Computational Challenges Associated with Mixed-Integer Program Formulations of the Transit Network Design Problem

Seyed Mohammad Nourbakhsh, Yanfeng Ouyang, and Michael J. Cassidy

UCB-ITS-RR-2014-1

March 10, 2014
Computational Challenges Associated with Mixed-Integer Program Formulations of the Transit Network Design Problem

Seyed Mohammad Nourbakhsh a, Yanfeng Ouyang a, and Michael J. Cassidy b

a Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA
b Department of Civil and Environmental Engineering, University of California, Berkeley, CA 94720, USA

Abstract
This document reports on recent experimental results using mixed-integer programming models and a commercial solver, CPLEX, to solve select state-of-art discrete formulations of the transit network design problem. Three mathematical models taken from the literature are presented and discussed; two are solved for sets of test cases. The models generally minimize user costs (including in-vehicle travel time) and agency costs (including total distance traveled by buses) subject to a prescribed number of routes. The numerical cases are adapted from a hypothetical mono-centric city under heterogeneous demand. The city is qualitatively similar to, though much smaller than, one examined in Ouyang, et al. (2014) where the network design problem was solved using a continuum approximation approach. The present computational results highlight the burdens and challenges associated with discrete optimization approaches for transit network design.

1. Introduction
Transit network design has been studied extensively. Reviews of the vast literature on this subject can be found in Kepaptsoglou and Karlaftis (2009), Guihaire and Hao (2008), Desaulniers and Hickman (2007), Fan and Machemehl (2004), Zhao and Gan (2003), Ceder and Wilson (1986), and Chua (1984). In this first section, we summarize three typical types of formulations from the literature. The notation generally follows that in the original references, and we make no efforts to unify notation across the three models.

1.1. Set-Covering Model
One classic type of model formulation follows a set covering structure, whereby routes are selected from a predefined set of candidates so as to “cover” passenger travel needs, via either direct service or by transferring across routes. An example of a mixed-integer, nonlinear program formulation of this type is presented in Ceder (2007). Following the notation in that reference, symbols $R$, $TR$, $S$, $N_r$, $N_{tr}$, $N_{sp}$ respectively represent the sets of transit routes, transfer paths, shortest paths, nodes located on route $r$, nodes located on transfer path $tr$, and nodes located on shortest path $sp$. Let $t_{i,j}^r$, $t_{i,j}^{tr}$, and $t_{i,j}^{sp}$ be the average travel time between any nodes $i$ and $j$ on route $r$, on transfer path $tr$ and on its shortest path, respectively. The decision variables include the following: route selection variable $x_r = 1$ if route $r \in R$ is selected in the transit network solution, or 0 otherwise; demand allocation variable $d_{i,j}^r$ (or $d_{i,j}^{tr}$) = 1 if the transit demand from $i$ to $j$ can be handled directly by route $r \in R$ (or by transfer path $tr \in TR$), or 0 otherwise. The model is formulated as follows (Ceder, 2007):

\begin{align*}
\text{(Model I)} & \quad \min \sum_{r \in R} c_r x_r + \sum_{r \in TR} c_{tr} \prod_{r \in TR} x_r \\
\text{s.t.} & \quad \sum_{r \in R} a_{i,j}^r x_r + \sum_{f \in F} a_{i,j}^{tr} \prod_{r \in tr} x_r \geq 1, \forall i, j \in N, \\
& \quad c_r = \sum_{i,j \in N_r} (t_{i,j}^r - t_{i,j}^{sp}), \forall r \in R,
\end{align*}

(1.1) (1.2) (1.3)
\[ c_r = \sum_{i,j} (t_{i,j}^r - t_{i,j}^x), \ \forall tr \in TR, \]  
(1.4)

\[ \sum_{re R} d_{i,j}^r + \sum_{j \in F} a_{i,j}^r = 1, \ \forall i, j \in N, \]  
(1.5)

\[ x_r = \{0,1\}, \ \forall r \in R, \]  
(1.6)

\[ a_{i,j}^r = \{0,1\}, \ \forall r \in R, \text{ for all } i, j \]  
(1.7)

\[ d_{i,j}^r = \{0,1\}, \ \forall tr \in TR, \text{ for all } i, j \]  
(1.8)

The objective function (1.1) minimizes the cost for the set of routes and the passenger travel costs, whether served directly or via transfers. Constraints (1.3) and (1.4) define \( c_r \) and \( c_{tr} \) as the extra cost of a direct route and a transfer path, respectively, as compared against the shortest path costs. Constraint (1.5) guarantees that a link exists for all O-D pairs, via either a direct route or a transfer path. Constraints (1.6) and (1.7) define the binary decision variables used in the model.

