Transpn. Res.-B Vol. 20B, No. 3, pp. 259–265, 1986
Printed in Great Britain.

0191-2615/86 $3.00+.00

SHORT NOTES AND RESEARCH COMMUNICATIONS

A BILEVEL PROGRAMMING ALGORITHM FOR EXACT SOLUTION OF THE NETWORK DESIGN PROBLEM WITH USER-OPTIMAL FLOWS*

LARRY J. LEBLANC
Owen Graduate School of Management, Vanderbilt University, 401 21st Ave. South,
Nashville, TN 37203

and

DAVID E. BOYCE
Department of Civil Engineering, University of Illinois at Urbana-Champaign, 208 N. Romine Street,
Urbana, IL 61801

(*Received* 30 *September* 1985; *in revised form* 27 *November* 1985).

1. INTRODUCTION

In transportation planning, a large class of problems is characterized by multiple levels of decision making. Examples include selection of links for capacity improvements (network design), pricing of freight transportation services and traffic signal setting. In each of these problems, government or industry officials make one set of decisions, which seek to improve the network's performance. At another level, users make choices with regard to route, mode, origin-destination, etc. Although the responses of users to a given improvement can be predicted, their decisions cannot be dictated. For example, officials choose locations for network improvements, and drivers then choose whatever routes they perceive to be best. Likewise, freight carriers determine shipment rates and transit times, and shippers choose the best mode of shipment.

In this paper, we show how to formulate such problems as *bilevel programming models*, and propose solution algorithms for evaluation. For the transportation network design problem, we show that this model can be solved exactly for networks with a few hundred nodes. The general form of a bilevel program is

$$\underset{y}{\text{MIN}}\ F(x, y)$$

where x is optimal for

$$\underset{x}{\text{MIN}}\, f(x, y)$$

$$\text{ST}\quad G(x, y) \geqq b$$

The objective MIN $F(x, y)$ is referred to as the *upper* problem, and the *lower* problem is defined as MIN $f(x, y)$ for fixed y. The variables y in the upper problem describe the controls available to network planners; the lower-problem variables x describe the mode-choice, route-choice, etc. decisions of the users. Our interest in bilevel programs originated from the papers by Bard and Falk (1982) and Bard (1983a,b).

In the next section, we formulate a *linear* bilevel programming model of the network design problem with user-optimal driver behavior and continuous link improvement variables. A piecewise linear model of this problem is considered because there is a well-developed theory

*This research was supported by the 1985 Dean's Fund of the Owen Graduate School of Management, Vanderbilt University.

for linear bilevel programs. This formulation leads to insights into the type of model appropriate for the network design problem with user-optimal driver behavior. The corresponding algorithm can be used to solve *exactly* network design problems with moderate sized networks. This is the first model of the network design problem with user-optimal driver behavior which can be solved exactly for networks with hundreds of link-improvement variables. We then show how to use more efficient nonlinear programming computational techniques to obtain highly accurate approximations for large-scale networks.

2. THE NETWORK DESIGN PROBLEM

The problem of allocating capacity increases to links of a transportation network is generally known as the *network design problem*; see Magnanti and Wong (1984) for a complete review. One version of the problem is the following. A network is given with links representing roads and nodes representing intersections and locations where trips originate and terminate. A trip table specifies the number of trips between each pair of nodes. We wish to determine which links to improve, and possibly which new links to add, so that total user costs are minimized. A budget constraint may limit the total cost incurred for network improvements.

It is well known that the generalized cost of travel increases nonlinearly with traffic flow. Travel cost functions are empirical functions, and any analytical representation is necessarily an approximation. The most widely-used analytical function appears to be the FHWA function (Comsis, 1973). Alternative ones include piecewise linear functions and Davidson functions (Taylor, 1977).

In the network design model formulated in this paper, we define the total travel cost for a link to be the user cost per vehicle multiplied by the number of vehicles using the link. We assume that the total travel cost on each link can be accurately measured with a piecewise linear function with M pieces or segments. In principle, if M is sufficiently large (20–30), a piecewise linear function is as accurate as any other analytical representation. An example of such a function with four pieces is shown in Fig. 1.

The following notation describes the travel cost functions:

M_l = number of pieces or segments in the piecewise linear total travel cost function for link l;
K_{lm} = capacity of piece m of link l;
c_{lm} = slope of piece m of link l;
x_{lm} = flow on piece m of link l;
x_l = total flow on link l.

