

The Robust Network Loading Problem under Polyhedral Demand Uncertainty: Formulation, Polyhedral Analysis and Computations *

Ayşegül Altın, Hande Yaman, and Mustafa Ç. Pınar
Department of Industrial Engineering, Bilkent University, Ankara, Turkey
{aysegula,hyaman,mustafap}@bilkent.edu.tr

June 18, 2007

Abstract

We consider the Network Loading Problem under a polyhedral uncertainty description of traffic demands. After giving a compact multi-commodity formulation of the problem, we prove an unexpected decomposition property obtained from projecting out the flow variables, considerably simplifying the resulting polyhedral analysis and computations by doing away with metric inequalities, an attendant feature of most successful algorithms on the Network Loading Problem. Under a specific choice of the uncertainty description (hose model), we study the polyhedral aspects of Network Loading Problems, used as the basis of an efficient Branch-and-Cut algorithm supported by a simple heuristic for generating upper bounds. The results of extensive computational experiments on well-known network design instances are reported.

Keywords: Network loading problem, polyhedral demand uncertainty, hose model, robust network design, polyhedral analysis, branch-and-cut.

1 Introduction

Consider the following simple problem of deciding the optimal (i.e., resulting in the least total installation cost) number of devices of unit capacity to be installed on the links of the triangle-shaped network in Figure 1a to support the communication demands between the nodes. The number on each edge gives the capacity installation cost of a unit capacity device on that edge. The communication demands are forecasted to be one unit of traffic flow among all pairs of nodes in both directions.

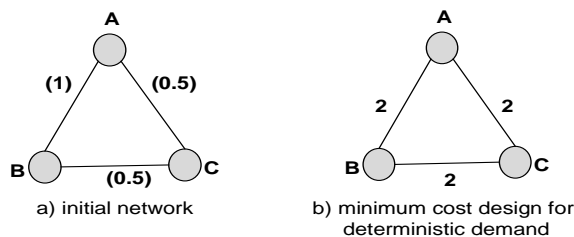


Figure 1: Example network for capacity loading

*Research partially supported by TUBITAK-CNRS (TUBITAK project no. 105M322, CNRS project BOSPHORE No. 10843 TD).

The optimal capacity installation is given in Figure 1b with a total cost of 4. Now, suppose that the communication demand from node A to nodes B and C are realized to be 0.999 and 1.001, which are 0.1% less and more than the expected unit demand, respectively. It is immediately seen that for even such small deviations from the expected value, our optimal design in Figure 1b would be infeasible.

In the present paper, we address the problem of designing networks that are able to support changing communication patterns in the least costly manner. More precisely, we study the *Network Loading Problem (NLP)* under a polyhedral uncertainty definition of possible traffic demands. For a given undirected network $G = (V, E)$ the traditional NLP deals with the design of a network by allocating discrete units of capacitated facilities on the links of G so as to support estimated pairwise demands between some endpoints of G . The purpose is to determine the least costly integer capacity configuration, where the installation cost is dependent on the specific edge and the type of the facility to be used. In the current work, we do not make the assumption of known demands but consider a polyhedral definition of feasible demands. The motivation for this study is to design networks robust to fluctuations in demand estimates, which are almost sure to happen in real life applications.

It is well accepted that data are always subject to some uncertainty in real life problems. In some occasions researchers completely ignore uncertainty by using nominal values that are supposed to represent the average behavior of the system. On the other hand, Stochastic Programming (SP) has been the traditional tool to deal with uncertainty, which yields decisions that might become infeasible with some probability. But there are cases where such a tolerance is not possible since it might have drastic consequences. This is the point where Robust Optimization (RO) is useful since it aims to make the best decision that remains ‘operational’ for any realization of data within a prescribed uncertainty set. The popularity of RO increases rapidly and an overview of some topics in RO domain is given by Ben-Tal and Nemirovski [12].

In RO, one decides on an uncertainty set \mathbb{U} , which defines all likely data realizations for which one is willing to take full responsibility, without making any assumption on the stochastic model of the data. Then, a robust design is the one whose worst performance over \mathbb{U} is the best. There are various ways of defining the uncertainty set \mathbb{U} : set of finite/infinite number of scenarios, finite intervals, a polyhedral or an ellipsoidal set. Moreover, the two measures of optimality are *absolute robustness* and *robust deviation*, which minimize the worst performance and the worst case deviation from optimality, respectively. For example, finding the least costly robust network design is related with absolute robustness whereas robust deviation measure can be used in determining the production plan minimizing the worst surplus cost.