This type of formulation often requires that the set of an exponential number of candidate routes be enumerated as input. The formulation is also nonlinear, as evidenced by the objective function (1.1) and constraint (1.2), which both contain the product of numerous \( x_r \). This makes the problem quite difficult to solve. Exact solutions to this type of formulation have therefore been obtained only for very small networks. Greedy and/or meta-heuristic algorithms are typically used to solve this type of problem on larger scales (Ceder, 2007). We thus make no attempt to solve this type of formulation to exact optimality in the present study.

### 1.2. Network Design Model

Transit network design can also be considered as a special case of transportation network design for multiple commodities (e.g., see comprehensive reviews in Friesz, 1985; Yang and Bell, 1998; Magnanti, 1984). Model formulations along this line often assume three sets of input data: routes, \( R \); bus stops, \( N \); and directional transit route arcs, \( A \). Let \( D_{o,d} \) be the passenger demand flow from origin \( o \in N \) to destination \( d \in N \), and let \( d_{i,j} \) and \( t_{i,j} \) denote the distance and travel time from bus stop \( i \in N \) to \( j \in N \) respectively. We assume a constant bus cruising speed, \( v \).

The decision variables include a set of binary variables \( x = \{x_{i,j}^r : i, j \in N, r \in R\} \), \( p_i^r \), \( q_j^r \), \( i, j \in N, r \in R \) and continuous variables \( f = \{f_{i,j,r}^{o,d} : i, j, o, d \in N, r \in R\} \). Let \( x_{i,j}^r = 1 \) if on the route \( r \in R \) a bus travels from the head of the arc \( i \in N \) to the tail of the arc \( j \in N \); and 0 otherwise. Binary variables \( p_i^r \) and \( q_j^r \) specify whether node \( i \in N \) is the origin and destination of route \( r \in R \), respectively. Let \( f_{i,j,r}^{o,d} \) be the passenger flow from origin \( o \in N \) to destination \( d \in N \) that travels from stop \( i \in N \) to stop \( j \in N \) along route \( r \in R \). Equations (2.1)-(2.9) present the mathematical formulation.

(Model II) \[
\text{min } c_d \sum_{re R} \sum_{j \in N} \sum_{i \in N} d_{i,j} x_{i,j}^r + c_t \sum_{o \in N} \sum_{d \in N} \sum_{re R} \sum_{i \in N} \sum_{j \in N} t_{i,j} f_{i,j,r}^{o,d} \]  
(2.1)

s.t. \[
\sum_{r} \sum_{j} x_{i,j}^r \geq 1, \ \forall i \]  
(2.2)

\[
\sum_{j} x_{i,j}^r - \sum_{j} x_{j,i}^r = p_i^r - q_j^r, \ \forall i \in N, r \in R \]  
(2.3)

\[
\sum_{i \in N} p_i^r = 1, \ \forall r \in R \]  
(2.4)

\[
\sum_{j \in N} q_j^r = 1, \ \forall r \in R \]  
(2.5)
\[
\sum_{i \in N} x_{i,j}^r \leq 1, \forall i \in N, r \in R \quad (2.6)
\]

\[
\sum_{i \in N} x_{i,j}^r \leq 1, \forall j \in N, r \in R \quad (2.7)
\]

\[
\sum_r \sum_j f_{i,j,r}^{o,d} - \sum_r \sum_j f_{i,j,r}^{a,d} = \begin{cases} -D_{od}, & i = o \\ 0, & \text{otherwise, } \forall o,d \\ D_{od}, & i = d \end{cases} \quad (2.8)
\]

\[
f_{i,j,r}^{o,d} \leq M \chi_{i,j}^r, \forall i, j, r, o, d \quad (2.9)
\]

