Decomposition Scheme for Continuous Network Design Problem with Asymmetric User Equilibria

Xuegang Ban, Henry X. Liu, Jiangang Lu, and Michael C. Ferris

The continuous network design problem is formulated as a mathematical program with complementarity constraints (MPCC) and a Gauss–Seidel decomposition scheme is presented for the solution of the MPCC model. The model has an upper level as a nonlinear programming problem and the lower level as a nonlinear complementarity problem. With the application of the complementarity slackness condition of the lower-level problem, the original bilevel formulation can be converted into a single-level nonlinear programming problem. To solve the single-level problem, a decomposition scheme that can resolve the possible dimensionality problem (i.e., a large number of defining variables) is developed. The decomposition scheme is tested, and promising results are shown for well-known test problems.

The continuous network design problem (CNDP) aims to determine the optimal capacity enhancement for a set of selected links in a given network by minimizing both the total system cost and each driver's travel cost (*1*). The CNDP has long been formulated as a bilevel programming problem with the upper level a nonlinear programming (NLP) problem to minimize the system cost and the lower-level user equilibrium (UE) problem to account for driver's route choice behavior. It was first proposed by Morlok (*2*) and subsequently studied by Tan et al. (*3*), Marcotte (*4*), Suwansirikul et al. (*5*), Friesz (*6*), and Yang (*7*), to name but a few. Most of these works on the CNDP focused on heuristic approaches for solving the bilevel model. More-detailed reviews on the CNDP before 2001 may be found elsewhere (*8*).

It has been proved over the years that a bilevel formulation has broad applications not only in the transportation area but also in other engineering and science fields (*9*). Particularly in the mathematical programming literature, the bilevel programming problem is also termed a mathematical program with equilibrium constraints (MPEC), which has been extensively studied (*10*). However, solving such a problem is normally difficult because of the nonconvex and nonsmooth characteristics of the MPEC. Therefore, how to reformulate a general bilevel problem rigorously and solve it efficiently still remains an active research topic in both the transportation field and the mathematical programming community.

By exploring the special structure of the CNDP, Meng et al. (*11*) converted the bilevel problem to a single-level, yet smooth one by introducing a particular gap function for the lower-level UE problem. Though still a nonconvex model, the resulting single-level problem can be solved with existing NLP solution algorithms. Nevertheless, the model of Meng et al. was based on the symmetry assumption of the lower-level problem; that is, there is no interaction among flows on different links. A general UE problem cannot be formulated as an NLP; instead, a nonlinear complementarity problem (NCP) or variational inequality (VI) formulation needs to be adopted. Marcotte and Zhu (*12*) investigated such a general bilevel model, that is, an NLP for the upper level and a VI for the lower level. By defining certain gap functions, they transferred the bilevel problem to a single-level one and solved it with the penalty method. More recently, Patriksson and Rockafellar (*13*) presented a new reformulation technique to convert an MPEC into a constrained and locally Lipschitz minimization problem, which can be further solved with a descent algorithm proposed in the same paper. However, neither Marcotte and Zhu (*12*) nor Patriksson and Rockafellar (*13*) further tested their models by using well-known CNDP examples in the transportation field.

By formulating the asymmetric UE (AUE) as a link-node-based NCP, Ban et al. (*14*) modeled the CNDP with AUE as a mathematical program with complementarity constraints (MPCC). As a special case of MPEC, the MPCC has more plausible properties, which make it easier to solve. In particular, a variety of methods can be applied to convert an MPCC to a single-level NLP problem and then solve it by using existing solution techniques. Therefore, the MPCC has been extensively studied recently (*15–21*). In particular, Ferris et al. (*22, 23*) implemented as a solver a nonlinear program with equilibrium constraints (NLPEC) as a subsystem of general algebraic modeling systems (*24*). The MPCC model proposed by Ban et al. (*14*) was solved by an NLPEC directly. That is, the MPCC model was first converted to an equivalent single-level NLP problem by the application of the complementarity slackness condition. To solve the single-level NLP problem, the strict complementarity condition is relaxed by a relaxation parameter. Then this parameter is progressively reduced, with the resulting relaxed NLP problem solved with existing NLP solvers. Ralph and Wright (*25*) proposed certain conditions under which the relaxation scheme can guarantee to solve the original MPCC successfully. Ban et al. (*14*) demonstrated by using well-known CNDP test problems that such a direct solution approach can generate promising results compared with existing solution techniques for the CNDP.

X. Ban, California Center for Innovative Transportation, Institute of Transportation Studies, University of California, 2105 Bancroft Way, Suite 300, Berkeley, CA 94720-3830. H. X. Liu, Department of Civil Engineering, University of Minnesota–Twin Cities, 122 Civil Engineering Building, 500 Pillsbury Drive SE, Minneapolis, MN 55455. J. Lu, Department of Civil and Environmental Engineering, University of Wisconsin at Madison, 1415 Engineering Drive, Madison, WI 53706. M. C. Ferris, Computer Sciences Department, University of Wisconsin at Madison, 1210 West Dayton Street, Madison, WI 53706.

Transportation Research Record: Journal of the Transportation Research Board, No. 1964, Transportation Research Board of the National Academies, Washington, D.C., 2006, pp. 185–192.

Nevertheless, because the lower-level AUE problem has to be defined on the so-called disaggregated variables (i.e., link flows for different destinations), the direct conversion (and solution) may bring about the dimensionality problem (*11*) for large-scale CNDPs. That is, the resulting single-level NLP might have a large number of defining variables, especially for multiple-origin and multiple-destination (many-to-many) problems. In this study, it is observed that the lower-level AUE problem has a special structure such that it can be easily separated according to individual destinations. Furthermore, the reformulation method applied here can still maintain this special feature. Therefore, by exploring such a special structure of the MPCC model for the CNDP, the authors propose a decomposition scheme to resolve the dimensionality problem. Numerical examples in this paper show that the presented scheme can efficiently solve the CNDP without losing too much of the quality of the solutions.

