RC22808 (W0305-130) May 28, 2003
Mathematics

IBM Research Report

Designing Private Line Networks

Oktay Gunluk, Beate Brockmuller[*], Lawrence Wolsey[+]

IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

[*]BASF, Germany

[+]CORE
Universite` de Louvain
Belgium

Designing Private Line Networks

B. Brockmüller[*], O. Günlük[†] and L.A. Wolsey[‡]

May 27, 2003

Abstract

We study a capacitated network design problem arising in the design of private line networks. Given a complete graph, a subset of its node set (the "hub" set), and point-to-point traffic demands, the objective is to install capacity on the edges (using several batch sizes and nonlinear costs), and route traffic in the resulting capacitated network, so that 1) all the demand between a pair of nodes is routed along a single path, and 2) the demand is either sent directly from source to sink, or via a number of hub nodes. We first formulate an initial integer program, and various approximations to it. Valid inequalities are then derived for a special knapsack problem involving both integer and 0-1 variables arising from the capacity constraints on an edge. These and related inequalities are then used to strengthen the formulations. Computational results using these inequalities within a general purpose branch-and-cut system are presented.

[*]BASF, Germany.

[†]IBM Research, U.S.A. Research partially supported by a post-doctoral fellowship at CORE and by NSF grant DMS-9527124 to Cornell University.

[‡]CORE, Université Catholique de Louvain, Belgium.

1 Introduction

The problem studied in this paper is a minimum cost capacitated multi-commodity flow problem. Given a set of known demands between certain pairs of nodes, the problem is to install sufficient "bandwidth" capacity on each of the edges such that each demand can be routed on a single path from source to sink. The capacity on each edge is installed in discrete amounts and the associated costs are nonlinear. The total edge capacity installation costs are to be minimized.

Applications of this problem (or its variants) arise in the telecommunications industry for both service providers (i.e. long-distance carriers) and their customers. Our motivation for this study comes from the customer side where the nodes of the network correspond to the offices and production facilities of a big corporation with nationwide operations. Usually a corporation leases private lines between pairs of its sites from a telecommunications carrier in order to satisfy its data transmission requirements. A private line is a physical analog or digital line, which is permanently available and used exclusively by a single customer of the telecommunications carrier.

The customer has to pay monthly recurring charges depending on the terms of the contract with the telecommunications carrier, on the bandwidth of the line, and on the distance covered by the line. A bandwidth refers to the information-carrying capacity of the line. It is generally specified in kilobits per second (Kbps). Bandwidth is provided in certain discrete values, usually in integral multiples of 64Kbps (normal telephone lines) and of 1,544 Kbps (a batch of 24 lines carrying 64 Kbps each, called T1 lines). The associated costs are nonlinear in the number of leased 64Kbps lines, while the costs are linear in the number of 1,544Kbps lines. There is also a break-even cost, i.e. it is cheaper to lease a 1,544Kbps line if the bandwidth exceeds a certain number of single 64Kbps lines. The dependence of the tariffs on the distance is given by distance rate bands. The distance is usually determined by the mileage computed from the geographical locations of the customer sites.

Although it is difficult to obtain accurate input data on transmission requirements for this problem, the telecommunications staff of the corporation has sufficient information on which fixed logical site-to-site connections and bandwidths are needed. The objective is to (re-)design the structure of the corporation's private line network so that the total private line costs are minimized and that the given demands on data connections are routed via single paths through the network. Only a predefined subset of sites - the so called hub sites - can be used for routing purposes. These hub sites are provided with the appropriate technical routing equipment and trained personnel.

The results of the optimization serve as a decision support for the management whenever new contracts have to be negotiated with the telecommunications carriers, new sites are established or existing sites are shut down.

A second objective related to this study was the development of general purpose mixed integer programming software capable of solving such problems, see PAMIPS [15]. This influenced the choice of a multicommodity flow formulation by the user, and led us to develop an algorithmic approach adapted for use with a general purpose solver.

Many related network design problems have been studied recently in which the capacity installation options differ slightly. Barahona [3], Bienstock and Gunluk [4], Bienstock et al.[5] and Magnanti, Mirchandani and Vachani[12] among others treat problems in which demands can be split among several paths. A variety of inequalities and separation algorithms have been developed both for families of valid inequalities of a global nature such as cut and strengthened cut inequalities [3],[4],[5],[11],[12], partition inequalities [4],[5],[12] and flow cut-set inequalities [4], and local edge cuts such as the residual capacity inequalities [12].

Problems involving sending flow on a single path have been treated by Balakrishnan, Magnanti and Wong [2] for tree graphs, Gavish and Altinkemer [10] for the design of a backbone network with a nonlinear objective function, and Dahl, Martin, and Stoer [9] for a problem similar to that treated here except for additional "pipe network" constraints, and a simpler cost structure.

We now describe the contents of the paper. In Section 2 we describe the problem, and give an initial multi-commodity integer programming formulation of the problem. This contrasts with the cut formulation used in [9]. The nonlinear nature of the cost structure on each edge forces one to use a considerable number of 0-1 variables to obtain a linear integer programming formulation. This leads us to develop two simplified models, one a relaxation providing a lower bound on the optimal value of the original problem, and the other a restriction providing an upper bound.