Objective function (2.1) minimizes the total cost related to the total bus distance traveled,
\[
\sum_{r \in R} \sum_{j \in N} x_{i,j}^r, \text{ with cost coefficient } c_d, \text{ and the total passenger time, } \sum_{r \in R} \sum_{i \in N} \sum_{j \in N} t_{i,j}^r f_{i,j,r}^{o,d}, \text{ with cost coefficient } c_t. \]

Constraint (2.2) guarantees that each bus stop must be visited by at least one route. Constraint (2.3) guarantees that the number of directional arcs into a stop is the same as the number out of the stop, save for terminus stops along a route. Constraints (2.4) and (2.5) ensure that the origin and destination nodes for each route are unique. Constraints (2.6) and (2.7) enforce that a bus stops no more than once at each node. Constraint (2.8) ensures that the passenger flow into a bus stop equals the flow out of that bus stop, unless the stop is the origin or the destination for some of the passengers. Constraint (2.9) guarantees that if there are passengers going on a bus route arc between two bus stops, then that arc must exist. Under these constraints, \( M \) is a large number that is set to be the hourly passenger-carrying capacity of the route.

1.3. Wan and Lo’s Model

The previous model assumes that buses operate with a headway that is equal across all routes. Wan and Lo (2003) proposed instead an MIP formulation for the design of multi-route transit networks that allows for route-specific headways. That model considers a network graph \((N, A)\), where \( N \) is the node set and \( A \) is the arc set. It further takes \( W^O \) and \( W^D \) to represent the origin and destination node sets, and \( W \) to be the set of OD pairs. Set \( R \) represents the set of transit routes to be determined. Binary variable \( d_{s,t}^r = 1 \) if transit route \( r \in R \) connects node \( s \in N \) and node \( t \in N \); or 0 otherwise. Continuous variable \( q_{s,t}^r, r \in R, (s,t) \in W \) defines the passenger flow on transit route \( r \) from origin \( s \) to destination \( t \).

Continuous variable \( f_r \) defines the frequency of transit route \( r \in R \). Integer variable \( \psi_{r,k}^s, r \in R, k \in N \), defines the sequence of node \( k \) along transit route \( r \). Binary variables \( \xi_{r,k}^s \) and \( \phi_{r,k}^t \), \( r \in R, k \in N \), respectively equal 1 if node \( k \) is the starting or ending node of route \( r \); or 0 otherwise. The model formulation is as follows:

(Model III) \[
\begin{align*}
\text{min} & \quad \sum_{r \in R} \sum_{a \in A} c z_a^r \\
\text{s.t.} & \quad \sum_{r \in R} q_{s,t}^r = q_{s,t}, \forall (s,t) \in W, \quad (3.1) \\
\quad & \quad C f_r - \sum_{(s,t) \in W} d_{s,u}^{s,r} = \sum_{(u,v) \in W} d_{u,v}^{s,r}, \forall u \in W^O \quad (3.3) \\
\quad & \quad C f_r - \sum_{(s,t) \in W} d_{r,u,s}^{r,s} = \sum_{(u,v) \in W} d_{u,v}^{r,s}, \forall u \in W^O \quad (3.4) \\
\quad & \quad f_{\min} \leq f_r \leq f_{\max}, \forall r \in R \quad (3.5) \\
\quad & \quad 0 \leq q_{s,t}^r \leq M \left( d_{s,t}^r + d_{t,s}^r \right), \forall (s,t) \in W, r \in R \quad (3.6)
\end{align*}
\]
\[ \xi_k^r + \sum_{he \in N^-} x_{hh}^r \leq \psi_k^r \leq \xi_k^r + M \sum_{he \in N^-} x_{hh}^r, \forall k \in N, r \in R \] (3.7)

\[ \psi_k^r \leq M \left( \sum_{he \in N^-} x_{hh}^r + \sum_{he \in N^-} x_{hh}^r \right), \forall k \in N, r \in R \] (3.8)

\[ x_{hh}^r + \psi_h^r - M (1 - x_{hh}^r) \leq \psi_k^r \leq x_{hh}^r + \psi_h^r + \xi_k^r - M (1 - x_{hh}^r - \xi_k^r), \] (3.9)

