Application and Testing of the Diagonalization Algorithm for the Evaluation of Truck-Related Highway Improvements

HANI S. MAHMASSANI, KYRIACOS C. MOUSKOS, AND C. MICHAEL WALTON

Highway and transportation officials are increasingly concerned about accommodating rising truck traffic and the associated size and weight trends. A network traffic assignment procedure is an essential component of the methodological support for the identification, evaluation, and selection of truck-related physical and operational improvements in a highway system. A general mechanism is presented for the network representation of improvements consisting not only of physical capacity expansion but also corresponding operational strategies in the form of (existing or new) lane-access restrictions to either vehicle class; this mechanism allows the consideration, as a special case, of exclusive truck lanes or facilities contemplated by several agencies. The special requirements of the traffic assignment procedure in this context, including the need to explicitly consider the asymmetric interaction between cars and trucks, give rise to potentially serious methodological difficulties that must be addressed for specific types of applications. The applicability of the diagonalization algorithm to such problems is investigated by using numerical experiments on three test networks under varying conditions. The three test networks include an abstracted condensed representation as well as a full-scale version of the Texas highway network, thus providing a realistic case application. The main aspects of the algorithm's performance addressed in these experiments are its convergence characteristics as well as the effectiveness of some computational streamlining strategies. Although convergence is not guaranteed a priori, it was actually achieved in all test cases. Furthermore, it is shown that shortcut strategies can considerably reduce the algorithm's computational requirements.

During the past 30 years, there has been a considerable increase in the size of the fleet of passenger cars and trucks, with an increasingly diverse mix of vehicles in the traffic stream. Different types of vehicles are entering the highway system, and they have different physical and performance characteristics. Recent trends toward less stringent regulations have allowed larger and heaver trucks in the highway system, jeopardizing geometric and capacity considerations in some parts of the system and resulting in increased pavement deterioration. Furthermore the interaction of vehicles with different sizes and performance characteristics, such as large combination trucks, on the one hand, and subcompact passenger cars, on the other, may have resulted in more hazardous driving conditions, with increased potential severity of collisions.

Department of Civil Engineering, University of Texas at Austin, Austin, Tex. 78712.

The foregoing concerns have led the appropriate agencies to consider improvements in the highway infrastructure, such as the construction of exclusive facilities for different classes of users, as well as operational measures involving the restriction of access to existing selected lanes for certain vehicle types. The current work was motivated by the need for a network-modeling methodology to support the design and evaluation of such truck-related improvements in a highway system *(1)*. A central component in this methodology is the assignment of traffic flows to various parts of the network in response to contemplated improvements. These flows are essential in the calculation of costs and benefits incurred with a particular set of improvements. Specifically, the function of the network traffic assignment procedure is to determine the link flow patterns resulting from the allocation of origin-destination trip matrices (for cars and trucks, respectively) corresponding to present or future conditions to a given highway network. By changing the configuration of the latter or by modifying the characteristics of some of its links, the traffic assignment procedure allows the assessment of the impact in all parts of the network, and on various user groups, of selected truck-related improvements, such as special truck lanes or facilities in designated corridors or highway sections.

To be useful in the evaluation of truck-related improvements, which arise in urban as well as intercity contexts, the network traffic assignment model should (a) yield separate estimates of truck flows and passenger-car flows on every network link, (b) capture the nonlinear dependence of the travel time (or cost) incurred by link users on the total flow using that link, (c) recognize the interaction between vehicle classes (cars and trucks) sharing the same right-of-way, and (d) be policy sensitive in that it should allow the representation of the contemplated truck-related countermeasures, particularly because these are not limited to the construction of additional facilities but also include operational strategies affecting both existing and proposed facilities.

Items b and c are generally captured in the link performance functions, which yield the travel time as a function of the flow of each vehicle class on that link, or more generally as a function of the entire link flow pattern. The resulting network user equilibrium problem requires the simultaneous solution of the link flows (which determine link travel times) and the link travel times (which in turn affect the routes chosen by motorists in the network). Furthermore, when interactions among vehicle

classes are explicitly represented, they are effectively equivalent to interactions among links (where different conceptual links are defined for each user class, as discussed in the next section). Such interactions are asymmetric, meaning that the marginal contribution of a vehicle belonging to a given category to the travel time of other classes is different from the marginal contribution of a vehicle in the latter category to the former's travel time. For this type of problem, interactions between cars and trucks are generally asymmetric, thereby giving rise to a user equilibrium network assignment problem with asymmetric link interactions.

A more detailed presentation of the mathematical and algorithmic background of this problem as it arises in this context and of its solution approach can be found in the report by Mouskos et al. *(2)*. Essentially, there are no guaranteed procedures to solve the network equilibrium problem with asymmetric interactions. However, an approach known as the diagonalization algorithm has emerged *(3–5)* as a promising one to solve for such general equilibrium problems. Other methods, particularly linearization methods [of which the projection method proposed by Dafermos *(6)* is a special case] and simplicial decomposition *(7)*, have also been suggested. By and large, the diagonalization approach is the most easily accessible to practitioners and researchers, because it can be implemented with relatively simple modification of widely available packages for the Frank-Wolfe (F-W) solution algorithm for the standard single-class network equilibrium problem.

