, easily implementable using a solver for the single-class network equilibrium problem, together with a convergence proof.

Key words: multiclass traffic network equilibrium; nested monotonicity; variational inequalities *History*: Received: June 2002; revision received: November 2002; accepted: January 2003.

Introduction

The multiclass network equilibrium problem is a natural and important extension of the classic network equilibrium problem. While the latter can be expressed as a convex optimization program under the relatively mild assumption of separable and increasing travel cost functions, the former admits no such formulation in general. Instead, the multiclass problem is expressed mathematically as a variational inequality problem as introduced by Smith (1979) and Dafermos (1980). Indeed, in the case of a multiclass equilibrium problem, the cost operator for each class depends on the flow of more than one traffic class, and these dependencies are generally asymmetric. As a result, the Jacobian matrix of the overall cost operator, whenever it exists, is itself asymmetric. When this occurs, the variational inequality defining the Wardrop equilibrium conditions may not be computed by solving the first-order optimality conditions of an equivalent optimization problem. That is, the overall cost operator is not the gradient of another function defined over the entire feasible region, and there is no analogous "Beckmann transformation" into an equivalent convex optimization problem.

Note that this definition of a multiclass network encompasses the multimodal networks found often in the transportation literature, as well as multiple quality of service (QoS) networks in the telecommunications literature. The special case of multiple user classes that are distinguished only by their values of time (rather than through class-specific delay functions) can often be simplified in ways that are not treated here (see Leurent 1994 or Marcotte 1999 for more details on those models).

While the multiclass network equilibrium problem cannot, in general, be solved using algorithms developed for the separable, single-class network equilibrium problem (Hammond 1984), algorithmic approaches specifically adapted to variational inequalities can, in principle, be applied. Methods for solving variational inequality problems are the focus of a large body of research in the nonlinear programming community (see Patriksson 1999 or Harker and Pang 1988 for a summary of resolution methods) when the delay mapping is monotone.

A popular strategy for solving the multiclass network equilibrium problem is the so-called *diagonalization method* (respectively, triangulation method), which mimics the Jacobi (respectively, Gauss-Seidel) decomposition approach used for solving systems of equations (Florian 1977, Harker 1988, Mahmassani and Mouskos 1998). The idea behind the method is to fix flows for all but one group of variables, and to iteratively solve a sequence of separable (singleclass) subproblems. Note that classes may represent genuine user classes, each interacting on the network links, or models of junctions in which interactions occur across links.

Each single-class equilibrium can then be obtained by applying methods of convex programming, such as the widely used Frank-Wolfe linear approximation algorithm, or variants thereof. The Jacobi approach leads to solving, at iteration k, a sequence of m problems of the form

$$\min_{x_i \in \mathcal{X}_i} \sum_{a \in \mathcal{A}} \int_0^{x_{ia}} t_{ia}(x_{1a}^k, x_{2a}^k, \dots, x_{i-1,a}^k, s, x_{i+1,a}^k, \dots, x_{ma}^k) \, ds,$$
(1)

where \mathcal{A} denotes the set of *n* arcs in the network, \mathcal{X}_i is the set of feasible flows for class *i*, the travel cost function is *t*, and the model includes *m* user classes, or modes. The Jacobi iteration lends itself naturally to parallelism. Alternatively, the Gauss-Seidel approach, which makes use of the latest vector updates, leads to subproblems of the form

$$\min_{x_i \in \mathcal{X}_i} \sum_{a \in \mathcal{A}} \int_0^{x_{ia}} t_{ia}(x_{1a}^{k+1}, x_{2a}^{k+1}, x_{i-1, a}^{k+1}, s, x_{i+1, a}^k, \dots, x_{ma}^k) \, ds.$$
(2)

These decomposition methods, as is the case for the variational inequality algorithms alluded to above, are provably convergent under relatively stringent conditions on the monotonicity property of the travel cost operator. The conditions take the shape of some form of monotonicity of the mapping t over the set $\mathcal{X} = \prod_i \mathcal{X}_i$, where the required form of monotonicity (quasi, strict, strong, pseudo, etc.) depends on the algorithm used. Several popular algorithms require strong monotonicity, although variants exist that allow the equilibrium problem to be defined in terms of monotone or pseudomonotone operators. In the case of the diagonalization method, Ahn and Hogan (1973), Florian and Spiess (1982), and Pang and Chan (1982) (see also Dupuis and Darveau 1986) give local convergence results based on the following two assumptions on the travel cost functions.

• The cost mapping *t* is strongly monotone and (Fréchet) differentiable in the neighborhood of an equilibrium solution *x*^{*}.

• $||I - D[\nabla t(x^*)]^{-1/2}BD[\nabla t(x^*)]^{-1/2}||^2 < 1$,

where *D* refers to the diagonal part of the matrix, $t(x^*)$ is the travel time function evaluated at the equilibrium solution x^* , $B = \nabla t(x^*) - D[\nabla t(x^*)]$, and $\|\cdot\|$ represents the matrix norm given by $\|M\|_D = \|D^{1/2}MD^{-1/2}\|_2$. These two conditions ensure that the diagonalization algorithm converges to a locally unique solution x^* (Florian and Spiess 1982). Global convergence requires that these conditions be satisfied over the entire domain \mathcal{X} .

The second condition above implies that $\nabla t(x^*)$ is positive definite. A sufficient condition for positive definiteness that is, in some cases, easy to check for is that $\nabla t(x^*)$ be both row and column diagonally dominant; that is, letting $M = \nabla t(x^*)$

$$|M_{ii}| > \max\left\{\sum_{i \neq j} |M_{ij}|, \sum_{i \neq j} |M_{ji}|\right\} \quad \forall i, \ j = 1, \dots, |\mathcal{A}|.$$
(3)

Stated in words, the Jacobian of a differentiable mapping t will be diagonally dominant when the travel delay of a class is influenced by its own traffic much more than by flows of other classes.

It is clear that if a model involves more than one class of traffic, then these interactions will not be negligible. (Otherwise, a separable approximation of the multiclass model would have been sufficient in the first place.) Numerical experiments have confirmed the nonmonotonicity of the travel cost operator in practice. Indeed, it is often the case (see Toint and Wynter 1996 for one such example) that the cost operator in a multiclass network equilibrium problem is almost never monotone. This is not surprising: the interactions across classes on links (or across links) are generally significant; otherwise, a multiclass model would not have been needed at all, and a series of single-class approximations would suffice. When, therefore, cross-class interactions are strong, the corresponding Jacobian of the cost mapping (if it exists) will not be diagonally dominant and, in general, the mapping itself will not be monotone. In this context, the majority of algorithms used to solve multiclass problems are no longer provably convergent and, indeed, it will be shown here that such algorithms may frequently converge to points that are not equilibria at all.