In Section 3 we study the polyhedral structure of an edge capacity submodel obtained by just considering the traffic flow and bandwidth installation on a single edge. This leads to a knapsack constraint in which $0-1$ variables appear on one side representing the demands that pass through that edge, and on the other side there is both a set of $0-1$ variables linked by a generalized upper bound constraint and an integer variable representing the choices for the installation of bandwidth capacity. Valid inequalities are derived. In Section 4 we investigate how cut-set inequalities can be adapted for our specific problem. As shown in [3] and [4], cut inequalities are typically the most important inequalities as they contain global, as opposed to local (single edge or single node) information. However finding out on which cuts to generate these inequalities is typically a difficult problem requiring the use of separation heuristics embedded in a cutting plane procedure and the development of a special purpose system.

In Sections 5 and 6 we discuss computational results using a prototype general purpose branch-and-cut mixed integer programming solver $bc-opt$ based on XPRESS. The desire to use such a system means that the main options are to strengthen the problem formulation a priori and define the problem-specific cuts to be used a priori so as to avoid having to develop special purpose cutting plane routines. Three approaches are examined all making use of some of the inequalities developed in Section 3. The first is to use the approximate problems to find hopefully good feasible solutions and tight bounds. The second is an exact solution approach, and the third is a cut-and-fix heuristic approach in which we use the LP values of certain variables after the addition of cuts to fix or restrict the branch-and-cut search tree.

2 A Formulation

The input consists of a node set $V = \{1, \ldots, n\}$, a nonempty subset $H \subseteq V$ of hub nodes, a demand graph $G = (V, Q)$, a demand vector $d \in R_+^{|Q|}$, and the capacity installation costs described below.

The quantity d_q associated with demand $q \in Q$ must flow on a single directed path from i^q to j^q where $i^q, j^q \in V$ with $i^q < j^q$. The choice of paths is restricted either to the single arc (i^q, j^q), or to a path of distinct nodes $i^q, v_1, \ldots, v_r, j^q$ where $r \geq 1$ and $v_1, v_2, \ldots, v_r \in H$. Thus all intermediate nodes on a path must be hub nodes. Therefore, although all arcs of the complete digraph on V can potentially carry flow, only a subset of the arcs are candidates for a specific instance. Specifically let A^q be the set of all arcs lying on at least one feasible path from i^q to j^q, $A = \cup_{q \in Q} A^q$ and $D = (V, A)$ be the resulting digraph associated with the demand graph G. Note that provided that the demand graph contains at least two distinct sources and two distinct sinks, A contains all arcs (i, j) with $i, j \in H$, A contains an arc (i, j) with $i, j \in V \setminus H$ if and only if $i < j$ and $(i, j) = (i^q, j^q)$ for some $q \in Q$, A contains an arc (i, j) with $i \in H, j \in V \setminus H$ if and only if there exists some $q \in Q$ with $j^q = j$, and similarly an arc (i, j) with $i \in V \setminus H, j \in H$ if and only if there exists some $q \in Q$ with $i^q = i$.

Cost data is provided for each edge of the complete graph on V. For each edge e, the costs $\{c_{e,k}\}_{k=1}^{m-1}$ of installing k individual units of capacity are given with $0 < c_{e,1} < \ldots < c_{e,m-1}$, and also the cost f_e per multiple of $m = 24$ units. Again the data suggests some immediate preprocessing. If $E^q = \{e = (i, j) : (i, j) \in A^q$ or $(j, i) \in A^q\}$, and $E = \cup_{q \in Q} E^q$, then only edges in E are candidates for the installation of capacity. In addition if $\kappa_e = \min[\min\{k : c_{e,k+1} \geq f_e\}, m - 1]$, one only needs to consider installing between 1 and κ_e single units, and/or multiples of m units on edge $e \in E$.

The following variables are used to formulate the problem:

$w_{ij}^q = 1$ if the path chosen for demand $q \in Q$ contains arc $(i, j) \in A^q$

$z_{e,k} = 1$ if k single units are installed on edge $e \in E$

x_e is the number of single units installed on edge $e \in E$,

y_e is the number of batches of m units installed on edge $e \in E$.

Letting N^q be the node-arc incidence matrix associated with A^q, the problem can now be formulated as (P):

$$\min: \quad \sum_{e\in E}(\sum_{k=1}^{\kappa_e} c_{e,k}z_{e,k} + f_e y_e) \tag{1}$$

subject to :

$$N^q w^q = \delta^q \text{ for } q \in Q \tag{2}$$

$$\sum_{q:(u,v)\in A^q} d_q w^q_{uv} + \sum_{q:(v,u)\in A^q} d_q w^q_{vu} \le x_e + m y_e \quad \text{for } e=(u,v)\in E \tag{3}$$

$$\sum_{k=1}^{\kappa_e} z_{e,k} \le 1 \quad \text{for } e \in E \tag{4}$$

$$x_e = \sum_{k=1}^{\kappa_e} k z_{e,k} \quad \text{for } e \in E \tag{5}$$

$$z_{e,k} \in \{0,1\} \text{ for } k=1,\ldots,\kappa_e, e\in E \tag{6}$$

$$w^q_e \in \{0,1\} \text{ for } e \in E^q, q \in Q \tag{7}$$

$$x_e, y_e \in Z_+ \text{ for } e \in E \tag{8}$$

where for all $q \in Q$, $\delta^q \in \{0,1\}^{|V|}$ with $\delta^q_v = 0$ for $v \notin \{i^q, j^q\}, \delta^q_{i^q} = 1$ and $\delta^q_{j^q} = -1$. Here constraints (2) are flow conservation constraints, (3) is the capacity constraint on edge e, while (4) and (5) describe how the small capacity x_e is constructed. Note that because all the installation costs are positive, there always exists an optimal solution in which each demand selects a route not containing a cycle, and therefore chooses a single path as required.