The piecewise linear total travel cost function for link l is then $\sum_{m=1}^{M_l} c_{lm}x_{lm}$, where for each piece m, $0 \leqq x_{lm} \leqq K_{lm}$ and $x_l = \sum_{m=1}^{M_l} x_{lm}$.

In network modeling, it is generally assumed that driver behavior and the resulting traffic flows are suitably represented by equilibrium or user-optimum behavior. Therefore, we formulate a network design model which assumes user-optimal behavior of drivers.

Bard (1983b) has proposed an algorithm for solving linear bilevel programs, i.e., ones in which F, f, and G are all linear functions. In order to investigate the use of this algorithm, we now show how to formulate such a linear bilevel model of the network design problem with

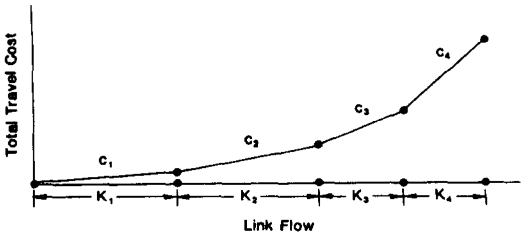

Fig. 1. Piecewise linear total travel cost function.

user-optimal flows. This requires travel cost vs. flow functions which are linear before and after improvements. We first define the following notation:

J = number of destination nodes;
x_l^j = flow on link l with destination j;
T_{ij} = required number of trips between origin node i and destination node j;
A_i = set of links pointing out of node i (after i);
B_i = set of links pointing into node i (before i).

Notice that x_{lm} is the total flow on piece m of link l going to all destinations, while x_l^j is the total flow on all pieces of link l going to destination j. The trips T_{ij} are regarded as constants, although this assumption can be relaxed. The conservation of flow constraints are

$$T_{ij} + \sum_{l\epsilon B_i} x_l^j = \sum_{l\epsilon A_i} x_l^j \quad \text{for all nodes } i \text{, for all destinations } j$$

(Flow originating at i + flow entering i equals flow leaving i). (1)

The total flow on link l is the sum of the flows traveling to the various destinations, $x_l^1 + x_l^2 + \ldots + x_l^J$. Also, the total flow on link l equals the sum of the flows on the individual link pieces, $x_{l1} + x_{l2} + \ldots + x_{lM_l}$. Thus, additional constraints are

$$\sum_{m=1}^{M_l} x_{lm} = \sum_{j=1}^{J} x_l^j \quad \text{for all links } i \quad \text{(Link flow definitions).} \tag{2}$$

In the network design model that we propose, the decision variables y_l denote the units of capacity added to link l. We model link improvements by assuming that when y_l units of capacity are added to link l, each piece m receives $\alpha_{lm} y_l$ capacity units, where the parameters α_{lm} are exogenously specified with $\Sigma_m \alpha_{lm} = 1$. Thus, the link capacity constraints are

$$x_{lm} \leqq K_{lm} + \alpha_{lm} y_l \quad \text{for each piece } m \text{ of each link } l \quad \text{(Link capacity constraints).} \tag{3}$$

For example, suppose that a travel-cost function has three linear pieces with 5, 10, and 10 units of capacity, respectively. Thus, the road has 25 units of capacity before improvement. If $\alpha_1 = 0.2$, $\alpha_2 = 0.3$, and $\alpha_3 = 0.5$, and if 10 units of capacity are added to the link, then piece 1 receives 2 additional units, and pieces 2 and 3 receive 3 and 5 units, respectively. Figure 2 shows the new travel cost function superimposed over the old one.

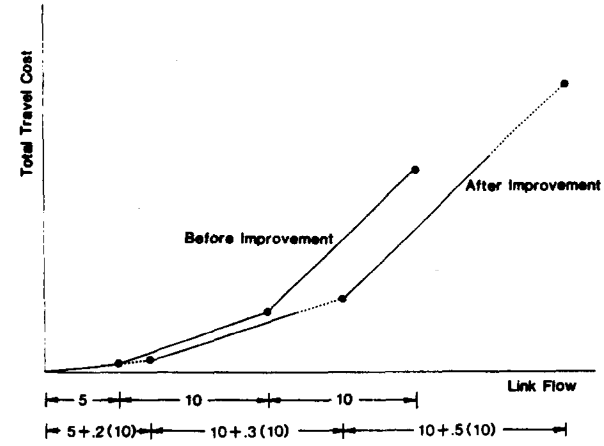

Fig. 2. Total travel cost functions before and after adding ten units of capacity.