In this paper, we examine the nature of the cost operator in a multiclass network equilibrium problem. We illustrate in §2 the behavior that can be exhibited by these algorithms when the monotonicity requirement is not met; that is, without the necessary monotonicity property, algorithms may converge to nonequilibrium points. Next, we show that a different property, strictly weaker than monotonicity, does hold in many cases, and provides a very elegant interpretation of the interactions across variables in the multivariate travel cost functions. A characterization of this property is provided in §3. Of significant importance is that this property has been shown to admit provably convergent algorithms. Such a class of algorithms is described in §4. Section 5 provides a new characterization of the property, slightly weaker still than the original definition, and that applies in a variety of practical situations. Then, §6 concludes with some suggestions for further work on this topic.

1. Drawbacks with Algorithms Requiring Monotonicity

Let us first recall the basic definition of network equilibrium. Let $G = (\mathcal{N}, \mathcal{A})$, be a transportation network defined by a set of nodes \mathcal{N} , a set of arcs \mathcal{A} , and a set of origin-destination (OD) pairs $\mathcal{W} \in \mathcal{N} \times \mathcal{N}$, each OD pair $w \in \mathcal{W}$ being endowed with a fixed demand d_w that must be satisfied. As in most traffic assignment problems, the arcs have associated with them cost functions, $t_a(x)$, that increase with the own flow x_a and, possibly, other arc flows as well. An equilibrium is reached when flow is distributed over the paths of the network in such a way that all demands are satisfied, and users are assigned to shortest paths with respect to current flow-dependent delays. This model is of practical interest for predicting the sensitivity of network usage with respect to modifications in the network's infrastructure or services.

The multiclass network equilibrium problem has the added complexity that demands exist between each OD pair for more than one class of flow, resulting in vector-valued OD flows. More precisely, let us denote, for each OD pair w, by d_i^w the demand for class *i* (*i* = 1, ..., *m*) and by x_{ia}^w the flow associated with arc a, class i and OD pair w. In the following example, for each class, the form of the classspecific travel cost function is the same on both of the two links and takes the form $t_{ia}(x) = t_{ia}(x_1, x_2, \dots, x_m)$, where $x_i = (\sum_{w \in \mathcal{W}} x_{ia}^w)_{a \in \mathcal{A}}$ denotes the total number of class *i* users travelling on arc *a*. This problem is referred to as *asymmetric* when the influence of one class on another is not symmetric; that is, assuming that $t \in C^1$, for some arc *a*, some flow vector x and some classes $i \neq j$, there holds $\partial t_{ia}(x) / \partial x_{ia} \neq j$ $\partial t_{ia}(x)/\partial x_{ia}$, whenever the functions involved are differentiable.

The multiclass traffic assignment problem has been paid close attention, as numerous planning scenarios require making the distinction between different flow types on the network, such as buses versus cars, heavy versus light vehicles, guided versus unguided vehicles. The interaction between modal flows, together with the accurate modelling of intersecting flows, induces nonseparable and, in general, asymmetric cost functions.

Next, we analytically examine a simple example that illustrates the occurrence of multiple isolated equilibria. Because the set of equilibria associated with a monotone problem is convex, this implies that the cost function is not monotone. These equilibria may be stable, i.e., any "reasonable" tâtonnement process initiated in the vicinity of such an equilibrium x^* converges to x^* , or unstable. Consider the two-arc network of Figure 1 with linear cost functions identical on both arcs, and where the cost functions are







Figure 2 Three Equilibrium Solutions

given by the equations (for ease of presentation, we drop the OD index)

$$t_{1a}(x) = 1.5x_{1a} + 5x_{2a} + 30$$

$$t_{2a}(x) = 1.3x_{1a} + 2.6x_{2a} + 28,$$
 (4)

where $t_{ia}(x)$ is the travel time and x_{ia} the flow of class i on arc a, a = 1, 2. Let d = (16, 4) be the demand vector. Then, the feasible domain \mathcal{X} consists of those x_{ia} for which $\sum_{a \in \{1, 2\}} x_{1a} = 16$, $\sum_{a \in \{1, 2\}} x_{2a} = 4$, and $x_{ia} \ge 0$ for all i, $a \in \{1, 2\}$. In this example, it turns out that there are three equilibrium solutions, x^{*1} , x^{*2} , and x^{*3} , expressed in matrix form as

$$\begin{aligned} x^{*1} &= \begin{pmatrix} 4/3 & 44/3 \\ 4 & 0 \end{pmatrix} \qquad x^{*2} = \begin{pmatrix} 8 & 8 \\ 2 & 2 \end{pmatrix} \\ x^{*3} &= \begin{pmatrix} 44/3 & 4/3 \\ 0 & 4 \end{pmatrix}, \end{aligned}$$

where the first row of each matrix gives the amount of the first class on Arcs 1 and 2, and the second row the amount of the second class (see Figure 2).

Netter (1972) proposed a similar example, also based on a two-arc network with linear cost functions. The article is important in that it was the first to discuss the presence of multiple equilibria. Furthermore, Netter (1972) argued that the middle solution, which corresponds to the two classes fully sharing the two arcs, was unstable in the above-mentioned sense.

The costs for each of the three solutions are

$$t(x^{*1}) = \begin{pmatrix} 52 & 52\\ 40 & 47 \end{pmatrix} \qquad t(x^{*2}) = \begin{pmatrix} 52 & 52\\ 44 & 44 \end{pmatrix}$$
$$t(x^{*3}) = \begin{pmatrix} 52 & 52\\ 47 & 40 \end{pmatrix},$$

where, again, the first row gives the travel times for a vehicle of Class 1 on Arcs 1 and 2, and the second gives the travel times for Class 2. Solution 2 is given by an equal split of the two classes between the two arcs. Although the equilibrium condition is satisfied at this point, it is an unstable solution in that altering even slightly the flow of Class 2 on the network will result in vehicles of that class seeking a different,



Figure 3 Convergence to Different Solutions According to the Starting Point

lower cost, flow pattern. Consequently, vehicles of both classes will reequilibrate into a new, stable equilibrium, and Solution 2 is a flow that would probably never be observed according to the cost function used.

In Figure 3, we illustrate the solutions found by the diagonalization method for 121 different starting points in the feasible region. For example, starting point $x_{11} = 0$ and $x_{22} = 0$ corresponds to starting the algorithm at $x_{11} = 0$, $x_{12} = d_1 = 16$, $x_{21} = d_2 = 4$, and $x_{22} = 0$, so that all of the Class 1 flow is initially assigned to Arc 2 and all of the Class 2 flow is assigned to Arc 1. The algorithm is halted as soon as either the difference between aggregate flows at two successive iterations is less than 10^{-5} or the number of diagonalization iterations reaches 100.