The necessary conditions for the diagonalization algorithm's convergence to the desired equilibrium solution are not well understood in the transportation science literature. However, known sufficient conditions *(3)* are recognized as being too strict and often far from necessary *(5)*. Therefore, it is necessary to test the approach in the specific context in which it is to be employed. Furthermore, in its complete version, the algorithm is rather demanding computationally. Fortunately, some shortcuts have been suggested to improve its performance in this regard *(3)*. However, these approaches remain to be tested, because numerical experience to date appears to have been limited to small unrealistic networks. A major objective of this effort is therefore to test these shortcut strategies and develop computational experience in realistic networks in order to assess their usefulness as operational tools in the analysis of truck-related improvements in a highway network.

The principal objective of this work is thus to assess the practical applicability of the diagonalization algorithm for the evaluation of truck-related improvements. This involves two principal tasks: (a) the development of a mechanism for representing the improvements of interest and (b) testing the performance of the diagonalization algorithm for this type of application. Two questions are of concern in this regard: (a) convergence of the algorithm, which, as noted earlier, is not guaranteed, and (b) streamlining or shortcut strategies in the implementation of the algorithm in order to reduce its otherwise heavy computational cost.

This paper is organized as follows. In the next section, the improvement options are defined and their network representation is described. This is followed by a brief presentation of the steps of the algorithm and the shortcut strategies. Next, the numerical experiments conducted to examine the convergence of the algorithm and to compare the effectiveness of the various shortcut strategies are presented. The results of these experiments are then summarized, followed by concluding comments.

NETWORK REPRESENTATION OF IMPROVEMENT OPTIONS

As noted previously, each highway link is used by two classes of vehicles, cars and heavy trucks, which interact, through their use of the common shared right-of-way, in determining the travel time incurred by vehicles of both classes using that link. Naturally, it is possible to further disaggregate the heavy vehicles into various vehicle categories, but the dichotomy between cars and trucks is sufficient for most purposes. The interaction between the two classes on a highway link is represented through the use of identical networks (referred to as "copies" of each other) for each class. Each physical highway link is thus decomposed into two "conceptual" links. Each link has its own performance (or travel cost) function, and the flow on any given link consists of one or the other designated class only. Interaction among the various classes using a particular physical link thus translates into interaction among links in this network representation.

In order to test the effect of truck-related improvements to a particular highway link, it is necessary to devise a general-purpose mechanism that allows the representation of this improvement not only in terms of lane addition, but also in terms of how this new lane might be operated in conjunction with the existing lanes (e.g., access restriction of a given lane to certain vehicle classes). In particular, the new lane can be used by trucks only (exclusive truck lane), passenger cars only (restricted lane), or all traffic. Similarly, the existing lanes can be used by either all traffic or car traffic only. The following four types of improvement options, defined in terms of different mutually exclusive combinations of these factors, are of particular interest in this study:

Option 1: Expand the link by one lane and allow all traffic on entire link.

Option 2: Expand the link by one lane but allow only truck traffic on new lane with all traffic allowed on old lanes.

Option 3: Expand the link by one lane, but all truck traffic must use new lane with all car traffic allowed on old lanes only.

Option 4: Expand the link by one lane, but allow only car traffic on new lane with all traffic allowed on old lanes. Note that this option is equivalent to building a new lane that would be open to both cars and trucks and at the same time restricting trucks from using the left-most lane.

Figure 1 shows the implementation of each of the four options on a link that currently has three lanes.

The general mechanism for representing the foregoing improvements is as follows. With each given physical highway section, with start node i and end node j, it has been shown that two conceptual link copies are defined, one for trucks and one for cars, with the "coupling" accomplished through a special numbering scheme as well as through the respective link performance functions, as described by Mouskos et al. *(2)*. One additional node (dummy node) and two additional links are

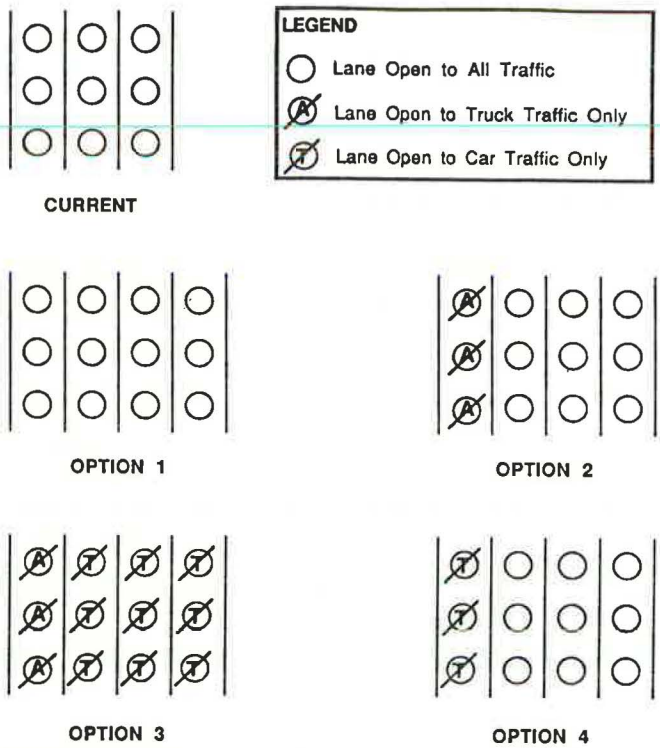

FIGURE 1 Link improvement options.