2.1 Edge Sets

So as to study certain edge sets, and then to obtain two simplified problems requiring less variables, we now introduce some additional variables and notation.

Note that for $e = (u,v) \in E^q$, either variable w^q_{uv} or variable w^q_{vu} or both exist. We introduce the new variable $\omega^q_e = w^q_{uv} + w^q_{vu}$. The variable ω^q_e is a $\{0,1\}$ variable because no directed path passes more than once through an edge and the solutions are cycle-free. Also let $Q(e) = \{q \in Q : e \in E^q\}$.

Now we define the two sets that we study in the next section: the *simple edge* *knapsack set*

$$Y_e = \Big\{(\omega_e, x_e, y_e) \in B^{|Q(e)|} \times Z^1_+ \times Z^1_+ : \sum_{q \in Q(e)} d_q \omega^q_e \le x_e + m y_e\Big\}$$

and the *mixed edge knapsack set*

$$X_e = \Big\{(\omega_e, x_e, y_e, z_e) \in B^{|Q(e)|} \times Z^1_+ \times Z^1_+ \times B^{\kappa_e} : (\omega_e, x_e, y_e) \in Y_e, (x_e, z_e) \text{ satisfies (4),(5)}\Big\}$$

The constraint set of the original problem can now be written as $U =$

$$\begin{aligned} N^q w^q &= \delta^q \quad \text{for } q \in Q \\ \omega^q_e &= w^q_{uv} + w^q_{vu} \quad \text{for } e = (u,v) \in E^q, q \in Q \\ (\omega_e, x_e, y_e, z_e) &\in X_e \quad \text{for } e \in E \\ w^q_{uv} &\in \{0,1\} \quad \text{for } (u,v) \in A^q, q \in Q \end{aligned}$$

2.2 Equivalent Formulations

Here we derive two formulations based on U involving less variables, obtained by projecting out some or all of the binary capacity variables $z_{e,k}$. In one we linearize the cost function for the unit capacities, and in the second we partially linearize it. First note that

$$\Big\{x_e : (x_e, z_e) \in Z^1_+ \times B^{\kappa_e} \text{ satisfies (4) and (5) }\Big\} = \Big\{x_e \in Z^1_+ : x_e \le \kappa_e\Big\},$$

and thus

$$\Big\{(\omega_e, x_e, y_e) : (\omega_e, x_e, y_e, z_e) \in X_e\Big\} = \Big\{(\omega_e, x_e, y_e) : (\omega_e, x_e, y_e) \in Y_e, x_e \le \kappa_e\Big\}.$$

It follows that the feasible region $\tilde{U}$:

$$\begin{aligned} N^q w^q &= \delta^q \quad \text{for } q \in Q \\ \omega^q_e &= w^q_{uv} + w^q_{vu} \quad \text{for } e = (u,v) \in E^q, q \in Q \\ (\omega_e, x_e, y_e) &\in Y_e \quad \text{for } e \in E \\ w^q_{uv} &\in \{0,1\} \quad \text{for } (u,v) \in A^q, q \in Q \\ x_e &\le \kappa_e \quad \text{for } e \in E. \end{aligned}$$

has the property that for any (x,y,w,ω) there exists z such that $(x,y,w,\omega,z) \in U$ if and only if $(x,y,w,\omega) \in \tilde{U}$.

In the second reformulation, the z variables are only partially eliminated. In particular we keep the exact value for one unit, and then linearize the costs for two units and upwards to give a better approximation. Specifically defining

$$u_e = \sum_{k=2}^{\kappa_e} k z_{e,k}, v_e = \sum_{k=2}^{\kappa_e} z_{e,k} \tag{9}$$

note that the set

$$\begin{aligned} W_e &= \{(x_e, z_{e,1}, u_e, v_e) : (x_e, z_e, u_e, v_e) \in Z^1_+ \times B^{\kappa_e} \times Z^1_+ \times B^1 \text{ satisfies (4),(5),(9) } \} \\ &= \{(x_e, z_{e,1}, u_e, v_e) \in Z^1_+ \times B^1 \times Z^1_+ \times B^1 : x_e = z_{e,1} + u_e, z_{e,1} + v_e \leq 1, \\ &\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad 2v_e \leq u_e \leq \kappa_e v_e\}. \end{aligned}$$

Now we obtain the feasible region $\bar{U}$:

$$\begin{aligned} N^q w^q &= \delta^q \quad \text{for } q \in Q \\ \omega^q_e &= w^q_{uv} + w^q_{vu} \quad \text{for } e = (u,v) \in E^q, q \in Q \\ (\omega_e, x_e, y_e) &\in Y^0_e \quad \text{for } e \in E \\ (x_e, z_{e,1}, u_e, v_e) &\in W_e \quad \text{for } e \in E \\ w^q_{uv} &\in \{0,1\} \quad \text{for } (u,v) \in A^q, q \in Q. \end{aligned}$$

with the property that for any (x,y,w,ω) there exists z such that $(x,y,w,\omega,z) \in U$ if and only if there exists $(z_{\cdot 1}, u, v)$ such that $(x,y,w,\omega,z_{\cdot 1},u,v) \in \bar{U}$.