In this example with multiple equilibria, the equilibrium solution that is reached clearly depends on the starting point. Recall that the axes of the Figure 3 are $(x, y) = (x_{11} = 0, x_{22} = 0)$. Aside from the origin, which constitutes a special case, starting the algorithm at an initial flow value along the first (respectively, last) three rows yields the equilibrium solution x^{1*} (respectively, x^{3*}). When the initial point is selected either along the line $x_{22} = 2$ or at the origin (that is, $x_{11} = 0$,

 $x_{22} = 0$), the algorithm converges to solution x^{2*} . Furthermore, when the algorithm is started around the line $x_{22} = 2$, it stops at a nonequilibrium point.

The preceding example shows that algorithms designed for monotone variational inequalities can fail to converge, or may even converge to a nonequilibrium value when applied to nonmonotone network equilibrium problems. Figure 3 illustrates the "instability" of Solution 2: The vector field that points toward the solution of the linearized subproblem (the "Frank-Wolfe" vector field when the cost mapping is a gradient) points away from the extremal Solution 2, except along the line $x_{22} = 2$, a set of measure zero.

2. A Weaker Form of Monotonicity

While monotonicity can be shown not to hold for most multiclass network equilibrium problems, we will show in this section that the notion of nested monotonicity, introduced in Cohen and Chaplais (1988), offers an elegant explanation of the process of interactions in a class multiclass problems. Moreover, nested monotone problems are amenable to provably convergent algorithms.

First, we introduce a number of definitions, notations, and assumptions. Throughout the paper, we assume that the delay mapping t is continuously differentiable on an open set containing X.

DEFINITION 2.1 (VARIATIONAL INEQUALITY PROBLEM VIP (t, \mathcal{X})). Find $x^* \in \mathcal{X}$ such that

$$\langle t(x^*), x - x^* \rangle \ge 0, \quad \forall x \in \mathcal{X}.$$
 (5)

In the multiclass network equilibrium problem, the feasible set \mathcal{X} may be expressed as the Cartesian product of component subspaces; that is,

$$x = (x_1, \dots, x_n) \in \mathcal{X}$$
(6)

$$x_i \in \mathcal{X}_i \tag{7}$$

$$\mathcal{X} = \prod_{i=1,\dots,n} \mathcal{X}_i,\tag{8}$$

where x_i represents the flow vector associated with mode *i*. This latter vector is itself decomposed into subvectors x_i^w that contain the arc flows associated with the OD indexed by *w*. If one denotes by *B* the node-arc incidence matrix of the network, one has

$$\mathcal{X}_i = \left\{ x_i \ge 0 \colon x_i = (x_i^w)_{w \in \mathcal{W}} \colon B x_i^w = b_i^w \right\}, \tag{9}$$

where

 ${\substack{w \\ il}} = \begin{cases} \text{demand for mode } i \text{ and OD pair } w \\ \text{if node } l \text{ is the origin of } w, \\ -(\text{demand for mode } i \text{ and OD pair } w) \\ \text{if node } l \text{ is the destination of } w, \\ 0 \text{ otherwise.} \end{cases}$

The delay on the arcs of the network is induced by the aggregated flow $z = Ax = \sum_{w} x^{w}$. If one denotes by $F_{ia}(z)$ the delay induced by the aggregated flow vector *z* on arc *a*, the cost operator can be expressed as

$$t(x) = A^t F(Ax). \tag{10}$$

ASSUMPTION 2.2. The following properties of Problem (5) will be assumed to hold.

(1) The subset $\mathcal{X} \subset \mathfrak{R}^n$ is closed and convex.

(2) The operator t is continuously differentiable on \mathcal{X} .

While, in general, one would like to assume the boundedness of \mathcal{X} , one can show that equilibrium flows are always acyclic, hence, the flows associated with the (fixed demand) network equilibrium may be assumed to be bounded through the addition of a nonbinding constraint on their magnitudes (such as $x_{ia} \leq \sum_{w \in \mathcal{W}} d_w$ for all links $a \in \mathcal{A}$). Note that, by using one of the many transformations available (see, e.g., Patriksson 1994), the elastic demand model, under the condition of an invertible and upperbounded demand function, could be made to fit this framework.

Using the above notation, the variational inequality (5) that defines the multiclass equilibrium is equivalent to the following system of coupled variational inequalities:

Definition 2.3 (Coupled System of Variational Inequalities). Find $x^* \in \mathcal{X}$ such that for i = 1, ..., n,

$$\langle t_i(x^*), x_i - x_i^* \rangle \ge 0, \quad \forall x_i \in \mathcal{X}_i.$$
 (11)

DEFINITION 2.4 (STRONG MONOTONICITY). An operator *t* is strongly monotone on \mathcal{X} if there exists a constant $\mu_t > 0$ such that

$$\langle t(x) - t(y), x - y \rangle \ge \mu_t ||x - y||^2, \quad \forall x, y \in \mathcal{X}.$$
 (12)

DEFINITION 2.5 (LIPSCHITZ CONTINUITY). An operator *t* is Lipschitz continuous on \mathcal{X} if there exists a constant $L_t > 0$ such that

$$||t(x) - t(y)|| \le L_t ||x - y||, \quad \forall x, y \in \mathcal{X}.$$
 (13)

To define nested monotonicity, it is necessary to further decompose the operator t and the feasible set \mathcal{X} . Let

$$x_{ (14)$$

$$x_{\geq i} := (x_i, \dots, x_n) \tag{15}$$

$$\mathcal{X}_{$$

$$\mathcal{X}_{\geq i} := \prod_{j \in \{i, n\}} \mathcal{X}_j.$$
(17)

Analogous definitions hold for $x_{>i}$ and $\mathcal{X}_{>i}$. Then, the definition of a nested monotone operator relative to the decomposition above is as follows (Cohen and Chaplais 1988).

DEFINITION 2.6. (STRONG NESTED MONOTONICITY). An operator $t: \mathcal{X} \mapsto \Re^n$ is strongly nested monotone on \mathcal{X} relative to the decomposition (7) if, and only if, the following three conditions hold.

(1) The operator $t_1(x_1, x_{>1})$ is strongly monotone in x_1 on \mathcal{X}_1 , uniformly in $x_{>1}$ on $\mathcal{X}_{>1}$. Under Assumptions 2.2, the VIP (11) involving the operator t_1 has a unique solution, parameterized by $x_{>1}$, and denoted $x_1^*(x_{>1})$.