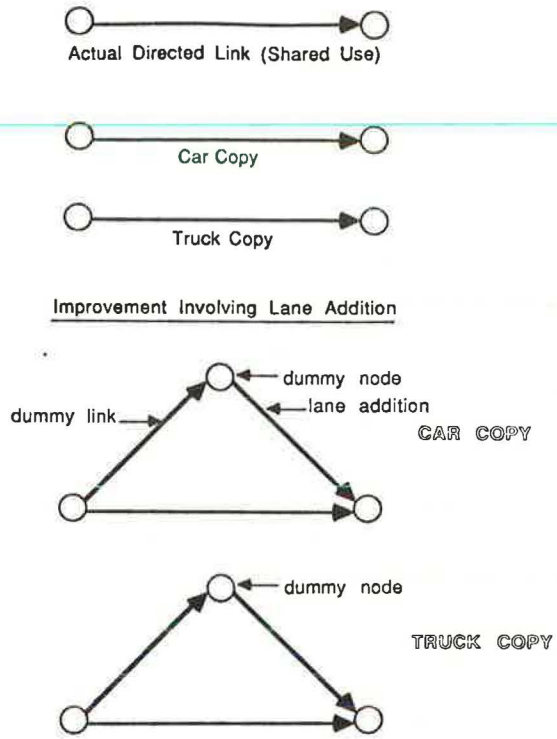

FIGURE 2 Special network representation of truck-lane addition.

also defined for each of the car and truck copies, as shown in Figure 2. The additional links are a dummy link from the start node i to the dummy node and a link from the dummy node to the end node j that represents the actual lane addition, which is included in each copy to allow either cars or trucks (or both) to use it. However, if it is desired to restrict its use to trucks only, the preceding dummy link in the car network copy will be associated with a very large positive cost (travel time), which effectively prohibits cars from using it (and thereby from getting onto the added lane). If, on the other hand, it is desired to allow cars to use this new lane, the cost is set equal to zero, and cars are therefore allowed to consider using the additional lane in their route choice, because no penalty is associated with the dummy link. Of course, the travel time that they will experience on that link (representing the new lane) is given by the associated performance function, which properly accounts for the interaction with the existing lanes as well as with the truck flows on the existing and new lanes (represented by the links defining the truck network copy).

To illustrate the foregoing mechanism, its application to the four improvement options of interest is described next. Consider existing directed link a and let a_A denote the copy of link a for car traffic; a_T, the copy of link a for truck traffic; a_A', the potential lane addition link for cars; a_T', the potential lane addition link for trucks; and a_{Ad} and a_{Td}, the corresponding dummy links in the car and truck copies, respectively.

Furthermore, define the link travel times t_{aA}, t_{aT}, $t_{a'A}$, $t_{a'T}$, and t_{aTd} on the corresponding links (see Figure 3). The travel times are, in the general case, related to the flow vector $\mathbf{X}_a = \{\mathbf{X}_{aA}, \mathbf{X}_{aT}, \mathbf{X}_{a'A}, \mathbf{X}_{a'T}\}$ comprising the respective flows on the just-defined links. Note that t_{aAd} and t_{aTd} are equal to either M (a

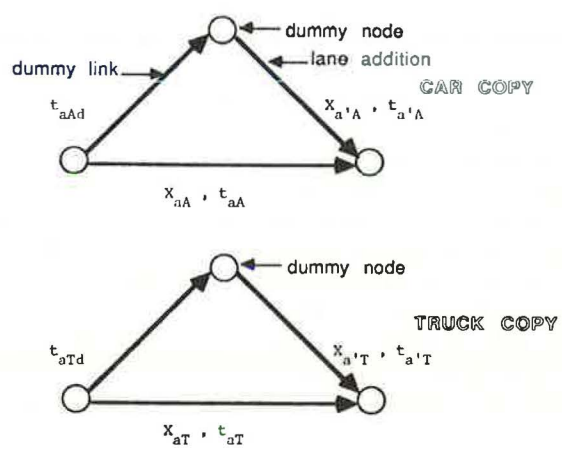

FIGURE 3 Notation for link improvement representation.

very large positive number) or 0, depending on the access rule for the additional lane, as seen hereafter. The performance functions for the nondummy links are denoted by $t_{aA}(.)$, $t_{a'A}(.)$, $t_{aT}(.)$, and $t_{a'T}(.)$, respectively. The operational schemes associated with each improvement option are translated through the specific dependence of the above link travel times on the components of the flow vector $\mathbf{X}_a$.