The computational tractability of the RO models is an important issue. Ben-Tal and Nemirovski discuss the conditions for tractability of the *robust counterparts* of several linear programming problems in [10], [11], and [12]. They mark ellipsoidal uncertainty as a reasonable choice of uncertainty definition so as to balance tractability and flexibility of the robust model ([11]) without building an excess of conservatism into the robust solution. On the other hand, discrete optimization applications with polytopic or interval uncertainty definitions are considered in [3], [6], [13], [14].

In terms of the robust optimization methodology, which is mostly based on single-stage optimization, we can cite [6], [9], [31], and [33] as the examples of two-stage methods. Moreover, Bertsimas and Sim [13, 14], and Yaman et al. [40] discuss how the conservatism of the robust solution can be limited for different uncertainty definitions and measures of optimization. Finally, Atamtürk [3] gives some strong reformulations of the robust mixed 0-1 optimization models for a restricted level of uncertainty in the objective function coefficients.

An important component subject to uncertainty in network design problems is the traffic matrix, i.e., the demand between some origin-destination pairs. In practice, it is not likely for network designers to have a precise estimate of the traffic matrix, and ignoring this uncertainty may lead to a failure to meet service level agreements. Thus, in order to ensure effective network management, Gupta et al. [19] use the hose model of Duffield et al. [18] to design

Virtual Private Networks (with continuous capacity installation) capable to support any traffic matrix from the hose polyhedron. Gupta et al. [19] do not resort to integer programming and address the computational complexity of the resulting combinatorial optimization problems and approximation algorithms. In the same vein, Ben-Ameur and Kerivin [8] discuss the *polyhedral model* where the feasible demand realizations are defined by an arbitrary polyhedron. Their motivation is to determine splittable routing and edge capacity configurations that are durable in case of allowable shifts in demand expectations and are not too costly to implement. They use an algorithm where a vertex of the demand polyhedron is used to define a candidate design first. If this routing is not feasible for all valid demands, then the algorithm continues with a new traffic matrix. Later, Altın et al. [1] propose a compact mixed-integer programming model for Virtual Private Network design with continuous capacity expansion under unsplittable routing along with a branch-and-price-and-cut algorithm. Their model considers all traffic matrices simultaneously and they discuss a robust provisioning problem using available traffic statistics. More recently Altın et al. [2] study a single stage robust routing problem for a polytopic uncertainty set using a ‘quotient’ robust deviation measure.

NLP under deterministic traffic demands is widely studied in the literature since it can be applied to different contexts like telecommunication, computer networks, transportation problems or freight loading on trucks as mentioned in Magnanti et al. [29] and Berger et al. [15]. The generic problem has several versions. Namely, the number of different facility types available for installation give rise to variants of NLP. The most frequently studied cases are the single- ([5], [16], [22], [28], [30], [34]) and two-facility ([17], [20], [29]) versions whereas Magnanti and Mirchandani [27] discuss the three-facility problem as well. In addition, different cost functions are used, and the most common distinction is due to the existence of flow costs. We cite Avella et al. [7], Bienstock et al. [16], van Hoesel et al. [22], Magnanti and Mirchandani [27], and Magnanti et al. [28, 29] for the zero flow cost problems. On the other hand, Rardin and Wolsey [34], Bienstock and Günlük [17], and Günlük [20] include the flow cost in their models. The routing of demands is another source of diversity. Although the multi-path routing (*bifurcated routing* or *splittable flow*) is more popular ([5], [7], [16], [17], [20], [27], [28], [29], [34]) some important studies with single path routing (*non-bifurcated routing* or *unsplittable flow*) are also available ([5], [15], [21], [22]). Nonetheless, the Capacity Expansion Problem (CEP), where the decision is to determine a capacity installation plan by expanding the current edge capacities so as to route multi-commodity demands, is also closely related with the network loading problem. Interested reader can refer to Atamtürk and Günlük [4], Atamtürk and Rajan [5], Bienstock and Günlük [17], Berger et al. [15], and Günlük [20] for polyhedral results on the capacity expansion problem. The latter reference also provides a branch-and-cut algorithm for CEP.