(2) The operators $t_i(x_{<i}^*(x_{\geq i}), x_i, x_{>i})$, for i = 1, ..., n - 1, are strongly monotone in x_i over \mathcal{X}_i , uniformly in $x_{>i}$ over $\mathcal{X}_{>i}$. Each such VIP has, under Assumptions 2.2, a unique parametric solution denoted $x_i^*(x_{>i})$.

(3) The operator $t_n(x_{<n}^*(x_n), x_n)$ is strongly monotone in x_n on \mathcal{X}_n . Under Assumptions 2.2, the VIP (11) involving the operator t_n has a unique solution.

It is easy to see that every strongly monotone function is strongly nested monotone, but that the converse is not always true. Examples in Cohen and Chaplais (1988) and in the following section illustrate that a strongly nested monotone function need not even be monotone. A consequence of strong nested monotonicity is the following important property:

PROPOSITION 2.7 (COHEN AND CHAPLAIS 1988). An operator $t: \mathcal{X} \mapsto \mathfrak{N}^n$ that is strongly nested monotone on \mathcal{X} relative to the decomposition (7) has a unique solution.

PROOF. This result follows from Definition 2.3, the strong nested monotonicity property of Definition 2.6, and the fact that the recursively defined $S_i(x_{\geq i}) := \{x_i^*(x_{>i}), S_{i-1}(x_{\geq i-1})\}$ is the solution to the set of coupled variational inequalities ranging from 1 to *i*. \Box

In other words, in spite of the nonmonotonicity of these multiclass cost operators under strong nested monotonicity, the equilibrium solution is unique.

It is not always advantageous, from a computational point of view, to decouple a problem into several low-dimensional subproblems, even when this decomposition preserves the strong monotonicity property. Indeed, the resulting *n*-level hierarchical problem may be difficult to manipulate. In our case, the situation is different because decoupling the problem allows us to obtain a form of monotonicity that is not present in the original problem.

In many cases, a small number of classes is found to underly a hierarchical problem, and this is particularly so for traffic networks. Consider, for example, road networks with flows of cars, trucks, and buses. A hierarchy can be observed if, for example, the influence of trucks and buses on the delay functions of cars is strong, but not the contrary. In practice, one can often identify two distinct classes, and this type of natural hierarchy between their cost functions (e.g, one class influences the other more than vice versa).

For this reason, and also to improve the clarity of the exposition, we shall focus our attention on multiclass equilibrium problems having two interacting classes. Then, the decomposition is immediate and the definition of nested monotonicity reduces to the following:

DEFINITION 2.8 (STRONG NESTED MONOTONICITY OVER TWO SUBSPACES). The operator t is strongly nested monotone if, and only if,

(1) $t_1(x_1, x_2)$ is strongly monotone in x_1 on \mathcal{X}_1 , uniformly in x_2 on \mathcal{X}_2 .

(2) Let $x_1^*(x_2)$ solve VIP $(t_1(x_1, x_2), \mathcal{X}_1)$. Then, $t_2(x_1^*(x_2), x_2)$ is strongly monotone in x_2 over \mathcal{X}_2 .

This decomposition corresponds to a multiclass network with two classes, x_1 and x_2 . Note, however, that t_1 and t_2 are still vector functions, where dim $\mathcal{X}_1 = n_1$ and dim $\mathcal{X}_2 = n_2$. In general, both n_1 and n_2 will be equal to the number of arcs of the network times the number of OD couples, though this is not limiting for the results or methods proposed in this paper.

We close this section by giving an example of a nonmonotone multiclass network equilibrium problem that is nested monotone. Consider a network with two traffic classes, e.g., in a road transport network, one may wish to model buses and cars or heavy and light vehicles. In a telecommunications network, 1 and 2 may represent priority classes on an Asynchronous Transfer Mode or Internet Protocol network.

Recall the two-arc network of Figure 1. In the following example, suppose that the total OD demand from Node 1 to Node 2 of Class 2 (cars for instance) is equal to 20, and 10 for Class 1 (buses for instance). We define the affine cost operator t(x) = Mx + q, where $x = (x_{11}, x_{21}, x_{12}, x_{22})$ (this ordering is more convenient than the ordering $x = (x_{11}, x_{12}, x_{21}, x_{22})$, for our purpose) and

$$M = \begin{pmatrix} 2 & 3 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

We have

$$\binom{x_{12}}{x_{22}} = \binom{10}{20} - \binom{x_{11}}{x_{21}}.$$
 (18)

Let us assume that, at equilibrium, all flows are positive. This can be achieved by selecting the vector q in a suitable fashion. To check for monotonicity, and take into account the presence of the demand constraints, we compute the *reduced Jacobian* matrix

$$J = t_{.1.1}(x_1, Dx_1 + b) + t_{.1.2}(x_1, Dx_1 + b)D$$

+ $D^t t_{.2.1}(x_1, Dx_1 + b) + D^t t_{.2.2}(x_1, Dx_1 + b)D,$

where $t_{i,j}$ denotes the submatrix of *J* associated with classes *i* and *j*, i.e.,

$$t_{1.1} = t_{2.2} = \begin{pmatrix} 2 & 3 \\ 0 & 1 \end{pmatrix} \qquad t_{1.2} = t_{2.1} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$
$$D = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \qquad b = \begin{pmatrix} 10 \\ 20 \end{pmatrix}. \tag{19}$$

This yields $J = \begin{pmatrix} 4 & 6 \\ 0 & 2 \end{pmatrix}$. Since $0.5 \det(J + J^t) = -1$, the function is not monotone over the feasible set. However, the mapping is strongly nested monotone because

• for fixed Class 2 flow, the Class 1 problem is separable and strongly monotone (positive diagonal terms, null off-diagonal terms);

• Class 2 equilibrium is independent of Class 1 flow (*M* is upper triangular) and involves the identity matrix; the corresponding mapping is obviously strongly nested monotone.

Note that not all multiclass cost operators satisfy this property. For instance, Netter's (1972) example (see §2), whose solution set is disconnected and therefore nonconvex, cannot be nested monotone.

REMARK 2.9. As in the previous example, it is frequently the case that, with respect to a given class index, the cost operator is a gradient mapping, and the single-class variational inequality reduces to a convex optimization problem. However, due to asymmetric cross-class interactions, the overall variational inequality need not be reducible to a convex problem.

REMARK 2.10 (TESTING FOR STRONG NESTED MONO-TONICITY). To test for the property of nested monotonicity it is necessary, in principle, to solve for the parametric equilibrium solution of one class, in terms of the other, and then examine the cost function of the second class as a function of this parametric solution. Clearly, over a large-scale network, such a study is highly impractical, if not impossible. If, however, the cost functions on each arc are structurally similar, up to additive and/or multiplicative constants, then a test for the nested monotonicity property over a small subnetwork, such as possibly one arc of the network, offers a good approximate test.