MPCC MODEL FOR CNDP

Link–Node NCP Formulation for AUE Problem

As shown by Ban et al. (*14*), the AUE problem can be mathematically formulated as follows:

$$\begin{cases} 0 \le \left[\pi_j^s + t_{ij}\left(\sum_{s\in S} v_{ij}^s\right) - \pi_i^s\right] \perp v_{ij}^s \ge 0 & \forall (i,j)\in A, s\in S \\ 0 \le \left[\sum_{(i,j)\in A} v_{ij}^s - \sum_{(j,i)\in A} v_{ji}^s - d_i^s\right] \perp \pi_i^s \ge 0 & \forall i\in N, i\ne s, s\in S \end{cases} \quad (1)$$

Here a given transportation network is denoted $G(N, A)$, where N is the set of nodes and A is the set of links. The index i, j is used to denote nodes in N and (i,j) or ij to denote a link in A. R is denoted as the origin node set, which is a subset of N and generates origin–destination (O-D) trips. Similarly, set S is defined as the destination set, which is also a subset of N and absorbs O-D trips. Further, π_i^s denotes the minimum travel cost from node i to destination s, d_i^s the travel demand from i to s, v_{ij}^s the (disaggregated) flow for link (i, j) for destination s, and t_{ij} the link travel cost for link (i, j). The symbol $\perp$ is the perpendicular operator such that $x \perp y \Leftrightarrow x^T y = 0$. Equation 1 can be rewritten in a matrix form:

$$\begin{cases} 0 \le \left[-\Lambda_s^T \pi^s + t\left(\sum_{s\in S} v^s\right)\right] \perp v^s \ge 0 \\ 0 \le \left[\Lambda_s v^s - d^s\right] \perp \pi^s \ge 0 \end{cases} \quad \forall s\in S \quad (2)$$

where vectors $\pi^s = (\pi_i^s)_{i\in N, i\ne s}$, $v^s = (v_{ij}^s)_{(i,j)\in A}$, and $d^s = (d_i^s)_{i\in N, i\ne s}$ are defined for any given destination node $s \in S$ and $t = (t_{ij})_{(i,j)\in A}$. Also, the standard node–link incidence matrix is represented as Λ, and Λ_s denotes Λ with the row corresponding to destination s removed, which guarantees that Λ_s has full row rank. Equation 2 is the link–node NCP formulation for the AUE problem, which will be utilized later for modeling the bilevel CNDP.

It can easily be observed that Equation 2 has a special structure such that it can be naturally decomposed according to individual destinations. The only place in which interactions exist for variables related to different destinations is the link travel cost vector t since t is defined on the aggregated link flows. This special structure has important effects on how to design a solution algorithm for both the UE problem itself (*26*) and the CNDP problem constructed on the basis of Equation 2, as will be discussed in more detail later.

MPCC Formulation for CNDP

First, additional notation is as follows:

- v_{ij} = total (aggregated) link flow on link (i, j), $v_{ij} = \sum_{s\in S} v_{ij}^s$;
- v = vector of v_{ij}, $v \in R^{|A|}$;
- y_{ij} = capacity enhancement for link $(i, j) \in A$;
- y = vector of y_{ij}, $y \in R^{|A|}$;
- $t_{ij}(v, y_{ij})$ = travel cost on link $(i, j) \in A$, defined as a function of aggregated link flow v and capacity enhancement of (i, j) (i.e., y_{ij});
- $g_{ij}(y_{ij})$ = cost function of capacity enhancement for link $(i, j) \in A$;
- g = vector of $g_{ij}(y_{ij})$, $g \in R^{|A|}$;
- θ = relative weight of total capacity enhancement cost and total travel cost in system design objective function;
- l_{ij}, u_{ij} = lower bound and upper bound for capacity enhancement for link $(i, j) \in A$; and
- l, u = vector of l_{ij} and u_{ij}, respectively, $l, u \in R^{|A|}$.

With this notation in place, as shown by Ban et al. (*14*), the CNDP can be formulated with AUE as the following MPCC model:

$$\min_{y, v^1 \ldots v^s, v^{|S|}, \pi^1 \ldots \pi^s, \pi^{|S|}} \sum_{(i,j)\in A}\left[t_{ij}\left(\sum_{s\in S} v^s, y_{ij}\right)v_{ij}\right] + \theta \sum_{(i,j)\in A} g_{ij}(y_{ij}) \quad (3a)$$

subject to

$$l_{ij} \le y_{ij} \le u_{ij} \quad \forall (i,j)\in A \quad (3b)$$

where (v^s, π^s), $\forall s \in S$, is the solution to the following NCP problem (AUE):

$$0 \le \left[\pi_j^s + t_{ij}\left(\sum_{s\in S} v^s, y_{ij}\right) - \pi_i^s\right] \perp v_{ij}^s \ge 0 \quad \forall (i,j)\in A, s\in S$$

$$0 \le \left[\sum_{(i,j)\in A} v_{ij}^s - \sum_{(j,i)\in A} v_{ji}^s - d_i^s\right] \perp \pi_i^s \ge 0 \quad \forall i\in N, i\ne s, s\in S \quad (3c)$$