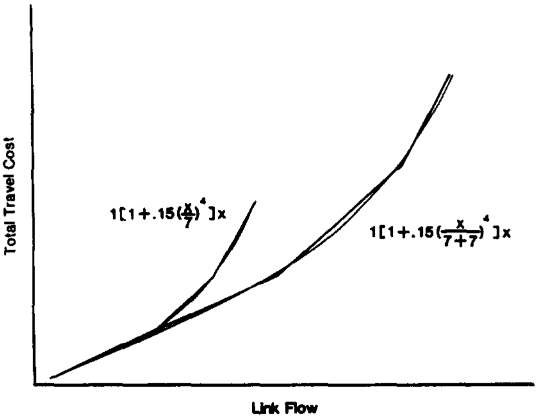

Fig. 3. FHWA and piecewise linear travel cost functions before and after adding seven units of capacity.

Figure 3 compares this treatment of link improvement with the commonly used FHWA method. Observe that both approaches result in nearly the same measure of travel time after improvements.

As illustrated by Fig. 4, improvements which increase a link's speed limit can be modeled with a piecewise linear travel cost function in which the first piece has zero capacity. This is not possible (except with heuristics) using conventional travel cost functions.

We denote the cost of network improvements by $\Sigma_l\ b_l y_l$, where b_l is the unit cost of improvements on link l. We assume that the cost of link improvements is a linear function of the improvement levels y_l. Alternatively, convex piecewise linear improvement-cost functions can be used.

To ensure that the link flows are user-optimal for any possible network improvements, we define a lower problem requiring that the flows minimize the piecewise linear approximations to the integrals of the improved user-cost functions, with $\bar{c}_{lm}$ as the coefficient of piece m. The user-optimal flows are determined by

$$\text{MIN} \sum_{lm} \bar{c}_{lm} x_{lm}, \tag{4}$$

subject to the conservation of flow, link flow definition and link capacity constraints (with y fixed) (1)–(3). Note that the $\bar{c}_{lm}$ are the slopes of the piecewise linear *integrals* of the user-cost functions, while the c_{lm} are slopes of the total-cost functions. By using a sufficiently large number of segments, the piecewise linear representations of the integrals of the user-cost functions become arbitrarily accurate. Since the user-cost function itself inherently contains some inaccuracy, regardless of its analytical form, there is essentially no loss in accuracy with this approach.

The bilevel programming model of the network design problem with user-optimal flows is to minimize travel costs plus improvement costs converted to equivalent units:

$$\underset{y}{\text{MIN}} \sum_{lm}\sum c_{lm} x_{lm} + \gamma \sum_{l} b_l y_l, \tag{5}$$

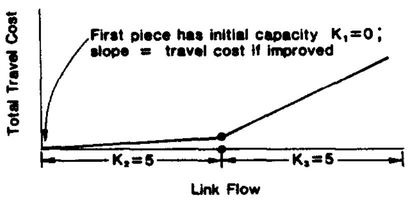

Fig. 4. Travel cost function for which improvements increase speed limit.

where x is optimal for

$$\underset{x}{\text{MIN}} \sum_{lm}\sum \bar{c}_{lm} x_{lm}, \tag{6}$$

subject to

$$T_{ij} + \sum_{l\epsilon B_i} x_l^j = \sum_{l\epsilon A_i} x_l^j \quad \begin{array}{l}\text{for each node } i\\ \text{for each destination } j,\end{array} \tag{7}$$

$$\sum_{m=1}^{M_l} x_{lm} = \sum_{j=1}^{J} x_l^j \quad \text{for each link } l, \tag{8}$$

$$x_{lm} \leqq K_{lm} + \alpha_{lm} y_l \quad \begin{array}{l}\text{for each piece } m \text{ of}\\ \text{each link } l\end{array} \tag{9}$$

This model is similar to a network design model with the approximating system-optimal behavior of drivers studied previously by Morlok et al. (1973). An alternative formulation includes improvement costs in a budget constraint.