2.3 Relaxations and Restrictions of (P)

Formulations $\tilde{U}$ and $\bar{U}$ can be used to obtain either relaxations or restrictions of (P). To obtain a relaxation, let $g_e = \min_{k=1,\ldots,\kappa_e} \frac{c_{e,k}}{k}$. Now if $(x_e, z_e) \in Z^1_+ \times B^{\kappa_e}$ satisfies (5),

$$\sum_{k=1}^{\kappa_e} c_{e,k} z_{e,k} \geq \sum_{k=1}^{\kappa_e} k g_e z_{e,k} = g_e x_e.$$

It follows that

$$(\underline{P}_0) \qquad \min\{\textstyle\sum_{e\in E} g_e x_e + \sum_{e \in E} f_e y_e : (x,y,w,\omega) \in \tilde{U}\}$$

is a relaxation of (P).

Now taking $h_e = \max_{k=2,\ldots,\kappa_e} \frac{c_{e,k}-c_{e,1}}{k-1}$, we see that for $(x_e, z_{e,1}, u_e, v_e) \in Z^1_+ \times B^1 \times Z^1_+ \times B^1$ satisfying (4),(5) and (9)

$$\begin{aligned}
\sum_{k=1}^{\kappa_e} c_{e,k} z_{e,k} &= c_{e,1} \sum_{k=1}^{\kappa_e} z_{e,k} + \sum_{k=2}^{\kappa_e} (c_{e,k} - c_{e,1}) z_{e,k} \\
&\leq c_{e,1}(z_{e,1} + v_e) + \sum_{k=2}^{\kappa_e} (k-1) h_e z_{e,k} \\
&= c_{e,1}(z_{e,1} + v_e) + h_e \sum_{k=2}^{\kappa_e} k z_{e,k} - h_e \sum_{k=2}^{\kappa_e} z_{e,k} \\
&= c_{e,1} z_{e,1} + (c_{e,1} - h_e) v_e + h_e u_e.
\end{aligned}$$

It follows that the problem $(\bar{P}^1)$

$$\min \Big\{ \sum_{e \in E} [c_{e,1} z_{e,1} + (c_{e,1} - h_e) v_e + h_e u_e] + \sum_{e \in E} f_e y_e \; : \; (x, y, w, z_{\cdot 1}, u, v) \in \bar{U} \Big\}$$

is a restriction of (P).

Note that the optimal values of $(\underline{P}_0)$ and $(\overline{P}^1)$ provide lower and upper bounds respectively on the optimal value of (P).

3 The Edge Capacity Submodel

Here we concentrate on developing valid inequalities for the following edge capacity sets

$$\begin{aligned}
Y &= \{(\omega, x, y) \in B^{|Q|} \times Z^1_+ \times Z^1_+ : \sum_{q \in Q} d_q \omega_q \leq x + my\}, \quad \text{and,} \\
X &= \{(\omega, x, y, z) \in B^{|Q|} \times Z^1_+ \times Z^1_+ \times B^\kappa : (\omega, x, y) \in Y, x = \sum_{k=1}^{\kappa} k z_k, \sum_{k=1}^{\kappa} z_k \leq 1\}
\end{aligned}$$

where the subscript e has been dropped and $\kappa \leq m$. We also abuse notation by using Q in place of $Q(e)$. Throughout this section, we suppose that $d_q \in Z^1_+$ for all $q \in Q$. In an earlier version of this paper [6], we consider the case when $d_q \in R^1_+$ and derive valid inequalities that are similar to the ones presented in this paper. Some of the inequalities presented in [6] are further studied in [1], see also [16].

3.1 Valid Inequalities

We introduce the notation $d_S = \sum_{q\in S} d_q, \eta_S = \lceil \frac{d_S}{m} \rceil, r_S = d_S - m(\eta_S - 1)$ and $c_S = \sum_{q\in S} \eta_q - \eta_S \geq 0$ for all $S \subseteq Q$. Also for simplicity we will write η_q for $\eta_{\{q\}}$, etc.

Proposition 1 *For all $S \subseteq Q$ with d_S not an integral multiple of m, the inequality*

$$\begin{aligned} x + r_S y + [r_S c_S - \sum_{q\in S}(r_S - r_q)^+] \quad \geq \quad & \sum_{q\in S}[r_S\eta_q - (r_S - r_q)^+]\omega_q \\ & + \sum_{q\in Q\setminus S} [r_S(\eta_q - 1) + (r_S + r_q - m)^+]\omega_q \end{aligned}$$

is valid for Y.

Proof. Letting $\bar{\omega}_q = 1 - \omega_q$ for $S \subseteq Q$, and dividing by m, all feasible points satisfy

$$\sum_{q\in S} \frac{-d_q}{m}\bar{\omega}_q + \sum_{q\in Q\setminus S} \frac{d_q}{m}\omega_q - y \leq \frac{-d_S}{m} + \frac{x}{m}$$

with all variables nonnegative and $\bar{\omega}_q, \omega_q$ and y integral.