Obviously, the MPCC-based CNDP model (Expressions 3*a* through 3*c*) is defined on the upper-level decision variable y_{ij}, $\forall (i, j) \in A$, and the disaggregated link flow (v^s, π^s), $\forall s \in S$. Expression 3*a* is the upper-level objective of the MPCC model, which tries to minimize a weighted summation of the total system travel cost and the enhancement cost; Constraint 3*b* is the bound constraint for the upper-level decision variable y_{ij}, $\forall (i, j) \in A$, and Expression 3*c* is the lower-level AUE formulation that (v^s, π^s), $\forall s \in S$, must satisfy. With matrix notation, Expressions 3*a* through 3*c* can be rewritten:

$$\min_{y, v^1, \ldots v^s, v^{|S|}, \pi^1, \ldots \pi^s, \pi^{|S|}} \left[t\left(\sum_{s\in S} v^s, y\right)\right]^T v + \theta e^T g(y) \quad (4a)$$

subject to

$$l \ge y \le u \quad (4b)$$

where e is the vector of all 1's and $\{(v^s, \pi^s), \forall s \in S\}$ is the solution to the following NCP model:

$$\begin{cases} 0 \le \left[-\Lambda_s^T \pi^s + t\left(\sum_{s\in S} v^s, y\right)\right] \perp v^s \ge 0 \\ 0 \le \left[\Lambda_s v^s - d^s\right] \perp \pi^s \ge 0 \end{cases} \quad \forall s \in S \tag{4c}$$

The MPCC model (Expressions 4a through 4c) can be tackled by being converted to a single-level equivalence and then solved by using a decomposition scheme that will be discussed in the next section.

SOLUTION ALGORITHM

Single-Level NLP Formulation for MPCC Model

Because the NCP formulation (Equation 4c) can be readily replaced by its equivalent complementarity slackness condition and additional nonnegativity constraints, the MPCC model of the CNDP (Expressions 4a through 4c) can be straightforwardly converted into a single-level NLP model as follows:

$$\min_{y, v^1, \ldots, v^s, v^{|S|}, \pi^1, \ldots, \pi^s, \pi^{|S|}} \left[t\left(\sum_{s\in S} v^s, y\right)\right]^T v + \theta e^T g(y) \tag{5a}$$

subject to

$$l \le y \le u \tag{5b}$$

$$-\Lambda_s^T \pi^s + t(v, y) \ge 0 \qquad \forall s \in S \tag{5c}$$

$$\Lambda_s v^s - d^s \ge 0 \qquad \forall s \in S \tag{5d}$$

$$v^s \ge 0 \qquad \forall s \in S \tag{5e}$$

$$\pi^s \ge 0 \qquad \forall s \in S \tag{5f}$$

$$\left(\Lambda_s v^s - d^s\right)_i \pi_i^s = 0 \qquad \forall s \in S, i \in N, i \ne s \tag{5g}$$

$$\left[-\Lambda_s^T \pi^s + t(v, y)\right]_{ij} v_{ij}^s = 0 \qquad \forall s \in S, (i,j) \in A \tag{5h}$$

where the lower-level NCP formulation in Expression 4c is replaced by its equivalent complementarity slackness condition in Constraints 5c through 5h. Evidently, under the assumption that both the link travel cost function t and the function g are smooth, the single-level NLP model (Expressions 5a through 5h) involves only smooth functions with respect to $(y, v, v^i, \ldots, v^s, v^{|S|}, \pi^1, \ldots, \pi^s, \pi^{|S|})$. Hence, it is a smooth and nonlinear optimization problem. However, this NLP formulation lacks sound mathematical properties because of the complementarity slackness constraints (5g and 5h). Actually, because of these two constraints, the single-level model is nonconvex and, most important, the Mangasarian-Fromovitz constraint qualification does not hold (*10*). Therefore, solving the single-level NLP model directly is usually difficult, and a progressive relaxation algorithm will normally be adopted instead.

Ban et al. (*14*) solved the single-level NLP problem (Expressions 5a through 5h) directly by applying a relaxation scheme, particularly by the NLPEC solver developed by Ferris et al. (*22*). Promising results were reported in their study for well-known test CNDPs in the literature. However, the NLP equivalence (Expressions 5a through 5h) involves the disaggregated variables explicitly and hence has a large dimension for large-scale problems. Such a dimensionality problem may likely prohibit the application of the direct solution. Nevertheless, it is clear that Constraints 5c through 5h are defined according to individual destinations, except for the interaction of disaggregated link flows on the link travel cost function t. This feature makes it possible to employ certain decomposition techniques to solve the single-level NLP model more efficiently.

Decomposition Scheme for Solving Single-Level NLP Model

For the single-level NLP model (Expressions 5a through 5h), Constraint 5b is defined on the upper-level decision variable y only and those from Constraints 5c through 5h are defined according to each individual destination, except for the interaction of the disaggregated link flow variables v^s, $\forall s \in S$, and y in the link travel cost function t. From this observation, it is intuitive to apply certain decomposition schemes for solving the single-level model. In the literature, decomposition schemes can be grouped into two categories: the Gauss–Seidel (GS) (or sequential) decomposition and the Jacobi (or parallel) decomposition (*27, 28*). Although the Jacobi decomposition method is amenable to parallel computing, the GS approach has been proved to have better convergence performance since it can incorporate the newest available information (*29*). Here, the discussion will concentrate on the GS method.

With the application of GS decomposition, the interaction of v^s, $\forall s \in S$, and y in t can be temporarily fixed. Then the (possible) large-size, single-level NLP model can be converted into multiple, yet smaller-dimensional optimization problems. In the current case, these smaller-dimensional problems will be defined on y and individual (v^s, π^s), $\forall s \in S$, respectively. That is, the single-level NLP model can be decomposed into the following $|S| + 1$ smaller-dimensional NLP problems:

$$\min_y \left[t\left(\sum_{s\in S} \bar{v}^s, y\right)\right]^T \left(\sum_{s\in S} \bar{v}^s\right) + \theta e^T g(y) \tag{6}$$

subject to $l \le y \le u$,

and

$$\min_{v^s, \pi^s} \left[t\left(\sum_{s'\in S, s'\ne s} \bar{v}^{s'} + v^s, \bar{y}\right)\right]^T v^s$$

subject to

$$\begin{aligned} & -\Lambda_s^T \pi^s + t\left(\sum_{s'\in S, s'\ne s} \bar{v}^{s'} + v^s, \bar{y}\right) \ge 0 \\ & \Lambda_s v^s - d^s \ge 0 \quad \forall s \in S \\ & \left[-\Lambda_s^T \pi^s + t\left(\sum_{s'\in S, s'\ne s} \bar{v}^{s'} + v^s, \bar{y}\right)\right]^T v^s = 0 \\ & \left(\Lambda_s v^s - d^s\right)^T \pi^s = 0 \\ & v^s \ge 0, \pi^s \ge 0 \end{aligned} \tag{7}$$