Since NLP is strongly NP-hard there have been various efforts for solving it as efficiently as possible through the use of alternative formulations, heuristics, and by a thorough polyhedral analysis. Magnanti and Mirchandani [27] work on the single commodity restriction of the network loading problem with up to three facility alternatives. They try to reformulate the problem as a shortest path problem and solve it using heuristics. They show that the single facility case can be solved as a shortest path problem, which is not true for the two- and three- facility versions in general. Although the single commodity is not a practical assumption, the authors argue that their results could be an edge for the multi-commodity problems. Alternatively, single arc/edge restrictions of the problem have also been the focus of attention. Magnanti et al. [28] provide the single arc design problem, which is solved efficiently by a greedy method. Moreover, van Hoesel et al. [22] concentrate on the polyhedral properties of such subproblems since they can be seen as a relaxation of the original problem and the corresponding polyhedra are closely related. Similarly, Atamtürk and Günlük [4] study the network design arc set with variable bounds, which they emphasize to be a common component of a good number of network design problems.

Even though there have been different approaches in the literature for handling the network loading problem efficiently, most of them investigate the polyhedral properties of the related

problems. The common approach is to define some strong valid inequalities to strengthen the linear programming relaxations and thus improve the performance of the branch-and-cut or branch-and-bound based solution methods. Moreover, projection of the feasible set on to the space of discrete design variables has been a common point of interest ([5], [7], [16], [17], [27], [28], [29], [30], [34]). On the other hand, Berger et al. [15] discuss a tabu search heuristic.

In all references on deterministic NLP, point-to-point communication requirements are assumed to be given, and the future demand forecasts are used. Obviously, the demand between each origin-destination pair can be considered as a single commodity, and hence the overall problem is of a multi-commodity flow nature. Although the problem for single commodity flow with two facility types is very well studied and the polyhedra of feasible flows is fully characterized ([30]), the same problem with multi-commodity flows remains very hard, and its solution requires the use of metric inequalities to define the projection of the corresponding polyhedron on the space of discrete design variables ([7], [16], [17], [20], [27], [28], [29], [30], [32], [34]).

Against this background, the main contribution of the present paper to the existing body of literature on the NLP is to relax the assumption of known traffic demands prior to designing the network. While the NLP with known (deterministic) demands is well-studied, we are not aware of any other attempt to study the NLP under polyhedral demand uncertainty with the exception of an earlier reference by Karaşan et al. [24] where uncertainty was incorporated into the design of fiber optic networks with an emphasis on modeling rather than on a detailed polyhedral analysis and branch-and-cut. An astonishing feature of the present formulation for NLP with polyhedral uncertainty is that we avoid the use of metric inequalities due to a decomposition property obtained from a projection on the design components. A similar projection is used in Mirchandani [30] both for single- and multi-commodity NLP with deterministic demands where all extreme rays of the related projection cone for the single-commodity case were characterized. On the other hand, only necessary conditions were obtained for the multi-commodity variant. The difficulty of the latter problem is due to the coupling bundle constraints, which prevent the decomposition of the problem into single commodity subproblems. However, we by-pass that difficulty using the observation that the existence of a multi-commodity flow can be certified by checking the existence of many single commodity flows, i.e., the projection problem can be decomposed into many smaller single-commodity problems for which the results of [30] remain valid. This result considerably simplifies the formulations, and opens the way to a thorough polyhedral analysis of NLP under a specific hose uncertainty definition that is well accepted in the telecommunications literature. Based on the polyhedral analysis we develop a Branch-and-Cut algorithm along with a simple heuristic, and use it to solve several well-known network design instances.

The studies on the polyhedral properties of the deterministic NLP are limited to the case of at most three facility types where the capacity of a facility is an integer multiple of the capacity of the smaller facility, e.g., if there are three facility types, their capacities are 1, λ , and λC where λ and C are positive integers different from 1. The second main contribution of the present paper is that valid inequalities presented are for arbitrary number of facilities and arbitrary capacity structures.