\[ \forall k \in N, h \in N^-_k, r \in R \]

\[ \frac{1}{M} \sum_{he \in A} x_a^r \leq \frac{1}{M} \sum_{he \in A} x_a^r \leq 1 - \frac{1}{M} \left( 1 - \sum_{he \in A} x_a^r \right), r \in R \] (3.10)

\[ \sum_{he \in N} \varphi_e^r \leq \sum_{he \in N} \varphi_e^r, r \in R \] (3.11)

\[ \sum_{he \in N} x_{hh}^r \leq 1, k \in N, r \in R \] (3.12)

\[ \sum_{he \in N_k^+} x_{hh}^r \leq 1, k \in N, r \in R \] (3.13)

\[ \sum_{he \in N} x_{hh}^r + \xi_k^r - \varphi_e^r = \sum_{he \in N} x_{hh}^r, \forall k \in N, r \in R \] (3.14)

\[ \frac{\psi_k^r}{M} \leq \overline{\psi}_k \leq \frac{M - 1}{M} + \frac{\psi_k^r}{M}, \forall k \in N, r \in R \] (3.15)

\[ \frac{1}{M} (-\psi_i^r + \psi_j^r) + M (\psi_i^r - 1) \leq d^r_{ij} \leq \overline{\psi}_i^r + \frac{1}{M} (-\psi_i^r + \psi_j^r), \] (3.16)

\[ \forall i, j \in N, i \neq j, r \in R \]

\[ z_a^r - f_r \leq 0, a \in A, r \in R \] (3.17)

\[ z_a^r - M x^r_a \leq 0, a \in A, r \in R \] (3.18)

\[ M \left( x^r_a - 1 \right) - z^r_a + f_r \leq 0, a \in A, r \in R \] (3.19)

\[ d^r_{su} + d^r_{ut} - 1 \leq d^a_{sut}, s, u, t \in N, r \in R \] (3.20)

\[ d^a_{sut} \leq d^r_{su}, s, u, t \in N, r \in R \] (3.21)

\[ d^a_{sut} \leq d^r_{ut}, s, u, t \in N, r \in R \] (3.22)

\[ d^a_{su} + d^r_{ut} - 1 \leq d^a_{sut}, s, u, t \in N, r \in R \] (3.23)

\[ d^a_{su} - q^r_a \leq 0, a \in A, r \in R \] (3.24)

\[ d^a_{su} - Md^r_a \leq 0, a \in A, r \in R \] (3.25)

\[ M \left( d^r_a - 1 \right) - d^a_{su} + q^r_a \leq 0, a \in A, r \in R \] (3.26)

\[ d^a_{su}, q^r_a, z^r_a \leq 0, s, u, t \in N, r \in R \] (3.27)

\[ d^a_{su}, q^r_a, z^r_a \geq 0, s, u, t \in N, r \in R \] (3.28)

\[ d^a_{su} \leq 0, s, u, t \in N, r \in R \] (3.29)