Option 1

Under Option 1, all traffic is allowed on the new lane, with no changes (i.e., still all traffic) on the existing lanes. Therefore,

$$t_{aA} = t_{a'A} = t^*_{aA}(X_{aA} + X_{a'A}, X_{aT} + X_{a'T})$$
$$t_{aT} = t_{a'T} = t^*_{aT}(X_{aA} + X_{a'A}, X_{aT} + X_{a'T})$$
$$t_{aAd} = t_{aTd} = 0$$

In other words, the average travel time on links a_A and a_A' is effectively the same (i.e., there is no basis for distinguishing between the performance of these lanes) and is dependent on the total automobile and total truck flows, respectively, on the upgraded highway link a; similarly for a_T and a_T'. The functions $t^*_{aA}(.)$ and $t^*_{aT}(.)$ denote the modified performance functions for the upgraded facility.

Option 2

Under Option 2, the new lane is an exclusive truck lane, with no car traffic allowed on that lane. No other restrictions apply on the existing lanes. This translates into

$$t_{aA} = t_{aA}(X_{aA}, X_{aT})$$
$$t_{a'A} = t_{aAd} = M \text{ (very large positive number)}$$

$$t_{aT} = t_{aT}(X_{aA}, X_{aT})$$
$$t_{a'T} = t_{a'T}(X_{a'T}) \quad t_{aTd} = 0$$

Effectively then, the new exclusive truck facility is assumed to operate virtually independently from the existing lanes. Therefore, travel time for trucks on the truck facility depends only on the flow of trucks using that facility. Naturally, travel time for cars on that facility (as well as on the corresponding dummy link) is set to a very large positive number to prohibit its use by cars.

Option 3

Under Option 3, the new lane operates as an exclusive truck facility. However, this is coupled with the restriction of truck traffic from using the other (existing) lanes of the highway. The corresponding relationships are

$$t_{aA} = t_{aA}(X_{aA}) \quad t_{aT} = 0$$
$$t_{a'A} = t_{aAd} = M$$
$$t_{a'T} = t_{a'T}(X_{a'T}) \quad t_{aTd} = 0$$

Here, the two types of facilities (existing lanes and new exclusive truck lane) operate virtually independently from one another, thus the absence of cross-link effects in the corresponding performance functions.

Option 4

Under Option 4, an exclusive new car-only lane is built; no other restrictions apply. This is represented by

$$t_{aA} = t_{aA}(X_{aA}, X_{aT})$$
$$t_{a'A} = t_{a'A}(X_{a'A}) \quad t_{aAd} = 0$$
$$t_{aT} = t_{aT}(X_{aA}, X_{aT})$$
$$t_{a'T} = t_{aTd} = M$$

In essence, under Option 4, the new exclusive car lane is assumed to operate virtually independently of the existing lanes, with no truck interference, whereas the same relationships remain in effect in the existing, shared-use lanes.

The foregoing equations illustrate the functional dependence between the various travel time and flow components associated with the particular network configuration introduced in this study to represent the truck-related link improvements of interest. The specific functional forms and parameter values of the link performance functions for different facility types should ideally be determined from actual field observations. The calibration of such functions was not within the scope of the present study. In the numerical experiments conducted to test the algorithm, performance functions of the well-known Bureau of Public Roads *(8)* form were modified to reflect the interaction between cars and trucks. For example, for a common link a shared by cars and trucks, the basic functional form is

$$t_{aA} = t^o_{aA} \{1 + ALPHA \cdot [(X_{aA} + h \cdot X_{aT})/C_a]^{BETA}\}$$

where

t^o_{aA}	=	free-flow time of passenger cars on link a,
$ALPHA, BETA$	=	parameters capturing the sensitivity of travel time (cost) to flow,
C_a	=	parameter that captures the "capacity" of link a in passenger-car equivalents per unit time, and
h	=	parameter that transforms the effect of trucks into passenger-car equivalents.

The parameter values used are adapted from published values reported in the literature and ensure that travel cost increases monotonically with the flow components. In the absence of observationally developed performance functions that explicitly address car–truck interactions, it is believed that the previous well-known functional form and the standard traffic engineering approach for treating trucks are adequate for the purpose of implementing the assignment methodology and performing the desired tests of the diagonalization algorithm for this type of problem.

STEPS OF THE DIAGONALIZATION ALGORITHM

Before the results of the numerical tests are described, it is useful to present a conceptual overview of the diagonalization algorithm to define the shortcut or streamlining strategies tested

in this study. The difficulty with the network equilibrium problem with asymmetric link interactions is that, unlike problems with symmetric or no interactions, it does not admit an equivalent mathematical programming formulation that can be readily solved to obtain the desired equilibrium flow pattern. The diagonalization algorithm is a so-called "direct" algorithm that is not guided by the minimization of some global objective function in its attempt to converge on a solution. Proofs exist that it will only converge on a solution that satisfies the desired equilibrium conditions *(3, 9)*.