The rest of the paper is organized as follows. In Section 2 we give a description of our problem along with a compact mixed-integer programming formulation. The aforementioned decomposition property from projection is developed in Section 3. We pass to a specific polyhedral uncertainty model, the hose model [18], in Section 4, and carry out a thorough polyhedral analysis for NLP under hose uncertainty in Section 5. Then we continue with separation algorithms for various valid inequalities and heuristics, all incorporated into a Branch-and-Cut algorithm in Section 6. We give an extensive summary of our computational results as well as comparisons with an off-the-shelf mixed-integer solver in Section 7 and conclude in Section 8 with some directions on future work.

2 Problem Definition

The deterministic NLP is defined as follows. Let $G = (V, E)$ be an undirected graph where V is the set of nodes and E is the set of edges. Let Q denote the set of commodities, i.e., $Q \subseteq \{(s, t) : s, t \in V, s \neq t\}$ such that $Q \neq \emptyset$. A set of alternative facility types having different capacities can be used to carry flow through the network. These facilities can only be installed on the edges (but not the nodes) of the network and the problem is about determining the number of facilities to be used on backbone edges such that all demand can be routed at minimum cost. Then NLP can be modeled as

$$\min \sum_{\{h,k\} \in E} \sum_{l \in L} p_{hk}^l y_{hk}^l \quad (1)$$

$$\text{s.t.} \quad \sum_{k: \{h,k\} \in E} (f_{hk}^{st} - f_{kh}^{st}) = \begin{cases} 1 & h = s \\ -1 & h = t \\ 0 & \text{otherwise} \end{cases} \quad \forall h \in V, (s, t) \in Q \quad (2)$$

$$\sum_{(s,t) \in Q} (f_{hk}^{st} + f_{kh}^{st}) d_{st} \leq \sum_{l \in L} C^l y_{hk}^l \quad \forall \{h, k\} \in E \quad (3)$$

$$y_{hk}^l \geq 0 \text{ and integer} \quad \forall \{h, k\} \in E, l \in L \quad (4)$$

$$f_{hk}^{st}, f_{kh}^{st} \geq 0 \quad \forall \{h, k\} \in E, (s, t) \in Q \quad (5)$$

where d_{st} is the forecasted demand from origin s to destination t for $(s, t) \in Q$, L is the set of facility alternatives, p_{hk}^l is the cost of installing one facility of type $l \in L$ on edge $\{h, k\} \in E$, and C^l is the transmission capacity of type $l \in L$ facility. On the other hand, the variables of the model are y_{hk}^l to show the number of type $l \in L$ facilities loaded on the edge $\{h, k\} \in E$ for the flow in both directions and f_{hk}^{st} to model the fraction of demand for commodity $(s, t) \in Q$ routed on the edge $\{h, k\} \in E$ in the direction from h to k . Constraints (2) are the usual flow conservation constraints for each demand pair at each node. Finally the constraints (3) are the edge capacity constraints, which ensure that the total capacity installed on each edge is enough to support the total flow on it in both directions.

The demand forecasts may not be precise most of the time, and it is very likely that the realization will be different from the expectations. Hence, rather than building our solution for a specific demand forecast, we use a general demand uncertainty definition with a motivation to design a network that is viable for any demand realization in the polyhedral set

$$D = \{d \in \mathbb{R}^{|Q|} : Ad \leq a, d \geq 0\} \quad (6)$$

where $A \in \mathbb{R}^{m \times |Q|}$ and $a \in \mathbb{R}^m$ with m being the number of linear inequalities that define the bounded and nonempty polyhedron D . This leads to the following polyhedral NLP model (NLP_{POL}):

$$\begin{aligned} & \min \sum_{\{h,k\} \in E} \sum_{l \in L} p_{hk}^l y_{hk}^l \\ \text{s.t.} & \quad (2), (4), (5) \\ & \max_{d \in D} \sum_{(s,t) \in Q} (f_{hk}^{st} + f_{kh}^{st}) d_{st} \leq \sum_{l \in L} C^l y_{hk}^l \quad \forall \{h, k\} \in E \end{aligned} \quad (7)$$

where f_{hk}^{st} is the fraction of the demand of commodity (s, t) routed on the directed arc (h, k) .

Unlike the case with known demands, NLP_{POL} is a semi-infinite optimization model due to the max problem we need to solve over the demand polyhedron for each edge $\{h, k\} \in E$. Next, following the method previously used in Altın et al. [1], we give a compact linear MIP formulation for NLP_{POL} .