Essentially, Expression 6 is defined on the upper-level decision variable only with v^s, $\forall s \in S$, fixed as $\bar{v}^s$, whereas for each destination

s, after $v^{s'}$, $\forall s' \in S$, $s' \neq s$ is temporarily fixed as $\bar{v}^{s'}$ and y as $\bar{y}$, an NLP model can be obtained, as shown in Expression 8. It can immediately be observed that the constraints in Expression 7 actually define an NCP problem for each individual destination; that is,

$$\begin{cases} 0 \le \left[-\Lambda_s^T \pi^s + t\left(\sum_{s' \in S, s' \neq s} \bar{v}^{s'} + v^s, \bar{y} \right) \right] \perp v^s \ge 0 \\ 0 \le \left[\Lambda_s v^s - d^s \right] \perp \pi^s \ge 0 \end{cases} \quad \forall s \in S \qquad (8)$$

Under certain monotonicity conditions (*26*), Expression 8 has a unique solution in terms of v^s. This fact means that solving the minimization problem (Expression 7) is equivalent to solving the NCP problem (Expression 8) for each destination $s \in S$. Because of the smaller dimension of both Expressions 6 and 8 compared with the original single-level NLP model, they can be solved much more efficiently. Solving these smaller-dimension problems tackles only the decomposed version of the original single-level NLP model. The overall solution method is thus an iterative one with the decomposed problems solved at each iteration. Under the assumption that the obtained solution from the decomposed problems defines a descent direction to the original single-level problem (although verifying this is not trivial), the optimal step size for computing the next iterate can be obtained by a line search as follows:

$$\min_{\alpha} \left(t\left\{ \sum_{s \in S} \left[\bar{v}^s + \alpha(\hat{v}^s - \bar{v}^s) \right], \bar{y} + \alpha(\hat{y} - \bar{y}) \right\} \right)^T \cdot \left\{ \sum_{s \in S} \left[\bar{v}^s + \alpha(\hat{v}^s - \bar{v}^s) \right] \right\} + \theta e^T - g\left[\bar{y} + \alpha(\hat{y} - \bar{y}) \right] \qquad (9)$$

where $\bar{v}^s$, $\bar{y}$ denotes current fixed variables, $\hat{v}^s$; $\hat{y}$, the solution obtained from Expressions 6 and 8; and α, the step size.

To summarize, the iterative algorithm for solving Expressions 5*a* through 5*h* can be given as follows:

Step 1. Initialization. Assign initial values for the defining variables, v^{s0}, π^{s0}, y^{s0}, and set iteration count $n = 0$.

Step 2. GS decomposition scheme.

Step 2.1. Solve the decomposed problem (Expression 6) by setting $\bar{v}^s = v^{sn}$, $\forall s \in S$. Denote the obtained solution as $\hat{y}^n$.

Step 2.2. For each destination $s \in S$, solve the decomposed NCP problem (Expression 8) by setting $\bar{y} = y^{sn}$ and $\bar{v}^{s'} = v^{s'n}$, $\forall s' \in S$, $s' \neq s$. Denote the obtained solution as $\hat{v}^{sn}$, $\forall s \in S$.

Step 3. Line search. Solve the one-dimensional NLP model (Expression 9) to obtain the optimal step size, denoted α*.

Step 4. Convergence test. If a certain convergence criterion is met, stop. Otherwise, set $v^{s(n+1)} = v^{sn} + \alpha^* (\hat{v}^{sn} - v^{sn})$, $y^{n(n+1)} = y^{sn} + \alpha^* (\hat{y}^{sn} - y^{sn})$, $n = n + 1$, and go to Step 2.1.

In Step 4 of the algorithm, the stopping criterion has to consider both the upper-level objective value and the lower-level UE condition. For the upper level, one can check if the objective values remain stable over the past several iterations, whereas for the lower-level UE condition, a criterion similar to the relative gap of Boyce et al. (*30*) is applied:

$$\text{relgap} = \frac{\sum_{(i,j) \in A} t_{ij}(v_{ij}, y_{ij}) \cdot (v_{ij} - v_{ij}^{\text{AON}})}{\sum_{(i,j) \in A} t_{ij}(v_{ij}, y_{ij}) \cdot v_{ij}} \qquad (10)$$

In Equation 10, v_{ij}^{AON} denotes the all-or-nothing link flow provided the link travel time is fixed at link flow v_{ij} and capacity enhancement y_{ij}. Clearly, relgap ≥ 0 and if relGap $= 0$, the lower-level UE condition will hold exactly.

NUMERICAL EXAMPLES

In this section, the proposed decomposition scheme is tested on a well-known and relatively large-scale CNDP with both symmetric and asymmetric user equilibria. In particular, the solutions will be compared with those obtained with existing CNDP algorithms. Before the actual network is introduced, however, the link travel time functions for both symmetric and asymmetric cases are provided.

Link Travel Time Function

For the symmetric UE, the link travel time function is separable; this implies that the travel time on a particular link is dependent only on its own traffic flow. In the transportation field, the most popularly used function form for the symmetric case is the Bureau of Public Roads (BPR) function:

$$t_{ij}(v_{ij}, y_{ij}) = A_{ij} + B_{ij} \left[v_{ij} / (K_{ij} + y_{ij}) \right]^4 \qquad (11)$$

where A_{ij}, B_{ij}, and K_{ij} are constants for any link (i,j).