3. An Algorithm for the Multiclass Equilibrium Problem

In this section, we describe a convergent algorithm for nested monotone problems. This is important, because multiclass network equilibrium problems, which cannot be assumed to be monotone, are likely to possess that property. The following assumption will be required throughout this section.

Assumption 3.1. The function t is Lipschitz continuous over \mathcal{X} .

LEMMA 3.2. If the operator t is strongly nested monotone, then the reaction function $x_1^*(x_2)$ and the operator $t_2(x_1^*(x_2), x_2)$ are both Lipschitz continuous with respect to x_2 .

PROOF. The strong monotonicity of $t_1(\cdot, x_2)$ and $t_2(x_1^*(x_2), x_2)$ follow from the definition of a strongly nested monotone operator. Then, the result follows from Lemma 5.1 of Cohen and Chaplais (1988).

The solution algorithm we propose is of the descent type and minimizes a merit (or "gap") function defined for a variational inequality VIP(t, X) as

$$g(x) := \max_{y \in \mathcal{X}} \phi(x) + \langle t(x) - \nabla \phi(x), x - y \rangle - \phi(y), \quad (20)$$

where ϕ is strongly convex and continuously differentiable over \mathcal{X} . In the dual-class, nested monotone case, the relevant gap function for the problem VIP(t_2 , \mathcal{X}_2) is expressed as

$$g_{2}(x_{1}^{*}(x_{2}), x_{2})$$

$$:= \max_{y \in \mathcal{X}_{2}} \phi_{2}(x_{2}) + \langle t_{2}(x_{1}^{*}(x_{2}), x_{2}) - \nabla \phi_{2}(x_{2}), x_{2} - y \rangle$$

$$- \phi_{2}(y), \qquad (21)$$

where, again, $\phi_2: \Re^{n_2} \mapsto \Re$ is strongly convex and continuously differentiable over the feasible set \mathcal{X}_2 . Let $y_2(x_1^*(x_2), x_2)$ denote the unique solution to (21). One interesting special case of the gap function (20) occurs when $\phi(x) := \frac{1}{2} ||x||_G^2$, where *G* is an a priori defined symmetric positive definite matrix. Using this particular choice of auxiliary function ϕ leads to the gap function introduced by Fukushima (1992)

$$g(x) := \max_{y \in \mathcal{X}} \langle t(x), x - y \rangle - \frac{1}{2} \langle x - y, G(x - y) \rangle, \quad (22)$$

which can be evaluated by solving the projection problem

$$\min_{y \in \mathcal{X}} \left\| y - (x - G^{-1}t(x)) \right\|_{G}^{2}.$$
 (23)

The differentiable, nonconvex optimization problem associated with (21) is given by

$$\min_{x_2 \in X_2} g_2(x_1^*(x_2), x_2).$$
(24)

Essential properties of gap functions are given below. Using Lemma 3.2, the proofs may be obtained as direct extensions of those found, for example, by Larsson and Patriksson (1994) and have, therefore, been omitted.

THEOREM 3.3. Under Assumption 3.1 and Lemma 3.2, the following properties hold.

(1) The gap function g is nonnegative and $g(x_2, x_1) = 0$ if, and only if, x^* is a solution of the variational inequality $VIP(t, \mathcal{X})$.

(2) Similarly, the gap function $g_2(x_1^*(x_2), x_2) \ge 0$ for all $x_2 \in \mathcal{X}_2$, and $g_2(x_1^*(x_2), x_2) = 0$ if, and only if, $x_2 \in \mathcal{X}_2$ solves the VIP $(t_2(x_1^*(x_2), x_2), \mathcal{X}_2)$.

(3) If $\nabla \phi(x_2)$ is Lipschitzian on \mathcal{X}_2 , then x_2^* is a stationary point of Problem (24) if, and only if, x_2^* is a global optimal solution of (24), i.e., a solution of the variational inequality VIP($t_2(x_1^*(x_2), x_2), \mathcal{X}_2)$.

THEOREM 3.4 (DIRECTIONAL DIFFERENTIABILITY OF g_2). Let $t_2 \in C^1$ and the auxiliary function $\phi_2 \in C^2$ on \mathcal{X}_2 . Let t_2 and t_1 be Lipschitz continuous in each of their arguments. Then, the implicit gap function g_2 is locally Lipschitz continuous (with Lipschitz constant denoted by L_{g_2}) and directionally differentiable.

PROOF. The assumptions and the Lemma 3.2 guarantee the Lipschitz continuity of the implicit mappings $x_2 \mapsto x_1^*(x_2)$ and $x_2 \mapsto y_2(x_2)$. Then, the result follows from, e.g., Hogan (1973). \Box

Based on the solution to (21), we introduce

$$d_2 := y_2(x_1^*(x_2), x_2) - x_2, \tag{25}$$

a descent direction for the gap function g_2 . For some positive symmetric and positive definite matrix G_2 , we focus on the special case of auxiliary function $\phi_2(x) := \frac{1}{2} ||x_x||_{G_2}^2$, which leads to the gap function

$$g_{2}(x_{1}^{*}(x_{2}), x_{2}) = \max_{y \in \mathcal{X}_{2}} \langle t(x_{1}^{*}(x_{2}), x_{2}), x_{2} - y \rangle - \frac{1}{2} \langle x_{2} - y, G_{2}(x_{2} - y) \rangle.$$
(26)

Before showing that d_2 is a descent direction, a technical definition is required.

DEFINITION 3.5 (GRAPHICAL DERIVATIVE (ROCKA-FELLAR AND WETS 1998, DEF. 8.33). The graphical derivative of a mapping $T: \mathfrak{N}^n \rightrightarrows \mathfrak{N}^n$ at a point $x \in$ dom(*T*) and for $v \in T(x)$ is the mapping $DT: \mathfrak{N}^n \mapsto$ \mathfrak{N}^n defined by

$$z \in DT(x \mid v)(d) \longleftrightarrow (d, z) \in \mathcal{T}_{gphT}(x, v), \qquad (27)$$

where \mathcal{T}_{gphT} is the tangent cone to the graph of the mapping *T* at (x, v). When *T* is single valued at *x*; that is, $T(x) = \{v\}$, then the notation simplifies to DT(x)(d).

DEFINITION 3.6 (SEMIDERIVATIVE (ROCKAFELLAR AND WETS 1998, Ex. 8.43)). The semiderivative of a mapping $T: \mathfrak{R}^n \rightrightarrows \mathfrak{R}^n$ at a point $x \in \text{dom}(T)$ for some $v \in T(x)$ and \overline{w} is the limit

$$\lim_{0 \to 0, w \to \bar{w}} \frac{T(x + \tau w) - v}{\tau}, \qquad (28)$$

whenever the limit exists. If it exists for each vector $\bar{w} \in \Re^x$, then *T* is semidifferentiable at *x* for *v*.