Proposition 2.1. *Assuming that demand is subject to polyhedral uncertainty as in (6), NLP_{POL} can be reformulated as the following linear MIP model (NLP_{GD}):*

$$\begin{aligned} \min \quad & \sum_{\{h,k\} \in E} \sum_{l \in L} p_{hk}^l y_{hk}^l \\ \text{s.t.} \quad & (2), (4), (5) \\ & \sum_{z=1}^m a_z \lambda_z^{hk} \leq \sum_{l \in L} C^l y_{hk}^l \quad \forall \{h,k\} \in E \end{aligned} \quad (8)$$

$$f_{hk}^{st} + f_{kh}^{st} \leq \sum_{z=1}^m a_z^{st} \lambda_z^{hk} \quad \forall (s,t) \in Q, \{h,k\} \in E \quad (9)$$

$$\lambda_z^{hk} \geq 0 \quad \forall z = 1, \dots, m, \{h,k\} \in E. \quad (10)$$

Proof. Consider NLP_{POL} . Then for a given flow vector f and edge $\{h,k\} \in E$ the worst-case capacity requirement can be found by solving the following problem $P(h,k)$:

$$\max \quad \sum_{(s,t) \in Q} (f_{hk}^{st} + f_{kh}^{st}) d_{st} \quad (11)$$

$$\text{s.t.} \quad \sum_{(s,t) \in Q} a_z^{st} d_{st} \leq a_z \quad \forall z = 1, \dots, m \quad (12)$$

$$d_{st} \geq 0 \quad \forall (s,t) \in Q. \quad (13)$$

Notice that (11)-(13) is a linear programming model and its dual is $D(h,k)$

$$\begin{aligned} \min \quad & \sum_{z=1}^m a_z \lambda_z^{hk} \\ \text{s.t.} \quad & \sum_{z=1}^m a_z^{st} \lambda_z^{hk} \geq f_{hk}^{st} + f_{kh}^{st} \quad \forall (s,t) \in Q \\ & \lambda_z^{hk} \geq 0 \quad \forall z = 1, \dots, m, \end{aligned}$$

where λ_z^{hk} is the dual variable corresponding to (12). Since $P(h,k)$ is feasible and bounded, we can use duality transformation similar to the one of Soyster [37]. Hence for each edge $\{h,k\} \in E$, we can replace (7) with

$$\begin{aligned} \sum_{l \in L} C^l y_{hk}^l &\geq \min \sum_{z=1}^m a_z \lambda_z^{hk} \\ \text{s.t.} \quad & \sum_{z=1}^m a_z^{st} \lambda_z^{hk} \geq f_{hk}^{st} + f_{kh}^{st} \quad \forall (s,t) \in Q \\ & \lambda_z^{hk} \geq 0 \quad \forall z = 1, \dots, m. \end{aligned}$$

Then, we can omit the \min since the objective function (8) tries to minimize the sum of the design variables y_{hk}^l with nonnegative weights. \square

3 Projecting out the flow variables

As there is no flow cost in our model, we can obtain a formulation of our problem in the space of $\lambda \in \mathbb{R}^{|W||E|}$ and design variables $y \in \mathbb{Z}^{|L||E|}$.

Proposition 3.1. *For a given pair $(\bar{\lambda}, \bar{y})$, there exists a feasible flow $f \geq 0$ for NLP_{GD} satisfying (2), (5), and (9) if and only if*

$$\sum_{(s,t) \in Q} [-\alpha_s^{st} + \alpha_t^{st} + \sum_{\{h,k\} \in E} \beta_{hk}^{st} \sum_{z=1}^m (a_z^{st} \bar{\lambda}_z^{hk})] \geq 0 \quad (14)$$

for all $(\alpha, \beta) \geq 0$ such that

$$\alpha_h^{st} - \alpha_k^{st} + \beta_{hk}^{st} \geq 0 \quad \forall (s, t) \in Q, \{h, k\} \in E \quad (15)$$

$$-\alpha_h^{st} + \alpha_k^{st} + \beta_{hk}^{st} \geq 0 \quad \forall (s, t) \in Q, \{h, k\} \in E. \quad (16)$$