However, for the asymmetric case, no appropriate function form has been suggested in the literature. In this study, the following link travel time function is adopted:

$$t_{ij}(v_{ij}, y_{ij}) = A_{ij} + B_{ij} \left[\frac{\left(\sum_{(k,j) \in A} \rho_{ij,kj} v_{kj} \right)}{(K_{ij} + y_{ij})} \right]^4 \qquad (12)$$

where $0 \le \rho_{ij,kj} \le 1$ denotes the impact factor of the flow on link (k, j) to the travel cost of link (i, j). Apparently, $\rho_{ij,ij} = 1$, $\forall (i, j) \in A$, and further if $\rho_{ij,kj} = 0$, $\forall j \in N$, $(i, j) \in A$, $(k, j) \in A$, $i \neq k$, Equation 12 will reduce to the standard BPR function in Equation 11. It should be pointed out here that Equation 12 is just an intuitive way to achieve (asymmetric) link interactions among adjacent links to demonstrate the proposed model and algorithm for the CNDP with AUE. How to design a practically reasonable asymmetric link cost function for a given network is beyond the scope of this paper.

Test Networks

The test network in this study is the Sioux Falls, North Dakota, network, which was first constructed and studied by LeBlanc (*31*). It contains 24 nodes and 76 links, as shown in Figure 1. All 24 nodes can be either origin or destination node, or both. The data of the network can be found elsewhere (*7*), including the parameters A_{ij}, B_{ij}, and K_{ij} in the link travel time function (Equation 11 or 12). In particular, only 10 links are selected for capacity enhancement, namely, Links 16, 17, 19, 20, 25, 26, 29, 39, 48, and 74 in Figure 1. The upper bound of the enhancement for each link is set to 25, that is, $0 \le y_{ij} \le 25$, $\forall (i, j) \in A$. Furthermore, the cost function for capacity enhancement for each link is $g_{ij}(y_{ij}) = 0.001 y_{ij}^2$, $\forall (i, j) \in A$. In other words, the upper-level objection function of CNDP is

$$z(v, y) = \sum_{(i,j) \in A} \left[t_{ij} \left(\sum_{s \in S} v^s, y_{ij} \right) v_{ij} + 0.001 \theta_{ij} y_{ij}^2 \right] \qquad (13)$$

Finally, the impact factors in Equation 12 for the asymmetric case are given in Table 1. Table 1 shows only the impact factors for $\rho_{ij,kj}$,

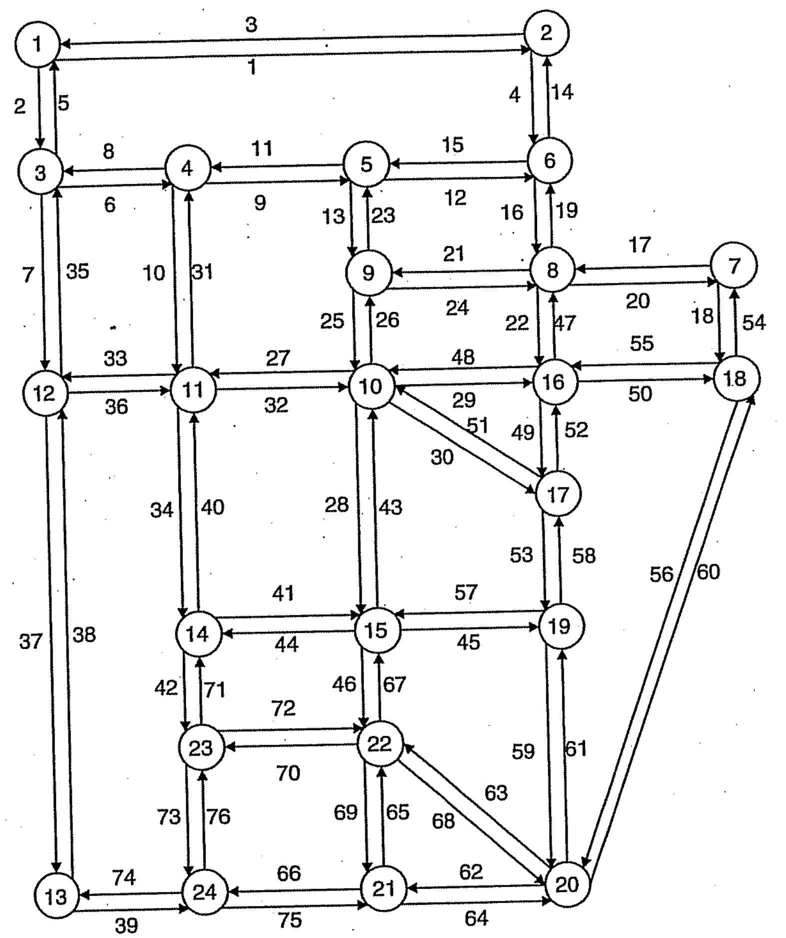

FIGURE 1 Test Network (Sioux Falls).

TABLE 1 Impact Factors for Test Network

Link	6.8	7.8	8.6	8.7	10.9	10.16	13.24	16.10
4 (2.6)		.1	.05					
12 (5.6)			.05					
13 (5.9)					.06			
16 (6.8)		0.06						
17 (7.8)	0.08							
22 (8.16)					0.1			
24 (9.8)						0.1		
25 (9.10)	0.06	0.08						
25 (10.9)								0.08
54 (18.7)						0.05		
55(18.16)				0.06				
66 (21.24)						0.07		
73 (23.24)							0.05	