Essentially, the diagonalization algorithm is an iterative procedure that involves solving a series of tractable single-class user equilibrium programs. Letting $t_a^n = f_a(X_a^n, \mathbf{X}_{\bar{a}}^n)$ denote the performance function for link a, where $t_{\bar{a}}$ is the travel time and X_a^n is the flow on link a at the nth interaction of the algorithm, whereas $\mathbf{X}_{\bar{a}}^n$ is a vector of flows on all links other than a that affect the travel time on that link. The vector $\mathbf{X}_{\bar{a}}^n$ will typically contain at least the flow on the link corresponding to the "other" vehicle class sharing the physical right-of-way of link a; that is, if a is the passenger-car copy of a particular highway link, $\mathbf{X}_{\bar{a}}^n$ will include at least the flow on the truck copy of that same physical highway link. In addition, $\mathbf{X}_{\bar{a}}^n$ may also contain flows on the links defined especially to study the impact of truck lanes and other truck-related improvement options, as described in the previous section.

Note that the effect of X_a on t_a is referred to as a "main" effect, whereas the effect of the components of $\mathbf{X}_{\bar{a}}$ on t_a are called "cross-link" effects. The diagonalization algorithm requires, at the nth iteration, that all cross-link effects be fixed at their current levels, with only the main effect allowed to vary in the solution of the equilibrium problem at any given iteration. In other words, at the nth iteration of the diagonalization algorithm, t_a^n and X_a^n are solved for jointly such that $t_a^n = f_a(X_a^n, \mathbf{X}_{\bar{a}}^{n-1})$ and user equilibrium conditions are satisfied assuming that the cross-link effects are fixed at their values from the $(n-1)$th iteration. The next iteration, if convergence is not yet achieved, will then fix the cross-link effects at their values from the nth iteration in solving for t_a^{n+1} and X_a^{n+1}.

The steps of the algorithm can be summarized as follows:

Step 0: Initialization; find a feasible link-flow vector.

Step 1: Diagonalization; at the nth iteration, solve a user equilibrium subproblem assuming that cross-link effects are fixed. This yields a link flow pattern X_a^n and associated link travel times t_a^n for all links a.

Step 2: Convergence test; if $X_a^n \cong X_a^{n-1}$, for all a, then convergence is reached. X_a^n is the desired solution. Otherwise, set $n = n + 1$ and go back to Step 1.

As can be seen, the diagonalization algorithm involves a number of iterations, referred to as outer iterations to distinguish them from the inner iterations that must be performed in solving the subproblem in Step 1. This subproblem is solved by using the F-W or convex combinations algorithm, the steps of which can be summarized as follows:

Step 0: Initialization; perform all-or-nothing assignment (by assigning all flows from a given origin to a particular destination to the shortest path between the two points) based on the free-flow or uncongested link travel times.

Step 1: Update travel times by using the link performance functions; only the main effects due to each link's own flow are allowed to affect that link's travel time. Cross-link effects remain constant throughout the solution of this subproblem, fixed at their values from the previous outer iteration.

Step 2: Direction finding; after solving a shortest-path problem based on the updated travel times from Step 1, perform all-or-nothing assignment from each origin to all destinations.

Step 3: Line search; find optimal move size parameter to update the current link flows by combining them with the flows generated in the direction-finding step (Step 2).

Step 4: Update the link flows by using the move size parameter calculated in Step 3.

Step 5: Convergence test; if the link flows generated in Step 4 are about equal to the flows from the previous iteration, convergence is reached and the subproblem is solved; otherwise, increment the iteration counter and go back to Step 1.

A more detailed mathematical presentation of the algorithm has been presented by Sheffi *(3)* and by Mouskos et al. *(2)* for the problem context of interest here. The main point for this discussion is that at each (outer) iteration of the diagonalization algorithm, a number of inner iterations need to be performed. Each of these inner iterations requires the solution of a shortest-path problem, from each origin to all destinations, which can be quite demanding computationally. However, by noting that the flow pattern of only the last outer iteration needs to be determined accurately and that the accuracy of the solution to that subproblem improves only marginally with each additional inner iteration, a streamlining of the algorithm has been proposed by Sheffi *(3)* whereby only one inner iteration is performed per outer iteration. Although more outer iterations will be required to reach convergence, the total number of (inner) iterations involving shortest-path calculations will in most cases be less than what it would have been in the algorithm's original form (i.e., if the diagonalized subproblem is solved to convergence in every outer iteration). This streamlined version has not, however, been sufficiently tested on realistic networks to ascertain its relative computational efficiency compared with other possible streamlining strategies.

Sheffi's streamlining strategy is generalized here by considering a family of shortcut strategies, each characterized by a different value of the (maximum) number of inner iterations per outer iteration. Numerical tests are conducted for values ranging from 1 to 10 inner iterations, in order to determine whether there is a "best" value that appears to apply in most cases for the type of network application of interest. The test networks and experimental conditions considered in the numerical work are described in the next section.