\[ d^a_{su} \geq 0, s, u, t \in N, r \in R \] (3.30)
The objective function (3.1) minimizes the total operating cost for all of the transit routes combined. Constraint (3.2) guarantees that the summation of demand on all routes for each origin, \( s \), and destination, \( t \), satisfies the total demand for each OD pair, \( q_{st} \). Constraints (3.3) and (3.4) ensure the passenger flow at each origin node \( u \) along each transit route \( r \) does not exceed the passenger-carrying capacity. Constraint (3.5) sets a lower bound, \( f_{\text{min}} \), and upper bound, \( f_{\text{max}} \), for service frequency on each route. Constraint (3.6) ensures that the routes must exist so as to have flow, \( q_{s,t}^{f} \), on them. Constraint (3.7) sets an upper bound on the sequence labels, \( \psi_{k}^{r} \), and (3.8) assures that the sequence labels, \( \psi_{k}^{r} \), are on the routes. Constraint (3.9) determines the sequence, \( \psi_{k}^{r} \), along each route \( r \). Constraints (3.10) and (3.11) guarantee that the numbers of starting nodes, \( \xi_{k}^{r} \), and ending nodes, \( \phi_{k}^{r} \), are equal for each route \( r \). Constraints (3.12) and (3.13) eliminate cyclic routes (i.e., those passing a same station more than once). Constraint (3.14) ensures the validity of the node labels, \( \psi_{k}^{r} \). Constraints (3.15) and (3.16) define the dummy variable \( \overline{\psi}_{k}^{r} \) which is equal to 1 if node \( k \) is visited by transit route \( r \) or 0 otherwise. Constraints (3.17)-(3.19) define variable \( z_{a} \) to be equal to \( x_{a}^{r}f_{r} \) so as to make the objective function linear. Constraints (3.20)-(3.23) define variables \( d_{su}^{n} \) to be the product of \( d_{su}^{r} \) and \( d_{su}^{a} \). Similarly, Constraints (3.24)-(3.26) and (3.27)-(3.29) define variables \( d_{su}^{mr} \) and \( d_{su}^{mr} \), respectively. Binary variables \( d_{su}^{n} \), \( d_{su}^{r} \) and \( d_{su}^{mr} \) are introduced so as to linearize the relevant constraints. Constraints (3.30)-(3.32), (3.33) and (3.34)-(3.36) define continuous, integer and binary variables, respectively.

2. Numerical Experiments

This section explores the computational performance of Models II and III for a square city of 100 km². The city is discretized into a general \( m \)-by-\( n \) grid network as illustrated in Figure 1. Each node represents a candidate location for a transit stop (i.e., \( |N| = mn \)), and links indicate candidate travel directions for transit routes. Both Models II and III are coded through the AMPL interface and solved by the CPLEX solver on a 3.1 GHz CPU desktop with 16 GB RAM.

The demand for transit travel is assumed to be continuously- and heterogeneously-distributed over the city, with highest demand in the central region, and lower demand in the periphery. (The demand topography is like that of Case I in Ouyang, et al, 2014.) The demand density [per unit time-area\(^2\)] from the origin, \( o \), at coordinates \( (x_{1},y_{1}) \) to destination, \( d \), at \( (x_{2},y_{2}) \) is as follows:

\[
\delta(x_{1},y_{1},x_{2},y_{2}) = \prod_{i=1}^{2} \left( 0.0016 + 0.065 \exp \left[ - \left( \frac{1}{2} x_{i} - \frac{5}{2} \right)^{2} - \left( \frac{1}{2} y_{i} - \frac{5}{2} \right)^{2} \right] \right). \tag{3}
\]
The parameters in the demand function are set so that the city’s total rate of trip-making by transit is (10,000/hr). The demand is aggregated around the nearest node to obtain the origin/destination demand, $D_{o,d}, o \in N, d \in N$, as model input. The distance matrix between each pair of nodes can be computed along the network. For example, for 4-by-4 and 5-by-5 cases the distance between two neighboring nodes is 2.5 km and 2.0 km, respectively. To compute travel time between origin and destination node pairs, the transit vehicle’s cruising speed is assumed to be $v=25$km/hr.

For Model II, the agency- and user-cost parameters in the objective function are set to be $c_d=10$ and $c_t=1$. In Model III, the only cost coefficient, $c_i$ is set to be 1. For both models, $M$ is set at 500.

For illustration, Figure 2 shows the optimal network design generated from Model II for the case of 16 stops and 4 transit lines.
Multiple additional cases were considered. For each case, we specified the numbers of: \( m, n, OD \) pairs and routes; and sought optimal solutions (for route configurations, stop locations and transit vehicle headways) using Models II and III.\(^1\) In all but one case, the maximum computation time was set to be 200 mins (12,000 s), because CPU times larger than this may severely restrict an analyst’s ability to examine alternative scenarios via numerical experiment. (The one case in which CPU time was allowed to exceed 200 mins will be discussed momentarily.)

Numerical results for our initial cases are presented in Table 1. Examination of that table reveals that the number of \( m \) and of \( n \) never exceeds 5. Thus, the networks are all quite small in their physical sizes; e.g. much smaller than the networks studied in Ouyang, et al. (2014) and much smaller than those typically encountered in real settings. Yet in many of the test cases the models failed to generate optimal solutions (even to within tolerances of 10\%) within the maximum CPU time of 200 mins: asterisks in Table 1 denote instances of this failure.