TABLE 2 Comparisons of Results for Symmetric Case

Variable	Murtagh and Saunders 1981 (*32*)	Abdulaal and LeBlance (1979) (*33*)	Suwansirikul et al. 1987 (*5*)	Friesz et al. 1992 (*6*)	Meng et al. 2001 (*11*)	Decomposition Scheme Proposed in This Paper
Initial value of y_{ij}	2	1	12.5	6.25	12.5	0
$y_{6,8}$ (Link 16)	4.8	3.8	4.59	5.38	5.5728	5.269770
$y_{7,8}$ (Link 17)	1.2	3.6	1.52	2.26	1.6443	1.378772
$y_{8,6}$ (Link 19)	4.8	3.8	5.45	5.5	5.6228	5.269853
$y_{8,7}$ (Link 20)	0.8	2.4	2.33	2.01	1.6443	1.378635
$y_{9,10}$ (Link 25)	2	2.8	1.27	2.64	3.1437	2.766501
$y_{10,9}$ (Link 26)	2.6	1.4	2.33	2.47	3.2837	2.766446
$y_{10,16}$ (Link 29)	4.8	3.2	0.41	4.54	7.6519	4.669070
$y_{13,24}$ (Link 39)	4.4	4	4.59	4.45	3.8035	4.350875
$y_{16,10}$ (Link 48)	4.8	4	2.71	4.21	7.382	4.668969
$y_{24,13}$ (Link 74)	4.4	4	2.71	4.67	3.6935	4.350856
Value of objective function	81.25	81.77	83.47	80.87	81.752	81.102
Number of solved UE	58	108	12	3900	2700	36

$\forall j \in N$, $(i, j) \in A$, $(k, j) \in A$, $i \neq k$ and those for $\rho_{ij,ij} = 1$, $\forall(i, j) \in A$, are not given explicitly.

Results Analysis

To ensure the convergence, the stopping criterion was set as relGap $\leq 1.0e - 6$ and the fluctuation of the objective values for the last five iterations was less than $1.0e - 5$. First, the solution for the symmetric case is shown in Table 2, which also gives the solutions obtained by existing solution techniques. In particular, the simulated annealing (SA) method (*6*) obtained the best solution found so far. From Table 2, it is obvious that the proposed decomposition scheme can generate a solution whose objective value is close to that obtained with SA. Furthermore, only 36 UE solves were performed in the current scheme, which is significantly less than with most other algorithms. This finding demonstrates that the proposed method can be much more time-efficient than other approaches. With the application of the direct solution method of Ban et al. (*14*), the symmetric problem can also be solved with an even lower objective value (80.5157); however, the direct solution method tends to be more time-consuming and cannot be applied to large-scale CNDPs because of the so-called dimensionality problem. So the direct solution method is not given or compared here.

Figure 2 further illustrates the convergence of the objective value. For each iteration of the algorithm, one run of the UE is mainly solved. It can be seen from Figure 2 that the objective value converges quickly. However, because of the nonconvexity of the problem, the objective value may not decrease monotonically. Normally, the objective value decreases dramatically at the first several iterations. It then starts to fluctuate in a relatively small range. Finally, it becomes stable rapidly. These findings are more evident in Figure 3 for the asymmetric case. Meanwhile, the convergence of the lower-level UE is shown in Figure 4. Apparently, the UE condition also converges quickly. If Figures 2 and 4 are combined, it can be concluded that if only an approximate solution is required, the proposed decomposition scheme can solve the CNDP efficiently by only a few iterations. Figure 4 also shows that the relative gap is always nonnegative, as discussed previously.

For the asymmetric case, since there is no solution reported in the literature (except the direct solution method, which produces an objective value of 83.9095), Table 3 only shows the one obtained with the proposed decomposition scheme. Table 3 also demonstrates that solving the asymmetric case requires more iterations and

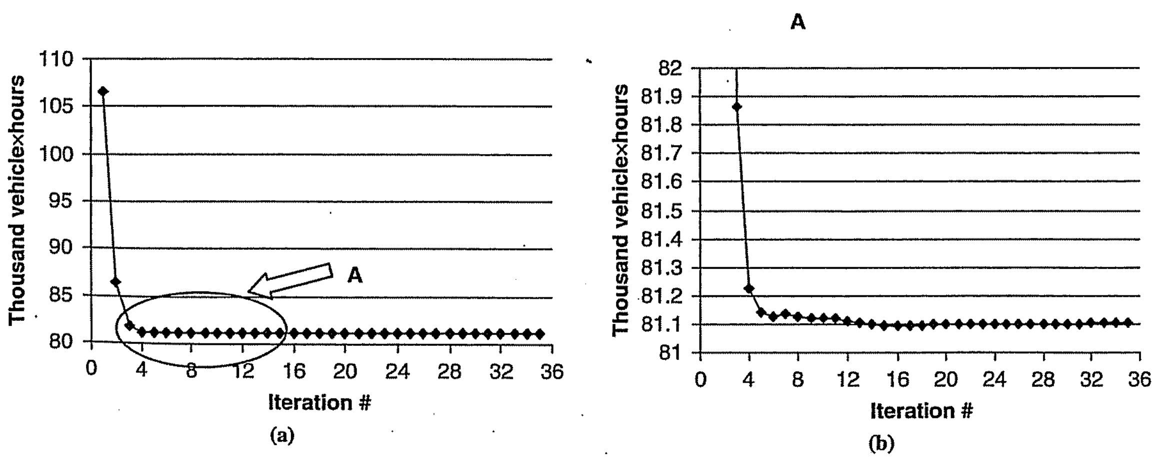

FIGURE 2 Convergence of objective value for symmetric case: (*b*) shows enlarged view of Area A in (*a*).