Bard (1983b) has shown that any linear bilevel program can be solved by solving a single linear program and then iteratively modifying the objective function and resolving it. In his procedure, the objective function is defined as a convex combination of the upper and lower objective functions. For (5)–(9), this objective function is

$$P(x, y; \lambda) \equiv \lambda \left[\sum_{lm}\sum c_{lm} x_{lm} + \gamma \sum_{l} b_l y_l\right] + (1 - \lambda)\left[\sum_{lm}\sum \bar{c}_{lm} x_{lm}\right] \tag{10}$$

where λ is a fixed parameter. Bard's procedure for model (5)–(9) involves iteratively solving the following linear program, which is equivalent to (10):

$$\underset{x,y,z}{\text{MIN}} \sum_{lm}\sum \bar{c}_{lm} x_{lm} + \lambda z \tag{11}$$

$$\text{subject to } z = \sum_{lm} c_{lm} x_{lm} + \gamma \textstyle\sum_l b_l y_l - \displaystyle\sum_{lm} \bar{c}_{lm} x_{lm}, \tag{12}$$

and subject to (7)–(9). In (11), z is a scalar variable defined by (12). Writing the linear program in this manner facilitates sensitivity analysis with respect to λ, the objective function coefficient of the single variable z.

Bard's algorithm for model (5)–(9) is the following. Set $\lambda = 1$, set the iteration index $k = 1$ and choose the tolerance $\epsilon > 0$.

1. Minimize (11) subject to (12) and (7)–(9) and denote the solution by (x^k, y^k, z^k).
2. Check whether the link flows x^k are user-optimal flows. If so, stop; the solution (x^k, y^k) is an optimal solution to the network design model.
3. Using linear programming sensitivity analysis on the objective function coefficient λ, find $\lambda_{\min} \geqq 0$, the smallest value of λ for which (x^k, y^k, z^k) remains optimal in the linear program. Change λ to $\lambda_{\min} - \epsilon$ and go to step 1.

Step 2 requires checking whether x^k is optimal in the linear program (6)–(9) with y fixed at y^k. Bard's procedure terminates in a finite number of steps, producing the optimal link improvements and user-optimal flows in step 2. The parameter λ is steadily decreased in step 3, but is not decreased below zero.

Observe that the first linear program solved has the parameter $\lambda = 1$. This LP is a network design model with system-optimal flows [LeBlanc and Abdulaal, 1984], since the linearized integrals of the user-cost functions have zero weight. As λ is decreased, the linearized integrals are weighted more heavily, and the solution eventually contains user-optimal flows.

In this bilevel program, the majority of the constraints are the conservation of flow constraints (7) for each node-destination combination in the network. If the network has N nodes, J origin/destination nodes, and L links, each with M pieces, then the linear programs in step 1 would have $NJ + L + ML + 1$ constraints. For example, a network with 600 links with 3 pieces each and 200 nodes, of which 40 are destination nodes, would result in linear programs with 10,401 constraints. A 150-node, 300-link network design model with 3 pieces per link and 40 destination nodes would result in a linear program with 7,201 constraints. Such linear programs can be solved with large computers. However, design problems for networks with thousands of nodes or many linear pieces per link cannot be solved directly with this approach.

As noted by Bard, if any of the LP's solved in step 1 has multiple optimal solutions, all optimal solutions must be checked in step 2. If a linear program does have multiple optimal solutions, this can be a very tedious process. For this reason, and because the literal application of Bard's algorithm is limited to moderate sized networks, we propose a very accurate approximation procedure in the next section.

3. AN EFFICIENT SOLUTION PROCEDURE FOR LARGE NETWORKS

In this section, we propose a solution procedure for networks too large for the corresponding bilevel program to be solved directly. It is well-known that the Frank-Wolfe technique can efficiently solve nonlinear optimization models for the user-optimal route choice problem and the network design problem with system-optimal flows on networks with thousands of nodes (LeBlanc, Morlok and Pierskalla, 1975). For this reason, we suggest using an equivalent optimization model in place of the linear program in step 1 of Bard's algorithm. This equivalent optimization model can be solved very efficiently using the Frank-Wolfe approach.

In this procedure, instead of minimizing the convex combination of the piecewise linear travel cost functions (10), the convext combination of nonlinear increasing functions is minimized. The user-cost function before improvement is $c_l = f_l(x_l, k_l)$ and the user-cost function after improvement is $c_l = f_l(x_l, k_l + y_l)$ where k_l is the capacity of the link. With this procedure, the objective function (11) in step 1 above is change to

$$\operatorname*{MIN}_{x,y} \lambda \left[\sum_l f_l(x_l, k_l + y_l)x_l + \gamma \sum_l b_l y_l \right] + (1 - \lambda) \left[\sum_l \int_0^{x_l} f_l(t, k_l + y_l)\, dt \right] \qquad (13)$$

As in (11) and (12), this objective function is the convex combination of the upper and lower objectives functions of total travel cost and the integral of user cost. Any travel cost functions for which (13) is strictly convex in x and y (such as the FHWA function) can be used in this procedure.