In addition to testing the foregoing shortcut strategies, another objective is to establish that the algorithm does indeed converge for this type of application, because, as noted earlier, convergence is not guaranteed a priori. It can be noted in this regard that the known sufficient conditions for convergence require that the cross-link effects resulting from the interaction between the various user classes be relatively weak compared with the main effect. For the problem of interest, this condition would require that the marginal effect of trucks on the travel time experience by cars sharing the same facility be much smaller than the corresponding marginal effect of cars. Such a

requirement cannot be expected to hold for car–truck interactions and will therefore violate the foregoing sufficient conditions. For this reason, experience with the algorithm in this problem context is quite valuable in assessing its applicability.

COMPUTATIONAL TEST RESULTS

The Experiments

The performance of the diagonalization algorithm and the shortcut strategies was examined in several numerical experiments conducted on three different test network configurations. Network 1 is an arbitrary hypothetical network, whereas Networks 2 and 3 are developed from the Texas highway network. Network 2 is a condensed abstraction of the Texas highway system, developed as the principal network for methodological development and testing. It reflects the essential features of the Texas system in terms of the location of the major origins and destinations, as well as the principal transport arteries, with a minimum of unessential detail. The motivation for this condensed version is of course to limit cost and effort associated with methodological development yet to obtain sufficiently meaningful insights for the large-scale implementation of the algorithm. However, less intensive testing was also performed on Network 3, which is a full-scale representation of the Texas network. As seen from the following summary of characteristics of each network, Network 3 contains an order of magnitude more nodes and links than does Network 2:

	No. of OD Pairs	*No. of Nodes*	*No. of Links*
Network 1	220	78	202
Network 2	364	128	336
Network 3 (Texas network)	364	1,400	3,912

As explained in the previous section, the basic experiment performed is to apply the diagonalization algorithm with different constraints on the maximum number of inner iterations of the F-W algorithm in solving the diagonalized subproblems. The values tested included all integer values from 1 to 10 inner iterations (per outer iteration) for Networks 1 and 2, and from 1 to 5 for Network 3 (values greater than 5 were not run for this large network, primarily because of cost considerations).

These experiments were performed both for cases of "closed" improvement options and for "open" improvement options for Networks 2 and 3, and only for "closed" options for Network 1. Under the "closed" improvement options, and referring to the three-link representation of improvement options presented earlier, the additional-lane links were not available for either cars or trucks (i.e., a high-cost M is placed on the dummy access links for both car and truck copies). Under the "open"improvement options, it is assumed that exclusive lanes are available to both cars and trucks, respectively, in addition to the common existing lanes.

In addition, and for Network 2 only, the experiments were performed under four different congestion levels. The mechanism for controlling this factor is the parameter C_a in the link performance functions. With the origin-destination (OD) matrices unchanged, four levels of this parameter were tested on the configuration of Network 2: the reference, or actual, value C; $0.5C$; $0.8C$; and $4.0C$.

The results of these experiments are summarized next.

Summary of Results

For each test case, two principal figures of merit were considered: the total number of internal iterations until convergence and the corresponding CPU time, which is virtually perfectly correlated with the total number of iterations (for a given network configuration and capacity level).

Table 1 is a summary of the results for each network configuration under the experimental conditions described. It presents the ranking of each shortcut strategy based on its relative performance (with the best-ranked first) for the various test cases. In addition, in that same table, the difference between the total number of (inner) iterations required for convergence and the corresponding number for the optimal (minimum) strategy is given in parentheses next to the ranking. Table 2 presents the total number of inner iterations until overall convergence for Network 2 under the open and closed improvement options. Table 3 presents similar information for Network 3 (the full-scale Texas network).

The principal conclusion from these results is that considerable reduction in computational effort can be achieved by constraining the maximum number of inner iterations in solving each diagonalized subproblem. Table 1 indicates that no single strategy (i.e., value of the maximum number of inner iterations) consistently outperforms all others. However, a strong case can be made for not going beyond three inner iterations under any circumstance, because using one, two, or three inner iterations ranked in the top three in most cases. Of the nine tests conducted with each strategy on Networks 1 and 2, the two-iteration strategy performed best five times, the three-iteration strategy performed best twice, and using one and six iterations performed best once each. For the two tests conducted on the large Texas network (Network 3), the one-iteration strategy performed best when the improvement options were open, whereas the two-iteration strategy performed best when the improvement options were closed (Table 3). The latter is more representative of actual traffic conditions because the assumption of both car and truck exclusive lanes (associated with all physical highway segments) in the "options open" scenario is intended as an extreme case. It can also be noted from the results in Table 2 (for Network 2) that the best performance of the one-iteration strategy occurred exclusively under the "options open" scenario. Table 1 further indicates that the one-iteration strategy was ranked second six times, with a corresponding deviation from the "best" strategy ranging from one to eight total iterations.