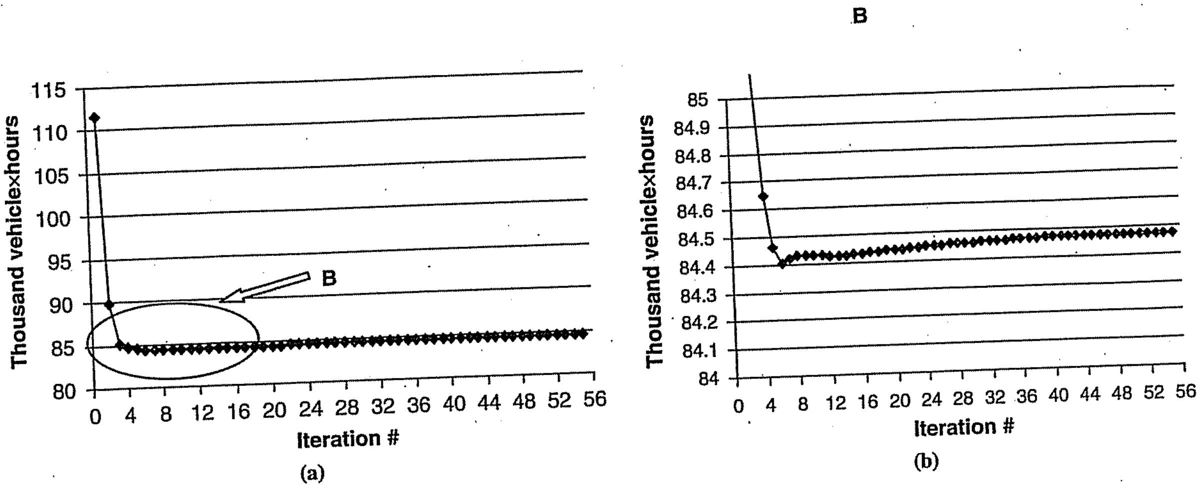

FIGURE 3 Convergence of objective value for asymmetric case (*b*) shows enlarged view of Area B in (*a*).

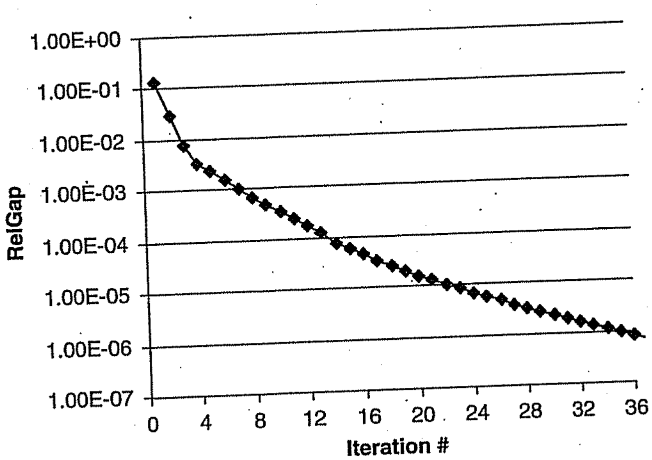

FIGURE 4 Convergence of lower-level UE for symmetric case.

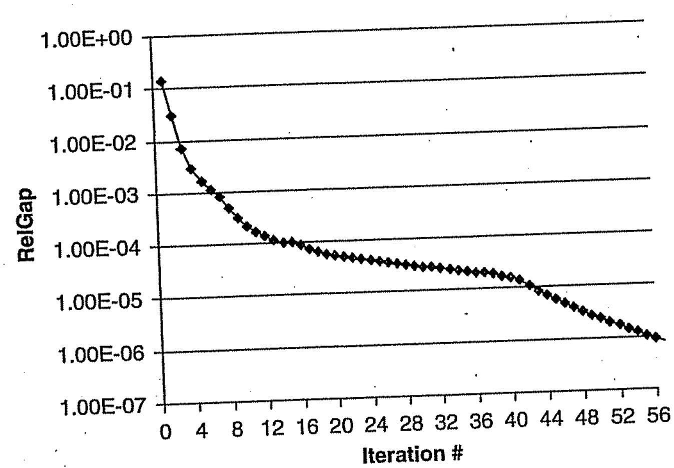

FIGURE 5 Convergence of lower-level UE for asymmetric case.

that the objective value is larger than that of the symmetric case. Figures 3 and 5 further illustrate the convergence of the objective value and the lower-level UE, respectively. It can easily be observed from Figure 3 that the objective value becomes stable after only the first several iterations. This case is identical to the symmetric case in Figure 2, implying that an approximate solution can be also efficiently obtained for the asymmetric case.

CONCLUSIONS

A decomposition scheme is proposed for solving the MPCC model for the CNDP with AUE. Instead of directly solving the equivalent single-level NLP problem of the MPCC model, the GS decomposition scheme was applied to this single-level problem by exploring its special structure. Then the possible large-size single-level NLP problem can be converted into multiple yet smaller-dimensional problems, which can be tackled more easily. The numerical examples discussed showed that the presented decomposition scheme can generate promising results, especially the efficient solution of a well-known CNDP test problem with the objective value close to the best-known one in the literature.

For future studies, conditions under which the proposed decomposition scheme can guarantee to generate an optimal solution for the CNDP will be investigated. Further, the decomposition scheme was tested only on the Sioux Falls network in this study. Extensive testing of the proposed model and algorithm for solving larger-scale CNDPs will be conducted in future research.

TABLE 3 Solution for the Asymmetric Case

Initial y	y Link 16	y Link 17	y Link 19	y Link 20	y Link 25	y Link 26	y Link 29	y Link 39	y Link 48	y Link 74	Obj. Value	# UE
12.5	4.96983	1.63806	5.27757	1.43692	3.12381	2.87238	5.25048	4.70207	4.92401	4.12252	**84.475**	56