Letting $f_S = \frac{-d_S}{m} - \lfloor \frac{-d_S}{m} \rfloor, f_q = -\frac{d_q}{m} - \lfloor -\frac{d_q}{m} \rfloor$ for $q \in S$ and $f_q = \frac{d_q}{m} - \lfloor \frac{d_q}{m} \rfloor$ for $q \in Q \setminus S$, the MIR inequality [14] is:

$$\begin{aligned} \sum_{q\in S} \Big[-\eta_q + \frac{(f_q - f_S)^+}{1 - f_S} \Big]\bar{\omega}_q \quad + \quad \sum_{q\in Q\setminus S} \Big[(\eta_q - 1) + \frac{(f_q - f_S)^+}{1 - f_S} \Big]\omega_q - y & \\ \leq \quad -\eta_S + \frac{x}{m(1 - f_S)} & \end{aligned}$$

It is readily checked that $r_S = m(1 - f_S), r_q = m(1 - f_q)$ for $q \in S$, and $r_q = mf_q$ for $q \in Q \setminus S$. Now after multiplying by r_S, the inequality can be rewritten as :

$$\begin{aligned} r_S \sum_{q\in S}[-\eta_q + (r_S - r_q)^+]\bar{\omega}_q \quad + \quad \sum_{q\in Q\setminus S} [r_S(\eta_q - 1) + (r_S + r_q - m)^+]\omega_q & \\ \leq \quad -r_S\eta_S + r_S y + x. & \end{aligned}$$

Finally after recomplementing the variables $\bar{\omega}_q$ for $q \in S$, we obtain the inequality.

Corollary. The following inequalities

$$x + r_S(y + c_S) \quad \geq \quad \sum_{q\in S} r_S\eta_q\omega_q + \sum_{q\in Q\setminus S} [r_S(\eta_q - 1) + (r_S + r_q - m)^+]\omega_q, \quad (10)$$

$$y + c_S + 1 \quad \geq \quad \sum_{q\in S} \eta_q\omega_q + \sum_{q\in Q\setminus S} (\eta_q - 1)\omega_q \quad (11)$$

are valid for Y.

Proof. To obtain inequality (10), it suffices to take a weakening of the MIR inequality in the proof of Proposition 1.

$$\sum_{q\in S} -\eta_q\bar{\omega}_q + \sum_{q\in Q\setminus S}[(\eta_q-1)+\frac{(f_q-f_S)^+}{1-f_S}]\omega_q - y \leq -\eta_S + \frac{x}{m(1-f_S)}$$

After multiplication by r_S and complementation this gives (10).

As $x \leq m$ for all feasible points, we have that $m + my \geq \sum_{q\in Q} d_q\omega_q$. Following exactly the same steps as in the proof above, leads to the MIR inequality

$$\sum_{q\in S}[-\eta_q + \frac{(f_q-f_S)^+}{1-f_S}]\bar{\omega}_q \quad + \quad \sum_{q\in Q\setminus S}[(\eta_q-1)+\frac{(f_q-f_S)^+}{1-f_S}]\omega_q - y$$
$$\leq -\eta_S + 1.$$

Weakening the inequality by dropping the fractional part of the coefficients gives

$$\sum_{q\in S} -\eta_q\bar{\omega}_q + \sum_{q\in Q\setminus S}(\eta_q-1)\omega_q - y \leq -\eta_S + 1$$

which after complementation gives (11). ∎

Now we derive valid inequalities for X.

Proposition 2 *For any $S \subseteq Q$ with d_S not an integral multiple of m, the inequalities*

$$\begin{aligned}\sum_{k<r_S} kz_k + r_S\sum_{k\geq r_S} z_k + r_S y + [r_S c_S - \sum_{q\in S}(r_S-r_q)^+] \quad &\geq \\ \sum_{q\in S}[r_S\eta_q - (r_S-r_q)^+]\omega_q + \sum_{q\in Q\setminus S}&[r_S(\eta_q-1)+(r_S+r_q-m)^+]\omega_q\end{aligned} \tag{12}$$

and

$$\sum_{k\geq r_S} z_k + y + c_S \geq \sum_{q\in S}\eta_q\omega_q + \sum_{q\in Q\setminus S}(\eta_q-1)\omega_q \tag{13}$$

are valid for X.

Proof. Consider a point $(\omega, x, y, z) \in X$. If $z_k = 0$ for all $k > r_S$, the lhs (left hand side) of the inequality (12) is $x + r_S y + [r_S c_S - \sum_{q \in S}(r_S - r_q)^+]$. From Proposition 1, this is greater than or equal to the rhs of the inequality.

If $z_k = 1$ for some $k > r_S$, lhs then equals $r_S + r_S y + [r_S c_S - \sum_{Q \in S}(r_S - r_q)^+]$. As $x \leq m$, the point $(w, y+1, 0)$ is also feasible, and so by Proposition 1, $r_S + r_S y + [r_S c_S - \sum_{Q \in S}(r_S - r_q)^+] \geq$ rhs. Thus inequality (12) is valid.

From a weakening of inequality (10) in the Corollary, we have that

$$\frac{x}{r_S} + y + c_S \geq \sum_{q \in S} \eta_q \omega_q + \sum_{q \in Q \setminus S} (\eta_q - 1)\omega_q$$

If $\sum_{k \geq r_S} z_k = 0$, then $x < r_S$ or $\frac{x}{r_S} < 1$, and therefore

$$y + c_S > \sum_{q \in S} \eta_q \omega_q + \sum_{q \in Q \setminus S} (\eta_q - 1)\omega_q - 1.$$

Therefore as all the terms in this expression are integral, $y + c_S \geq \sum_{q \in S} \eta_q \omega_q + \sum_{q \in Q \setminus S}(\eta_q - 1)\omega_q$ and inequality (13) is valid.