When not ranked best, the deviation (in terms of total number of iterations) of the two-iteration strategy from the corresponding best strategy ranged from 1 to 26 iterations, whereas that of the three-iteration strategy ranged from 5 to 53. However, the upper bounds of these ranges of deviation (26 and 53, respectively) occurred in the same test case. The latter used Network 2 under extremely low congestion levels (options open and capacity C_a equal to 4 times the reference

TABLE 1 SUMMARY OF RESULTS; RANK-ORDER POSITION OF SHORTCUT STRATEGIES

Shortcut Strategy[a]	Network 1	Network 2								Network 3	
		Options Open				Options Closed				Options Open	Options Closed
		1.0C	0.8C	0.5C	4.0C	1.0C	0.8C	0.5C	4.0C		
1	1(0)	2(1)	2(2)	2(1)	7(56)	2(8)	2(8)	2(7)	3(6)	1(0)	2(7)
2	2(12)	1(0)	1(0)	4(8)	3(26)	1(0)	1(0)	1(0)	2(1)	2(7)	1(0)
3	3(21)	5(9)	3(5)	1(0)	5(53)	3(15)	3(9)	3(17)	1(0)	4(10)	3(17)
4	5(30)	8(13)	7(14)	3(6)	2(2)	4(25)	4(13)	4(18)	6(21)	3(8)	5(22)
5	4(29)	3(6)	6(13)	4(8)	6(53)	6(31)	5(25)	7(34)	4(8)	5(12)	4(20)
6	6(46)	5(9)	4(9)	5(9)	1(0)	8(36)	5(25)	10(51)	5(12)		
7	7(55)	7(11)	8(15)	7(13)	10(72)	7(34)	6(29)	6(25)	4(8)		
8	8(66)	6(10)	5(11)	7(13)	4(42)	5(26)	8(46)	5(24)	4(8)		
9	9(78)	9(15)	9(16)	6(11)	9(62)	7(34)	7(39)	8(42)	4(8)		
10	10(80)	4(8)	7(13)	7(13)	8(58)	9(46)	9(48)	9(45)	4(8)		

NOTE: Difference between total number of iterations required for convergence and corresponding number for the optimal strategy is given in parentheses.
[a]Maximum number of internal iterations per outer iteration.

TABLE 2 TOTAL NUMBER OF INTERNAL ITERATIONS: NETWORK 2

Shortcut Strategy[a]	Capacity Value			
	1.0C	0.8C	0.5C	4.0C
Options Open				
1	16	13	14	79
2	15	11	21	49
3	24	16	13	76
4	28	25	19	25
5	21	24	21	76
6	24	20	22	23
7	26	26	26	95
8	25	22	26	65
9	30	27	24	85
10	23	25	26	81
Options Closed				
1	21	25	24	10
2	13	17	17	5
3	28	26	34	4
4	38	30	35	25
5	44	42	51	12
6	49	42	68	16
7	47	46	42	12
8	39	63	41	12
9	47	52	59	12
10	59	65	62	12

[a]Maximum number of internal iterations per outer iteration.

TABLE 3 SUMMARY OF RESULTS FOR NETWORK 3

Shortcut Strategy[a]	Total No. of Internal Iterations Until Convergence	
	Options Open	Options Closed
1	16	32
2	23	25
3	26	42
4	24	47
5	28	45

[a]Maximum number of internal iterations per outer iteration.

value). Excluding that test case, the upper bounds of the deviation ranges become 17 and 25 iterations for the two- and three-iteration strategies, respectively.

As has been seen, Sheffi's one-iteration streamlining strategy is not necessarily nor most frequently the most efficient strategy. It is, however, ranked second best in many cases and definitely provides a considerable improvement over the complete version of the algorithm. In the problem context used here, it seems that the two-iteration strategy is the most frequent best performer, though one cannot guarantee a priori that it will outperform the others in any given application.

The analysis of the CPU time required for each test case indicated that, as expected, the cost per iteration will increase with the number of OD pairs and with the size (number of nodes and links) of the network. For instance, the average cost per iteration for Network 3 is 7.51 sec compared with 0.29 and 0.53 sec for Networks 1 and 2, respectively. Interestingly, although the cost per iteration is obviously higher for the large Texas network, the total number of iterations required for this network was not much different from that of its reduced abstracted version (Network 2), thereby validating the premise of using the smaller network as a laboratory for computational testing before addressing the full-scale network. This approach will be pursued for evaluating heuristics for a special version of the network design problem (to determine optimal truck-related improvements in the network), for which extensive testing on the large network is prohibitive.

Another result that is quite important to this study's objective is the fact that convergence was reached in all tests conducted for all three network configurations and experimental conditions considered. It is notable that convergence was achieved despite the fact that the known sufficiency condition was violated in this type of application, thereby confirming the applicability of the algorithm to problems involving asymmetric interactions between cars and trucks on highway links. However, it must be noted that uniqueness of the solutions cannot be guaranteed.

Further details on the results of these experiments, as well as on the application of the diagonalization algorithm for the system optimal assignment problem, can be found in reports by Mouskos et al. *(2)* and by Mahmassani and Mouskos *(10)*.