REFERENCES

1. LeBlanc, L., and D. E. Boyce. A Bilevel Programming Algorithm for Exact Solution of the Network Design Problem with User-Optimal Flows. *Transportation Research,* Vol. 20B, 1986, pp. 259–265.
2. Morlok, E. K. *Development and Application of a Highway Network Design Model.* FHWA, U.S. Department of Transportation, 1973.
3. Tan, H.-N., S. B. Gershwin, and M. Athana. *Hybrid Optimization in Urban Traffic Networks.* Report DOT-TSC-RSPA-79-7. Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, 1979.
4. Marcotte, P. Network Optimization with Continuous Control Parameters. *Transportation Science,* Vol. 17, 1983, pp. 181–197.
5. Suwansirikul, C., T. L. Friesz, and R. L. Tobin. Equilibrium Decomposed Optimization: A Heuristic for the Continuous Equilibrium Network Design Problem. *Transportation Science,* Vol. 21, 1987, pp. 254–263.
6. Friesz, T. L. Simulated Annealing Approach to the Network Design Problem with Variational Inequality Constraints. *Transportation Science,* Vol. 26, 1992, pp. 18–26.
7. Yang, H. Sensitivity Analysis for the Elastic Demand Network Equilibrium Problem with Applications. *Transportation Research,* Vol. 31B, 1997, pp. 55–70.
8. Yang, H., and M. G. H. Bell. Transport Bilevel Programming Problems: Recent Methodological Advances. *Transportation Research,* Vol. 35B, 2001, pp. 1–4.
9. Outrata, J., M. Kocvara, and J. Zowe. *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints.* Kluwer Academic Publishers, 1998.
10. Luo, Z.-Q., J. S. Pang, and D. Ralph. *Mathematical Programs with Equilibrium Constraints.* Cambridge University Press, Cambridge, United Kingdom, 1996.
11. Meng, Q., H. Yang, and M. G. H. Bell. An Equivalent Continuously Differentiable Model and a Locally Convergent Algorithm for the Continuous Network Design Problem. *Transportation Research,* Vol. 35B, 2001, pp. 83–105.
12. Marcotte, P., and D. L. Zhu. Exact and Inexact Penalty Methods for the Generalized Bilevel Programming Problem. *Mathematical Programming,* Vol. 74, 1996, pp. 141–157.
13. Patriksson, M., and R. T. Rockafellar. A Mathematical Model and Descent Algorithm for Bilevel Traffic Management. *Transportation Science,* Vol. 36, 2002, pp. 271–291.
14. Ban, X. G., X. H. Liu, and M. C. Ferris. A General MPCC Model and Its Solution Algorithm for Continuous Network Design Problem. *Mathematical and Computer Modeling,* Vol. 43, No. 5–6, 2006, pp. 493–505.
15. Scheel, H., and S. Scholte. Mathematical Programs with Complementarity Constraints: Stationarity, Optimality, and Sensitivity. *Mathematics of Operations Research,* Vol. 22, No. 1, 2000, pp. 1–22.
16. Anitescu, M. On Solving Mathematical Programs with Complementarity Constraints as Nonlinear Programs. Preprint ANL/MCS-P864–1200. Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Ill., 2000.
17. Anitescu, M. Global Convergence of an Elastic Mode Approach for a Class of Mathematical Programs with Complementarity Constraints. Preprint ANL/MCS–P1143-0404. Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Ill., 2004.
18. Fletcher, R., and S. Leyffer. Solving Mathematical Programs with Complementarity Constraints as Nonlinear Programs. *Optimization Methods and Software,* Vol. 19, No. 1, 2004, pp. 15–40.
19. Hu, X., and D. Ralph. A Note on Sensitivity of Value Functions of Mathematical Programs with Complementarity Constraints. *Mathematical Programming,* Vol. 93, No. 2, 2002, pp. 265–279.
20. Jiang, H., and D. Ralph. Smooth SQP Methods for Mathematical Programs with Nonlinear Complementarity Constraints. *Society for Industrial Applied Mathematics Journal of Optimization,* Vol. 10, No. 3, 2000, pp. 779–808.
21. Jiang, H., and D. Ralph. Extension of Quasi-Newton Methods to Mathematical Programs with Complementarity Constraints. *Computational Optimization and Applications,* Vol. 25, No. 1–3, 2003, pp. 123–150.
22. Ferris, M. C., S. P. Dirkse, and A. Meeraus. *Mathematical Programs with Equilibrium Constraints: Automatic Reformulation and Solution via Constrained Optimization.* Numerical Analysis Group Research Report NA-02/11. Oxford University Computing Laboratory, Oxford University, Oxford, United Kingdom, 2002.
23. Ferris, M. C., S. P. Dirkse, and A. Meeraus. Mathematical Programs with Equilibrium Constraints: Automatic Reformulation and Solution via Constrained Optimization. In *Frontiers in Applied General Equilibrium Modeling* (T. J. Kehoe, T. N. Srinivasan, and J. Whalley, eds.), Cambridge University Press, Cambridge, United Kingdom, 2004.
24. Brooke, A., D. Kendrick, and A. Meeraus. *GAMS, a User's Guide.* GAMS Development Corporation, 1998.
25. Ralph, D., and S. J. Wright. Some Properties of Regularization and Penalization Schemes for MPECs. *Optimization Methods and Software,* Vol. 19, 2004, pp. 527–556.
26. Ban, X. G. *Quasi-Variational Inequality Formulations and Solution Approaches for Dynamic User Equilibria.* Ph.D. thesis. University of Wisconsin, Madison, 2005.
27. Ortega, J. M., and W. C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables.* Academic Press, New York, 1970.
28. Zadeh, N. A Note on the Cyclic Coordinate Ascent Method. *Management Science,* Vol. 16, 1969–1970, pp. 642–644.
29. Bertsekas, D. P., and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods.* Prentice-Hall, London, 1997.
30. Boyce, D. E., B. Ralevic-Dekic, and H. Bar-Gera. Convergence of Traffic Assignment: How Much Is Enough? *Journal of Transportation Engineering,* ASCE, Vol. 130, No. 1, 2004, pp. 49–55.
31. LeBlanc, L. An Algorithm for the Discrete Network Design Problem. *Transportation Science,* Vol. 9, 1975, pp. 183–199.
32. Murtagh, B. A. and M. A. Saunders. *The Implementation of a Lagrangian-Based Algorithm for Sparse Nonlinear Constraints.* SOL Technical Report. Stanford University, Stanford, California, 1981.
33. Abdulaal, M., and L. J. LeBlanc. Continuous Equilibrium Network Design Models. *Transportation Research,* Vol. 13B, 1979, pp. 19–32.

The Transportation Network Modeling Committee sponsored publication of this paper.