If $\sum_{k \geq r_S} z_k = 1$, then adding this equality to (11) of the Corollary, we see that the inequality is again valid. ∎

Example. Consider an instance with $d_1 = 22, d_2 = 34, d_3 = 15$ and $m = 10$. We have that $\eta_1 = 3, \eta_2 = 4, \eta_3 = 2, r_1 = 2, r_2 = 4, r_3 = 5$.

Taking $S = \{1, 2\}, \eta_S = 6, r_S = 6$ and $c_S = 1$, Proposition 1 tells us that the inequality

$$14\omega_1 + 22\omega_2 + 7\omega_3 \leq x + 6y$$

is valid for Y, and Proposition 2 that the inequalities

$$14\omega_1 + 22\omega_2 + 7\omega_3 \leq \sum_{k<6} kz_k + 6\sum_{k \geq 6} z_k + 6y$$

and

$$3\omega_1 + 4\omega_2 + \omega_3 \leq 1 + y + \sum_{k \geq 6} z_k$$

are valid for X.

With $S = \{1,2,3\}, \eta_S = 8, r_S = 1$ and $c_S = 1$. In this case all three inequalities obtained are identical

$$3\omega_1 + 4\omega_2 + 2\omega_3 \leq 1 + y + x.$$

■

4 Cut-set inequalities.

In this section, we discuss some extensions to the well-known cut-set inequalities. The cut-set inequalities have been successfully applied to similar problems (see [3], [4], [5], [11]) and [12]) .

For $S \subseteq V$ and $K \subseteq Q$ let $\delta(S) = \{\{i,j\} \in E : |\{i,j\} \cap S| = 1\}$, $Q(S) = \{q \in Q : |\{i_q, j_q\} \cap S| = 1\}$, $d(K) = \sum_{q \in K} d_q$ and $r(K) = d(K) - m(\lceil d(K)/m \rceil - 1)$. It is easy to see that the following simple MIR inequality

$$\sum_{e \in \delta(S)} (x_e + r(Q(S)) y_e) \geq r(Q(S)) \lceil d(Q(S))/m \rceil \tag{14}$$

associated with the cut-set inequality $\sum_{e \in \delta(S)} (x_e + m\ y_e) \geq d(Q(S))$ is valid for (P).

Using the special structure of (P), these inequalities can be modified as follows: Let $S \subset V$ and $\bar{K} \subset Q(S)$ be given. Define $\bar{\delta} = \delta(S) \cap (\cup_{q \in \bar{K}} E^q)$ to be the collection of edges in $\delta(S)$ that can carry at least one demand in $\bar{K}$. Furthermore, for $e \in \bar{\delta}$ let $\bar{m}_e = \min\{m, \sum_{q \in \bar{K} : e \in E_q} d_q\}$. Then, the following modified cut-set inequality:

$$\sum_{e \in \bar{\delta}} x_e + \sum_{e \in \bar{\delta} : m_e < r(\bar{K})} \bar{m}_e\ y_e + \sum_{e \in \bar{\delta} : m_e \geq r(\bar{K})} m\ y_e \geq d(\bar{K}). \tag{15}$$

and thus, the associated simple MIR inequality:

$$\sum_{e \in \bar{\delta}} x_e + \sum_{e \in \bar{\delta}} \min\left\{\bar{m}_e, r(\bar{K})\right\}\ y_e \geq r(\bar{K}) \lceil d(\bar{K})/m \rceil. \tag{16}$$

are valid for (P).

5 Computation

In this section, we describe the main components of our computational study: the preprocessing of the initial model, the inequalities added a priori and as "model" cuts, and the cuts generated in the branch and cut system.

In our computational study we have examined three problems: the original problem (P), the problem $(\underline{P}^0)$ that provides a lower bound on the optimal value of (P), and the problem $(\overline{P}^1)$ that provides an upper bound. The discussion below is essentially restricted to problem (P), but is similar for the other two models.

Our initial formulation for (P) consists of (1)-(8) except that the x_e variables are eliminated by substituting (5) in (3).

5.1 Initial Reformulation

The first modification just involves simple tightening of some bounds. Letting $L(e) = \sum_{q \in Q(e)} d_q$ denote the maximum possible flow on edge e, setting $h_e = \min\{\kappa_e, L(e)\}$ and $m_e = \min\{m, L(e)\}$, we can replace (3) by

$$\sum_{q \in Q(e)} d_q w_{uv}^q + \sum_{q \in Q(e)} d_q w_{vu}^q \leq \sum_{k=1}^{h_e} k z_{e,k} + m_e y_e$$

with $y_e \leq \lceil \frac{L(e)}{m} \rceil$. Specifically if $D(u) = \sum_{q \in Q: u \in \{i^q, j^q\}} d_q$ is the sum of all demands with source or sink in node u, then if $e = (u, v)$ with $u \notin H$ and $v \in H$, we have that $L(e) = D(u)$.

We also add an inequality for each node. Specifically for each node $i \in V$, we take the aggregate cut-set constraint

$$\sum_{e \in \delta(i)} (x_e + m y_e) \geq \sum_{q \in Q: i \in \{i^q, j^q\}} d_q$$

and add the corresponding MIR inequality

$$\sum_{e \in \delta(i)} (x_e + r y_e) \geq r\eta$$

where $\eta = \lceil \sum_{q \in Q: i \in \{i^q, j^q\}} d_q / m \rceil$ and $r = \sum_{q \in Q: i \in \{i^q, j^q\}} d_q - m(\eta - 1)$. These we call *simple cut-set inequalities.*

This completes the first a priori reformulation. Some further tightening is possible if one uses the fact that certain edges are only potentially used for a single demand.