τ

REMARK 3.7. Under the assumption of strong nested monotonicity, and following Proposition 2.7, the mappings t_1 and t_2 are single valued.

Rockafellar and Wets (1998) provide properties and illustrations of graphical derivatives, which generalize subgradients and directional derivatives to the case of vector functions and multivalued mappings. We are concerned with one particular property of graphical derivatives presented below. THEOREM 3.8 (ROCKAFELLAR AND WETS 1998, TH. 12.65). Consider a maximal monotone mapping $T: \mathfrak{R}^n \rightrightarrows \mathfrak{R}^n$, any point $x \in dom(T)$, and $v \in T(x)$. If T is continuous and single valued at x, then the mapping $DT(x \mid v)(d) = DT(x)(d)$ is also maximal monotone and everywhere continuous. Also, the following properties are equivalent:

(1) T is semidifferentiable at x for v;

(2) DT(x | v)(d) is single valued.

The next result, due to Marcotte and Zhu (1993) in the context of variational inequalities involving continuous but nondifferentiable operators, establishes a descent property for a certain direction. It is given a new, shorter proof that will be extended in the next section.

THEOREM 3.9. The vector $d_2 := y_2(x_1^*(x_2), x_2) - x_2$ satisfies the following descent condition:

$$g'_2(x_1^*(x_2), x_2; d_2) \le -\mu_{t_2} \|d_2\|^2$$
 (29)

for every $x_2 \in \mathcal{X}_2$, where μ_{t_2} is the strong monotonicity constant of t_2 .

PROOF. From graphical derivative calculus (Rock-afellar and Wets 1998, Ch. 8.G.), one has

$$g_{2}'(x_{1}^{*}(x_{2}), x_{2}; d_{2}) = \sup_{z \in \{y_{2}(x_{1}^{*}(x_{2}), x_{2})\}} \{ \langle t_{2}(x_{1}^{*}(x_{2}), x_{2}) - Dt(x_{1}^{*}(x_{2}), x_{2})(z - x_{2}) + G_{2}(z - x_{2}), d_{2} \rangle \}.$$
(30)

Because the supremum is uniquely attained then, letting $y_2(x_2) := y_2(x_1^*(x_2), x_2)$, one can express (30) as

$$g'_{2}(x_{1}^{*}(x_{2}), x_{2}; d_{2})$$

$$= \langle t_{2}(x_{1}^{*}(x_{2}), x_{2}) + G_{2}(y_{2}(x_{2}) - x_{2}), y_{2}(x_{2}) - x_{2} \rangle$$

$$- \langle Dt_{2}(x_{1}^{*}(x_{2}), x_{2})(y_{2}(x_{2}) - x_{2}), y_{2}(x_{2}) - x_{2} \rangle. \quad (31)$$

By the optimality of $y_2(x_2)$ for the subproblem, we have that

$$\langle x - G_2^{-1} t_2(x_1^*(x_2), x_2) - y_2(x_2), G_2(z - y_2(x_2)) \rangle \le 0$$
 (32)

for every $z \in \mathcal{X}$, which can be rewritten as

$$\langle t_2(x_1^*(x_2), x_2) + G_2(y_2(x_2) - x_2), y_2(x_2) - x_2 \rangle \le 0.$$
 (33)

Combining (31) and (33), we obtain

$$g'_{2}(x_{1}^{*}(x_{2}), x_{2}; d_{2})$$

$$\leq \langle -Dt_{2}(x_{1}^{*}(x_{2}), x_{2})(y_{2}(x_{2}) - x_{2}), y_{2}(x_{2}) - x_{2} \rangle, \quad (34)$$

$$\leq -\mu_{t_{2}} \|d_{2}\|^{2}, \quad (35)$$

where the final inequality follows from the monotonicity property of Dt_2 when Dt_2 is semidifferentiable, which, in turn, holds when the mapping t_2 is maximal monotone, continuous, and single valued, according to Theorem 3.8. Note that, since the set $\{y_2(x_1^*(x_2), x_2)\}$ is a singleton and $t_2(x_1^*(x_2), x_2)$ is strongly monotone, the function $t_2(x_1^*(x_2), x_2)$ is maximal monotone and continuous. \Box

A mixed descent algorithm for solving the coupled pair of optimization programs is provided next. For simplicity, we continue to assume that the auxiliary function chosen is $\phi_2(x) := \frac{1}{2} ||x_2||_{G_2}^2$.

3.1. Mixed Descent Method

(1) Initialization: Obtain an initial feasible pair of flow vectors (x_1^0, x_2^0) . Set the iteration counter *k* to zero.

(2) Solve a parametric, strongly monotone optimization problem to find $x_1^*(x_2^k) \in \mathcal{X}_1$.

(3) Given the solution $x_1^*(x_2^k)$, perform the update $x_2^{k+1} := x_2^t + s_2^k d_2^k$, where d_2^k is the search direction, given by

$$d_2 := y_2(x_2) - x_2 \tag{36}$$

$$= \operatorname{Proj}_{x_2, G_2}(x_2 - G_2^{-1}t_2(x_1^*(x_2), x_2)) - x_2, \quad (37)$$

and s_2^k is a step size obtained through an exact line search, with respect to the gap function, along direction d_2^k .

(4) If some stopping criterion is not met, then set k = k + 1 and return to Step 2.

This algorithm is referred to as a *mixed* descent algorithm since it combines a convex optimization problem for obtaining the optimal response $x_1^*(x_2)$ with a descent procedure for updating x_2 .

REMARK 3.10. The mixed descent method is a straightforward, convergent algorithm for the multiclass network equilibrium problem, and has the further advantage that it requires little additional programming if a solver for the single-class network equilibrium problem is available.

THEOREM 3.11. Under the stated assumptions, starting from any feasible point (x_1^0, x_2^0) , the mixed descent algorithm generates a well-defined and bounded sequence $\{(x_1^k, x_2^k)\}$ for which every cluster point of the sequence $\{x_2^k\}$ is the solution to $VIP(t_2(x_1^*(x_2), x_2), \mathcal{X}_2)$. Furthermore, the entire sequence $\{(x_1^k, x_2^k)\}$ converges to the unique solution of the $VIP((t_1(x_1, x_2), t_2(x_1, x_2)), \mathcal{X}_1 \times \mathcal{X}_2)$, i.e., $VIP(t, \mathcal{X})$.