For one case, denoted case 7 in Table 1, CPU time was allowed to exceed 200 mins in the application of Model II. Note for that case that \( n = m = \) number of routes = 5, and that Model II produced an optimal solution -- to within a tolerance of 9\% -- only after nearly 19,000 s (more than 5 hrs) of CPU time.

Of yet greater concern, Model III was never able to produce optimal solutions within 200 mins of CPU time.\(^2\) In light of this, we next considered small networks in which \( m = n \) =3; the number of routes was never more than 3; and the number of OD pairs with non-zero demand was set to be extremely small (either 8 or 16). Outcomes are shown in Table 2.

### Table 1. Numerical results for cases with complete OD pairs

<table>
<thead>
<tr>
<th>Case</th>
<th>( m )</th>
<th>( n )</th>
<th># of OD pairs</th>
<th>Model II</th>
<th>Model III (Wan and Lo, 2003)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td># of routes</td>
<td># of binary var.</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>3</td>
<td>72</td>
<td>11</td>
<td>506</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
<td>240</td>
<td>8</td>
<td>640</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>5</td>
<td>380</td>
<td>5</td>
<td>510</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>5</td>
<td>600</td>
<td>2</td>
<td>260</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>600</td>
<td>3</td>
<td>390</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>5</td>
<td>600</td>
<td>4</td>
<td>520</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>5</td>
<td>600</td>
<td>5</td>
<td>650</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>5</td>
<td>600</td>
<td>6</td>
<td>780</td>
</tr>
</tbody>
</table>

* : No results after 200 minutes CPU time for a tolerance of 10\% optimality gap.

### Table 2. Numerical results of Model III for cases with partial OD pairs

<table>
<thead>
<tr>
<th>Case</th>
<th>( m )</th>
<th>( n )</th>
<th># of OD pairs</th>
<th># of binary var.</th>
<th># of integer var.</th>
<th># of constraints</th>
<th>Opt. gap</th>
<th>CPU time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>3</td>
<td>8</td>
<td>2</td>
<td>376</td>
<td>18</td>
<td>10,268</td>
<td>0%</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
<td>8</td>
<td>3</td>
<td>376</td>
<td>18</td>
<td>15,398</td>
<td>*</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>16</td>
<td>2</td>
<td>376</td>
<td>18</td>
<td>10,300</td>
<td>9%</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>3</td>
<td>16</td>
<td>3</td>
<td>564</td>
<td>27</td>
<td>15,442</td>
<td>*</td>
</tr>
</tbody>
</table>

* : No results after 200 minutes CPU time for a tolerance of 10\% optimality gap.

---

1 Recall that optimal solutions were not sought using Model I because it typically requires the choice of some heuristic.

2 Tellingly perhaps, the numerical example in Wan and Lo (2003) consisted of a network of only: 10 nodes, 16 links, 9 OD pairs and 3 routes.
That table reveals that for the downsized example cases, Model III could sometimes generate optimal or near-optimal solutions within 200 mins of CPU time. Yet this only occurred when the number of routes on the network was kept to just 2.

3. Conclusion
The computational performance of two state-of-the-art mixed-integer programming formulations for the transit network design problem was studied. A standard commercial solver was used for this purpose, and only smaller-sized networks were considered. The maximum CPU time for each solution was typically set to 200 mins (3.3 hrs).

The results unveil the problems that can arise due to the computational burdens of the select mixed-integer (i.e. discrete) models. Optimal or near-optimal solutions could be obtained for only the very smallest of networks; i.e. for those comparable in size to Mandl’s network, for example. Those networks are much smaller than the ones studied in Ouyang, et al (2014) and smaller than those often encountered in real-world settings.

The findings indicate that the application of discrete modeling methods to real-sized networks will often require heuristic techniques of one type or another. Promising alternatives would seem to be approximation techniques of the kind used in Ouyang, et al (2014).

Acknowledgment
The authors thank Ryan Smith, University of Illinois at Urbana-Champaign, for his help with mathematical modeling and AMPL coding.

References