5.2 A Priori (Model) Cuts

The next step involves the edge capacity inequalities developed in Proposition 2. For each singleton set $S = \{q\} \subseteq Q$, and for each edge $e \in E^q$, take a weakening

$$\sum_{k \geq r_q} z_{e,k} + y_e \geq \eta_q \omega_e^q$$

of the second inequality from Proposition 2. These inequalities are generated a priori in the model and classified as *model cuts*. For models $(\underline{P}_0)$ and $(\overline{P}^1)$, we use the corresponding weakening of inequality (11) of the Corollary to Proposition 1:

$$x_e + r_q y_e \geq r_q \eta_q \omega_e^q.$$

5.3 The Branch-and-Cut Algorithm

The approach using $bc - opt$ is now described:

- Take the initial formulation. Tighten and reformulate adding the simple cut-set and edge capacity inequalities.
- Run bc-opt, designating the edge capacity inequalities as model cuts.
- $bc - opt$ solves the resulting linear programming relaxation without the model cuts, and then adds violated model cuts as cutting planes.
- $bc - opt$ then generates violated integer knapsack constraints on the edge capacity constraints.
- In the tree, directives (priorities) are used with priority given to the y, z, ω variables in that order. Knapsack cuts are generated every 8 nodes in the tree. Non-binding cuts are deleted at the top node as well as in the tree. A "best bound among all" search strategy is used for the first 255 active nodes.

5.4 An Upper Bound Cut-and-Fix Heuristic

In an effort to obtain a good feasible solution quickly, we use a cut-and-fix heuristic, based on the observation that in the optimal solution to the tightened formulation many of the capacity variables are either integral, or are very close to integral. The idea is to apply branch-and-cut to the tightened formulation, but to restrict the search to solutions that are not very different from this fractional one. In particular if $\hat{p} = [\hat{z}, \hat{y}, \hat{w}]$ is the solution to the LP before branching, then we choose a small positive number ϵ and modify the bounds on the capacity variables as follows: $\lceil \hat{y}_e + \epsilon \rceil \geq y_e \geq \lfloor \hat{y}_e - \epsilon \rfloor$ for all $e \in E$. The resulting formulation, called the "restricted formulation", is solved to optimality by applying branch-and-cut. It is hoped that it will solve more quickly and that the resulting solutions give tight upper bounds for the original problem.

6 Computational Results

6.1 The Test Instances

Below we discuss our algorithmic approach and computational results obtained for the two approximate problems ($\underline{P}^0$) and ($\overline{P}^1$) and the real problem (P). The data set consists of four instances. All four instances are generated with the real cost and demand data. The demand data is integer and $m = 24$. The largest model corresponds to the real problem to be solved. The size of the four instances for the initial formulations of ($\underline{P}_0$) and (P) is indicated in Table 1, where *rows* is the number of constraints, *cols* the total number of variables, and $0-1$ the number of binary variables.

6.2 Branch-and-Cut for the Approximate Problems ($\underline{P}_0$) and ($\bar{P}^1$)

The results presented below have all been obtained using $bc - opt$ based on version 9.36 of the XPRESS-MP subroutine library (XOSL) running on a Pentium 166 MX PC.

In the Tables, InitLP gives the linear programming value for the original formulation, LP the value with bound tightening and single node cuts, XLP1 after also

Problem Instance	Total Nodes	Hub Nodes	All Demands	Problem $\underline{P}_0$			Problem P		
				rows	cols	$0-1$	rows	cols	$0-1$
net20	20	8	24	333	1504	1256	457	1939	1815
net28	28	8	32	469	2088	1712	657	2647	2459
net32	32	10	49	806	4923	4381	1077	5915	5644
net54	54	10	74	1303	7660	6676	1795	8954	8462

Table 1: Problem Instances

adding model cuts, and XPL2 after $bc - opt$ has generated knapsack cuts. Time and Nodes denote the total running time in secs, and the number of nodes in the branch-and-cut tree.

In Table 2 we present the results for the lower bounding problem ($\underline{P}_0$).

Instance	InitLP	LP	XLP1	XLP2	IP	Time	Nodes
net20	12096	13664	14926	15633	16,030	28	36
net28	15422	17293	18708	19357	19848	31	39
net32	20254	21419	23829	24696	25248	1330	168
net54	28149	28826	31522	32453	32840	2084	169

Table 2: Computational Results for LB($\underline{P}_0$)

In Table 3 we present the results for the upper bounding problem ($\overline{P}^1$) for the smaller instances with 20 and 28 nodes. Just from these instances we see that the running times for this reformulation are very large, and we decided that there was little to gain from this approach. However it was observed that the quality of the solutions obtained was very good.

6.3 Branch-and-Cut for the Real Problem (P)

The results are shown in Table 4. The difference in values between InitLP and IP suggests that these problems are difficult to solve without using cuts. This

Instance	InitLP	LP	XLP1	XLP2	IP	Time	Nodes
net20	12096	13688	15197	16106	17360	737	2706
net28	15422	17388	19040	19917	21229	2319	4153

Table 3: Computational Results for UB($\bar{P}^1$)

Instance	InitLP	LP	XLP1	XLP2	IP	Time	Nodes
net20	12096	13664	15445	16086	17263	320	712
net28	15422	17321	19271	19829	21107	542	936
net32	20254	21509	25102	25734	26689	4245	889
net54	28149	29256	32928	33498	34406	11887	1786

Table 4: Computational Results for (P)

is confirmed in that pure branch-and-bound takes 5865 seconds to solve net20 to optimality with directives running XPRESS version 10.04 on the same PC.