PROOF. Because the feasible set \mathcal{X}_2 is compact, the iterates $\{x_2^k\}$ remain in a compact subset of $\mathfrak{N}_+^{n_2}$. The closedness of the algorithmic map follows from the use of an exact line search, along with the closedness of the implicit mappings $x_2 \mapsto y_2(x_2)$ and $x_2 \mapsto x_1^*(x_2)$, where the latter follows from the continuity of the mappings t_2 and t_1 . The descent property $g_2(x_1^*(x_2^{k+1}), x_2^{k+1}) < g_2(x_1^*(x_2^k), x_2^k)$ is a direct consequence of Theorem 3.9 and the use of an exact line search. The desired result then follows from standard convergence theory. \Box

4. A Weaker Assumption: Partial Nested Monotonicity

The previous results have been obtained under strong assumptions that each nested problem be *strongly* monotone. However, such a condition is unlikely to hold in practice, as the solution set is not a singleton, even in the presence of a single mode. Indeed, the decomposition of the arc flow vector x into OD flow vectors x^k is not, in general, unique. To circumvent that problem, we introduce a slightly weaker notion of nested monotonicity and show that it preserves the descent property of the algorithm. For simplicity of notation, as before, we limit ourselves to a dual-class problem. We recall that Ax is the arc flow vector corresponding to OD flows x, that t assumes the special form $t = A^t F(Ax_1, Ax_2)$ and that the feasible set is separable, i.e., $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2$.

DEFINITION 4.1 (PARTIAL NESTED MONOTONICITY). The operator t is partially nested monotone if, and only if, the following two conditions hold.

(1) The operator $F_1(y_1, y_2)$ is strongly monotone in y_1 , uniformly in y_2 .

(2) The operator $F_2(Ax_1^*(y_2), y_2)$ is strongly monotone with respect to y_2 .

The first part of the definition specifies that, for fixed arc flow vector y_1 , the delay mapping is strongly monotone with respect to the arc flow vector y_1 . The second part asserts that the nested operator t_2 is strongly monotone with respect to arc flows of Class 2. The next results adapt Lemma 3.2 and Theorem 3.9 to partially nested monotone operators.

LEMMA 4.2. If t is partially nested monotone, then the reaction function x^* and the operator $t_2(x_1^*(x_2), x_2)$ are both Lipschitz continuous.

PROOF. Let x_2^1 and x_2^2 be two feasible OD flow vectors. By construction, we have

$$|A^{t}F_{2}(Ax_{1}^{*}(x_{2}^{1}), Ax_{2}^{1}), x_{1}^{*}(x_{2}^{1}) - y\rangle \leq 0 \quad \forall y \in \mathcal{X}_{1}.$$
 (38)

Setting $y = x_1^*(x_2^2)$, we obtain

$$\langle A^t F_2(Ax_1^*(x_2^1), Ax_2^1), x_1^*(x_2^1) - x_1^*(x_2^2) \rangle \le 0.$$
 (39)

Symmetrically, we have

$$\langle A^t F_2(Ax_1^*(x_2^2), Ax_2^2), x_1^*(x_2^2) - x_1^*(x_2^1) \rangle \le 0.$$
 (40)

Let L_2 denote the Lipschitz constant of the mapping F_2 , μ_2 its modulus of strong monotonicity with respect to its first variable (index *b*) and set $M = \max_{x \in \mathcal{X}} ||F_2(Ax)||$, the maximum of the continuous function $F_2 \circ A$ over the compact set \mathcal{X} . From (40), we obtain the relationships

$$\langle F_2(Ax_1^*(x_2^2), Ax_2^2), Ax_1^*(x_2^2) - Ax_1^*(x_2^1) \rangle = \langle F_2(Ax_1^*(x_2^1), Ax_2^2), Ax_1^*(x_2^2) - Ax_1^*(x_2^1) \rangle + \langle F_2(Ax_1^*(x_2^2), Ax_2^1 - Ax_2^2), Ax_1^*(x_2^2) - Ax_1^*(x_2^1) \rangle$$
(41)

$$\geq \langle F_2(Ax_1^*(x_2^1), Ax_2^2), Ax_1^*(x_2^2) - Ax_1^*(x_2^1) \rangle - ML_2 \| Ax_2^1 - Ax_2^2 \| \cdot \| Ax_1^*(x_2^2) - Ax_1^*(x_2^1) \|.$$
(42)

Now, adding (39) and (40) and making use of (42) and the strong monotonicity of the mapping F_2 with respect to x_1 , we obtain that

$$\mu_2 \|Ax_1^*(x_2^1) - Ax_1^*(x_2^2)\|^2 \leq ML_2 \|Ax_2^1 - Ax_2^2\| \cdot \|Ax_1^*(x_2^1) - Ax_1^*(x_2^2)\|,$$
(43)

which, after division by $||Ax_1^*(x_2^1) - Ax_1^*(x_2^2)||$, yields that both the total flow vector $Ax_1^*(x_2)$ and the operator $t_2(x_1^*(x_2), x_2)$ are Lipschitz continuous with respect to x_2 . \Box

LEMMA 4.3. If the operator t is partially nested monotone, then the vector $d_2 := y_2(x_1^*(x_2), x_2) - x_2$ is a descent direction for the gap function g_2 .

PROOF. Let μ_2 be the strong monotonicity modulus of the function F_2 with respect to x_2 . Following into the steps of the proof of Theorem 3.9, we can write

$$g_{2}'(x_{1}^{*}(x_{2}), x_{2}; d) \leq -\langle Dt_{2}(x_{1}^{*}(x_{2}), x_{2})(y_{2}(x_{2}) - x_{2}), y_{2}(x_{2}) - x_{2} \rangle \quad (44)$$

$$= -(A^{t}DF_{2}(Ax_{1}^{*}(x_{2}), Ax_{2}))$$

$$\cdot A(y_2(x_2) - x_2), y_2(x_2) - x_2)$$
(45)

$$-\mu_2 \|A(y_2(x_2) - x_2)\|^2.$$
(46)

By construction of $y_2(x_2)$, the gap function is negative, unless x_2 is solution of the variational inequality, of course, in which case it is zero. Therefore,

 \leq

$$\langle t_2(x_1^*(x_2), x_2), y_2(x_2) - x_2 \rangle$$

+ $\frac{1}{2} \langle y_2(x_2) - x_2, G_2(y_2(x_2) - x_2) \rangle < 0.$ (47)

Since G_2 is positive definite, the second term is nonnegative and the first must be negative. Consequently,

$$\langle F_2(Ax_1^*(x_2), Ax_2), A(y_2(x_2) - x_2) \rangle < 0.$$
 (48)

This implies that $A(y_2(x_2) - x_2)$ is nonzero and, according to (46), that the direction *d* is a descent direction with respect to the gap function. \Box

COROLLARY 4.4. Under the assumption of partial nested monotonicity, the mixed descent method generates a sequence of total flows converging to the equilibrium total flow couple $(Ax_1^*(x_2), Ax_2^*)$. Any commodity flow vector compatible with $(Ax_1^*(x_2), Ax_2^*)$ is a solution of the multiclass equilibrium problem.