CONCLUSION

This paper has addressed the applicability of the diagonalization algorithm to the evaluation of truck-related link improvements in a highway network. This problem involves asymmetric interaction between cars and trucks sharing the right-of-way, posing potentially serious methodological questions that need to be resolved in the problem context of interest. The results were generally positive; they indicated that the algorithm converged in most situations even though it did not conform to the only known sufficient conditions for convergence, which appear to be too strict and far from necessary.

In addition, it was demonstrated that considerable savings can be achieved by using shortcut strategies in implementing the diagonalization algorithm. Sheffi's *(3)* streamlined version of the algorithm, which uses only one inner F-W iteration in solving the diagonalized subproblems, was shown to perform considerably better than the unmodified algorithm and some of the other strategies tested here. However, it was not uniformly the best streamlining strategy. In these tests, the two-iteration strategy (using two internal F-W iterations per outer iteration) performed best in the majority of cases; however, the one- and three-iteration strategies were often acceptable second- or third-best strategies. It is nevertheless difficult to determine a priori which strategy will perform best in a particular situation. Additional testing may be necessary if more specific practical guidelines are to be developed in this regard.

A general mechanism was introduced for the network representation of an array of truck-related highway improvements, consisting not only of physical capacity expansion, but also of operating strategies in the form of lane access restrictions for either cars or trucks. This representation allows the use of the assignment methodology to support the analysis, identification, and selection of specific sections of the highway network as candidates for the construction and implementation of particular types of truck-related improvements such as exclusive truck lanes or facilities, which are of great interest to transportation and highway agencies.

Naturally, it is highly desirable to develop link performance functions that explicitly capture the interactions between cars and trucks, based on actual observations of such traffic. Limited attempts along these lines have been described by Kim *(11)*. Finally, it should be noted that because the primary interest of the present work is to examine the operational implications on the highway system, the OD patterns of both cars and trucks were assumed known and given. However, for studies with a strategic orientation, where the broader implications for the economy, carrier operations, or shipper logistics are of concern, more general formulations along the lines discussed by Friesz et al. *(12)* should be considered.

ACKNOWLEDGMENTS

The diagonalization code used in this study was modified from an F-W User Equilibrium program listing provided by Fred Mannering of Pennsylvania State University, who in turn modified a code supplied by Stella Defermos of Brown University. The collaboration of John J. Massimi, formerly a graduate assistant working with the study team, in defining the improvement options is acknowledged. The work presented in this paper was performed in conjunction with Research Project 356, Study of Truck Lane Needs, funded by the Texas State Department of Highways and Public Transportation. The authors, of course, remain solely responsible for the contents of this paper.

REFERENCES

1. H. S. Mahmassani, C. M. Walton, K. Mouskos, J. J. Massimi, and I. L. Levinton. *A Methodology for the Assessment of Truck Lane Needs in the Texas Highway Network.* CTR Report 356-3F. University of Texas at Austin, 1986.
2. K. Mouskos, H. S. Mahmassani, and C. M. Walton. *Network Assignment Methods for the Analysis of Truck-Related Highway Improvements.* CTR Report 356-2. University of Texas at Austin, 1986.
3. Y. Sheffi. *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods.* Prentice-Hall, Englewood Cliffs, N.J., 1984.
4. S. C. Dafermos. Relaxation Algorithm for the General Asymmetric Traffic Equilibrium Problem. *Transportation Science*, Vol. 16, No. 2, 1982, pp. 231–240.
5. T. Friesz. Transportation Network Equilibrium, Design and Aggregation: Key Developments and Research Opportunities. *Transportation Research A*, Vol. 19A, No. 516, 1985, pp. 413–427.
6. S. C. Dafermos. An Iterative Scheme for Variational Inequalities. *Mathematical Programming*, Vol. 26, No. 1, 1983, pp. 40–47.
7. S. Lawphongpanich and D. W. Hearn. Simplicial Decomposition of the Asymmetric Traffic Assignment Problem. *Transportation Research B*, Vol. 18B, No. 2, 1984, pp. 123–133.
8. *Traffic Assignment Manual.* Bureau of Public Roads, U.S. Department of Commerce, 1984.
9. M. Abdulaal and L. J. Leblanc. Methods for Combining Modal Split and Equilibrium Assignment Models. *Transportation Science*, Vol. 13, No. 4, 1979, pp. 292–314.
10. H. S. Mahmassani and K. Mouskos. Some Numerical Results on the Diagonalization Algorithm for Network Assignment with Asymmetric Interactions Between Cars and Trucks. *Transportation Research B* (in press).
11. Y.-G. Kim. *Performance Functions for Highway Links with Asymmetric Interactions Between Cars and Trucks.* M.S. thesis. Department of Civil Engineering, University of Texas at Austin, 1986.
12. T. L. Friesz, R. L. Tobin, and P. T. Harker. Predictive Intercity Freight Network Models: The State of the Art. *Transportation Research A*, Vol. 17A, No. 6, 1983, pp. 409–417.