6.4 Results with the Cut-and-Fix Heuristic

We also tested the heuristic approach described in Section 5.4 based on the fractional solution $[\hat{z}, \hat{y}, \hat{w}]$ to the tightened formulation (with value XLP2). We used $\epsilon = 0$ and thus fixed the bounds of the capacity variables y to $\lceil \hat{y}_e \rceil \geq y_e \geq \lfloor \hat{y}_e \rfloor$ for all $e \in E$ before starting with the enumeration.

The results and running times are shown in Table 5. Note that the LP and XLP values have increased because the XPRESS preprocessor is able to profit from the tightened bounds. Observe that for each of the four instances, the heuristic finds the optimal solution. However the running time for the 54 node instance is significantly less than the time needed to prove optimality as shown in Table 4.

Instance	LP	XLP	IP	Time	Nodes
net20	13664	16414	17263	134	565
net28	17321	20226	21107	253	742
net32	21509	25895	26689	2103	987
net54	29256	33651	34406	5703	991

Table 5: Heuristic Results with *bc-opt*

7 Conclusion

Above we have presented results based on a combination of simple special purpose cuts, and general purpose integer knapsack cuts to solve and prove optimality for a set of fixed charge network design problems arising in practice. The results show that it is possible to solve such problems using a formulation with flow conservation constraints to represent the paths. The fact that the set of hub nodes is small may be one reason for the success of the approach based on local cuts that look only at individual nodes and edges. In many other applications, cut constraints of the type discussed in Section 4 have turned out to be the most critical cuts.

We have also shown that an appropriate "cut and fix" heuristic permits us to find very good solutions in a fraction of the time required to prove optimality. This means that, combined with the tight lower bounds, solutions guaranteed within a small percentage of optimality can be obtained in reasonable time.

The approximating problems tested give somewhat disappointing results in that the simple lower bounding problem gives weak lower bounds, and the quality of the resulting feasible solutions was not very good, whereas the upper bounding problem gave good solutions, but was very difficult to solve. This approach probably deserves further exploration.

More generally the valid inequalities developed in Section 3 show that our knowledge of the polyhedral structure of knapsack problems is still in it infancy. Also the fact that the cuts we use are dominated by Gomory mixed integer cuts suggests that developing a separation routine for Gomory mixed integer cuts working on rows of the initial formulation is a promising research topic.

Acknowledgement We are grateful to H. Marchand for his observation, that the derivation of the inequalities described in Section 3 could be based on Gomory cuts, which allowed us to significantly simplify the proofs from [6].

References

[1] A. Atamturk and D. Rajan, On Splittable and unsplittable flow capacitated network-design arc-set polyhedra, *Mathematical Programming* **92**, 315-333 (2002).

[2] A. Balakrishnan, T.L. Magnanti and R.T. Wong, A Decomposition Algorithm for Local Access Telecommunications Network Expansion Planning, *Operations Research* **43**, 58-76 (1995).

[3] F. Barahona, Network Design Using Cut Inequalities, *SIAM Journal on Optimization* **6**, 823-837 (1996).

[4] D. Bienstock, and O. Günlük, Capacitated Network Design - Polyhedral structure and Computation, *ORSA Journal on Computing* **8**, 243-259 (1996).

[5] D. Bienstock, S. Chopra, O. Günlük, and C-Y. Tsai, Minimum Cost Capacity Installation for Multicommodity Network Flows, *Math. Programming B* **81**, 177-199 (1998).

[6] B. Brockmüller, O. Günlük, and L.A. Wolsey, Designing Private Line Networks - Polyhedral Analysis and Computation, Core Discussion Paper 9647, Université Catholique de Louvain (1996).

[7] D. Bienstock, Private communication.

[8] S. Ceria, C. Cordier, H. Marchand and L.A. Wolsey, Cutting Planes for Integer Programs with General Integer Variables, *Mathematical Programming B* **81**, 201-214 (1998).

[9] G. Dahl, A. Martin and M. Stoer, Routing through virtual paths in layered telecommunication networks, Research Note N78/95, Telenor Research and Development, Kjeller, Norway (1995).

[10] B. Gavish and K. Altinkemer, Backbone network design tools with economic tradeoffs, *ORSA Journal on Computing* **2**, 58-76 (1990).

[11] T.L. Magnanti and P. Mirchandani, Shortest paths, single origin-destination network design and associated polyhedra, *Networks* **23**, 103-121 (1993).

[12] T.L. Magnanti, P. Mirchandani and R. Vachani, Modelling and Solving the Two-Facility Network Loading Problem, *Operations Research* **43**, 142-157 (1995).

[13] H. Marchand, Private Communication, December 1997.

[14] G.L. Nemhauser and L.A. Wolsey, *Integer and Combinatorial Optimization*, Wiley, New York (1988).

[15] PAMIPS: Development of parallel algorithms and software for mixed integer programming in industrial scheduling, Esprit Project 8755.

[16] C.P.M. van Hoesel, A.M.C.A. Koset, R.L.M.J. van de Leensel and M.W.P. Savelsbergh, Polyhderal results for the edge capacity polytope, *Mathematical Programming* **92**, 335-358 (2002).