Figure 4 Convergence of the Mixed Descent Method: Difference Between Successive Flow Values on Each of the Two Arcs, $x_{21}^{k+1} - x_{21}^k$ and $x_{22}^{k+1} - x_{22}^k$

5. Numerical Results

The mixed descent method was implemented in the Scilab programming environment. A projection method was used to solve the single-class subproblem of Step 2, as well as the direction finding subproblem of Step 3. The algorithm converged rapidly and in 16 iterations satisfied the convergence criterion of 10^{-8} , defined as the norm of the difference between successive flow values. Note that, as the overall variational inequality can be expressed as a fixed-point problem, the use of this convergence criterion is appropriate.

The actual differences between successive iterates, $x_2^{k+1} - x_2^k$ is illustrated in Figure 4, where the upper curve is the difference of successive flow values for class *b* on Arc 1, and the lower curve gives the difference between successive values on Arc 2. Note that



Figure 5 Convergence of the Mixed Descent Method: Norm of the Flow Differences at Successive Iterations, $\|x_2^{k+1} - x_2^k\|$ on Linear (Left) and Logarithmic (Right) Scales

the difference between successive iterates rapidly approaches zero for both arcs (one from above, and one from below). In Figure 5, the convergence criterion; that is, the norm of this difference over both arcs is presented: $||x_2^{k+1} - x_2^k||$.

6. Conclusions

This paper has provided a theoretically convergent method for solving problems belonging to an important subclass of nonmonotone, asymmetric, network equilibrium problems, under an assumption that is weaker than the traditional one. This assumption is related in a natural manner to the hierarchical structure of the problem that occurs when some classes strongly influence the delays of other classes, but not the converse. If the cost functions on all arcs of the network assume a common form, then a check of whether or not the property holds on the network may be relatively easy to perform. Not all multiclass travel cost mappings will satisfy the property. However, whenever it holds, simple and provably convergent algorithms are available. We have provided one convergent algorithm for this class of network equilibrium problems, which is easily implementable and permits the use of any existing code for solving the separable network equilibrium problem as a subroutine.

Worthwhile topics for future study are the consequences of these results within a bilevel framework; that is, when the multiclass network equilibrium problem lies at the lower level of a hierarchical optimization problem, such as is the case in optimal pricing or network design problems.

Acknowledgments

This work was carried out when the second author was at INRIA Rocquencourt, France. The project was partially sponsored by the Programme National de Recherche et d'Innovation dans les Transports Terrestres (PREDIT) of the French Transport and Research Ministries, FCAR (Québec), and NSERC (Canada). The Scilab code was provided by Dr. Pablo Lotito of INRIA. This work benefited from discussions with Prof. Guy Cohen, of ENPC and INRIA, as well as from the careful reading and valuable comments of several referees and the Associate Editor.

References

- Ahn, B. H., W. W. Hogan. 1982. On convergence of the PIES algorithm for computing equilibria. Oper. Res. 30 281–300.
- Cohen, G., F. Chaplais. 1988. Nested monotony for variational inequalities over product of spaces and convergence of iterative algorithms. J. Optim. Theory Appl. 59 369–390.
- Dafermos, S. 1980. Traffic equilibrium and variational inequalities. *Transportation Sci.* 14 42–54.
- Dupuis, C., J.-M. Darveau. 1986. The convergence conditions of diagonalization and projection methods for fixed demand asymmetric network equilibrium problems. *Oper. Res. Lett.* 5 149–155.

- Florian, M. 1977. A traffic equilibrium model of travel by car and public transit modes. *Transportation Sci.* 11 166–179.
- Florian, M., H. Spiess. 1982. The convergence of diagonalisation algorithms for asymmetric network equilibrium problems. *Transportation Res.* 16B 477–483.
- Fukushima, M. 1992. Equivalent optimization problems and descent methods for asymmetric variational inequality problems. *Math. Programming* 53 99–110.
- Hammond, J. H. 1984. Solving asymmetric variational inequality problems and systems of equations with generalized nonlinear programming algorithms. Doctoral dissertation, Department of Mathematics, Massachusetts Institute of Technology, MA.
- Harker, P. T. 1988. Accelerating the convergence of the diagonalization and projection algorithms for finite-dimensional variational inequalities. *Math. Programming* **41** 29–59.
- Harker, P. T., J. S. Pang. 1988. Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms, and applications. *Math. Programming* 48 161–220.
- Hogan, W. W. 1973. Point-to-set maps in mathematical programming. SIAM Rev. 15 591-603.
- Larsson, T., M. Patriksson. 1994. A class of gap functions for variational inequalities. *Math. Programming* 64 53–79.
- Leurent, F. 1994. Cost versus time equilibrium over a network. *Eur. J. Oper. Res.* **71** 205–221.
- Mahmassami, H. S., K. C. Mouskos. 1998. Some numerical results on the diagonalization algorithm for network assignment with

asymmetric interactions between cars and trucks. *Transporta*tion Res. **22B** 275–290.

- Marcotte, P. 1999. Reformulations of a bicriterion equilibrium model. M. Fukushima, L. Qi, eds. *Reformulation—Nonsmooth*, *Piecewise Smooth, Semismooth and Smoothing Methods*. Kluwer Academic, Dordrecht, The Netherlands, 269–292.
- Marcotte, P., D. L. Zhu. 1993. Modified descent methods for solving monotone variational inequalities. *Oper. Res. Lett.* 14 111–120.
- Netter, M. 1972. Affectations de trafic et tarification au coût marginal social: Critique de quelques idées admises. *Transportation Res.* 6 411–429.
- Pang, J. S., D. Chan. 1982. Iterative methods for variational and complementarity problems. *Math. Programming* 24 284–313.
- Patriksson, M. 1994. The Traffic Assignment Problem: Models and Methods. VSP, Utrecht, The Netherlands.
- Patriksson, M. 1999. Nonlinear Programming and Variational Inequality Problems—A Unified Approach, Applied Optimization. Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Rockafellar, R. T., R. J.-B. Wets. 1998. Variational Analysis. Springer, Berlin, Germany.
- Smith, M. J. 1979. The existence, uniqueness, and stability of traffic equilibria. *Transportation Res.* 15B 443–451.
- Toint, Ph., L. Wynter. 1996. Asymmetric multiclass traffic assignment: A coherent formulation. J.-B. Lesort, ed. Proc. 13th Internat. Sympos. Transportation Traffic Theory, Pergamon, Exeter, U.K., 237–260.