Since minimizing (13) is essentially the same as solving the linear program in step 1 when each piecewise linear travel cost function has a very large number of pieces, the solution to these two models should be very similar. Indeed, both piecewise linear and nonlinear increasing cost functions are simply analytical representations of empirical data.

When the Frank-Wolfe technique is used to minimize (13) instead of (11) in step 1, the conservation of flow constraints (7) are implicitly satisfied, since the subproblem solution involves flow traveling along shortest paths. Furthermore, the link definition constraints (8) are unnecessary with the Frank-Wolfe technique, since the total link flows are accumulated during the process of solving the Frank-Wolfe subproblem. The link capacity constraints (9) are applicable only to the piecewise linear model of the network design problem. The nonlinear model incorporates the relationship between link flows, link improvements and travel cost directly in the objective function. Thus, the link capacity constraints (9) can also be omitted when using the Frank-Wolfe approach. Finally, the definitional constraint (12) is also unnecessary, since the two objectives are combined directly in the objective function (13), instead of indirectly as in (11) and (12). For these reasons, the Frank-Wolfe approximation for Bard's problems involves solving equivalent convex optimization models with only the implicit conservation of flow constraints (7). The computational requirements of each of these nonlinear models are very similar to those of an equilibrium assignment model.

Since the objective (13) is strictly convex in the link flows, each of these equivalent nonlinear models has a unique optimal solution. This eliminates the possible difficulty of searching for all optimal solutions to the linear programs in Step 1 of Bard's algorithm. The model (13) must be solved for different values of λ, such as $\lambda = 1.00, 0.95, 0.90, \ldots$ until the link flows obtained are user-optimal or nearly so. When using the Frank-Wolfe approach, the reoptimization of the new equivalent nonlinear model is very efficiently preformed by starting the Frank-Wolfe procedure from the optimal solution to the previous model solved. This is known to reduce the computational effort by as much as 60%.

4. CONCLUSIONS

The bilevel programming formulation proposed for the network design problem can be readily extended to a large class of transportation planning problems. The piecewise linear bilevel programming model of the network design problem should yield an exact solution for networks with fewer than 200 nodes. For larger networks, near-optimal solutions can be obtained by solving an approximating nonlinear program which is very similar to a user-optimal route choice model. Computational tests of both of the algorithms proposed in this paper are being undertaken.

REFERENCES

Bard J. (1983a) An algorithm for solving the general bilevel programming problem, *Mathematics of Operations Res.* **8**, 260–272.

Bard J. (1983b) An efficient point algorithm for a linear two-stage optimization problem, *Operations Res.* **31**, 670–684.

Bard J. and Falk J. (1982) An explicit solution to the multi-level programming problem, *Computers and Operations Res.* **9**, 77–100.

Comsis Corporation (1973) *Traffic Assignment*, prepared for Federal Highway Administration, U.S. Department of Transportation, Washington, D.C.

LeBlanc L. (1975) An algorithm for the discrete network design problem, *Transportation Science* **9**, 183–199.

LeBlanc L., Morlok E. and Pierskalla W. (1975) An efficient approach to solving road network equilibrium traffic assignment problems, *Transp. Res.* **9**, 309–318.

LeBlanc L. and Abdulaal M. (1984) A comparison of user-optimum vs. system-optimum traffic assignment in transportation network design, *Transp. Res.* **18B**, 115–121.

Magnanti T. and Wong R. (1984) Network design and transportation planning: models and algorithms, *Transportation Science* **18**, 1–55.

Morlok E., Pierskalla, W., Schofer J., Marsten R., Agarwal S., Edwards J., Spacek D., Stoner J., and LeBlanc L. (1973) *Development and application of a highway network design model*, Final report, prepared for Federal Highway Administration, U.S. Department of Transportation, Washington, D.C.

Taylor M. (1977) Parameter estimation and sensitivity of parameter values in a flow-rate/travel-time relation, *Transportation Science* **11**, 275–292.