Proof. First we replace the flow conservation constraints (2) with equivalent inequality forms (see Mirchandani [30]). We associate dual variables $\hat{\beta}_{hk}^{st}$ and $\hat{\alpha}_h^{st}$ to (9) and the new flow conservation constraints, respectively. Then we apply Farkas' Lemma and let $\alpha_h^{st} = -\hat{\alpha}_h^{st}$ for $h \in V$ and $(s, t) \in Q$. \square

A similar projection is obtained in Mirchandani [30] both for single- and multi-commodity NLP with deterministic demands. Although he could characterize all extreme rays of the related projection cone for the single-commodity case, he could only give necessary conditions for the multi-commodity variant. The difficulty of the latter problem is due to the bundle constraints (3), which prevent the decomposition of the problem into single commodity subproblems. However, the situation is completely different for NLP_{GD} as Proposition 3.1 reveals an unexpectedly nice property of our problem. The result states that the existence of a multi-commodity flow f can be certified by checking the existence of $|Q|$ single commodity flows, i.e., the projection cone in (15)-(16) can be decomposed into $|Q|$ smaller cones with one cone for each commodity $(s, t) \in Q$. Thus, we will have the set

$$S_{st} = \{ \alpha^{st} \in \mathfrak{R}^{|V|}, \beta^{st} \in \mathfrak{R}^{|E|} : \begin{aligned} \beta_{hk}^{st} &\geq -\alpha_h^{st} + \alpha_k^{st} && \forall \{h, k\} \in E \\ \beta_{hk}^{st} &\geq \alpha_h^{st} - \alpha_k^{st} && \forall \{h, k\} \in E \\ \alpha_h^{st} &\geq 0 && \forall h \in V \\ \beta_{hk}^{st} &\geq 0 && \forall \{h, k\} \in E \end{aligned} \}$$

for each commodity $(s, t) \in Q$ and Mirchandani [30] has fully characterized the extreme rays of S_{st} .

For $S \subset V$, let $\delta(S)$ denote the set of edges with exactly one endpoint in S . Moreover, for ease of notation, we will denote an edge $\{h, k\}$ as e when there is no need to know its endpoints.

Proposition 3.2. [Mirchandani [30]] *Let $(\alpha^{st}, \beta^{st})$ be a ray of the pointed finitely generated cone S_{st} . Then $(\alpha^{st}, \beta^{st})$ is an extreme ray of S_{st} if and only if it is in one of the following forms:*

- Node Form: $\alpha_h^{st} = 1$ for all $h \in V$ and $\beta_e^{st} = 0$ for all $e \in E$.
- Arc Form: $\alpha_h^{st} = 0$ for all $h \in V$ and $\beta_{\bar{e}}^{st} = 1$ for some $\bar{e} \in E$ and all other entries are 0.
- Cutset Form: $\alpha_h^{st} = 1$ for all $h \in S$ and $\alpha_h^{st} = 0$ for all $h \in V \setminus S$. $\beta_e^{st} = 1$ if $e \in \delta(S)$ and 0 otherwise where $S \subset V$ and the subgraph induced by S is connected.

Then we state the valid inequalities implied by each form of extreme rays in the following remark using Proposition 3.2.

Remark 1. *The inequalities of type (14) implied by each form of the extreme rays are as follows:*

- Node Form: $0 \geq 0$
- Arc Form: $\sum_{z=1}^m a_z^{st} \lambda_z^e \geq 0 \quad \forall e \in E, \forall (s, t) \in Q$
- Cutset Form:

$$\sum_{e \in \delta(S)} \sum_{z=1}^m a_z^{st} \lambda_z^e \geq \begin{cases} 1 & \text{if } s \in S, t \in T \\ -1 & \text{if } s \in T, t \in S \\ 0 & \text{otherwise} \end{cases} \quad \forall (s, t) \in Q, S \subset V, T = V \setminus S.$$

The Node Form implies a redundant inequality. Similarly the inequalities obtained by using the Arc Form or the Cutset Form may become redundant depending on the entries a_z^{st} in (6). For example, if all a_z^{st} are nonnegative, then the Arc Form valid inequalities as well as the last two inequalities of the Cutset Form mentioned above are redundant.

Notice that the inequalities listed in Remark 1 are actually the feasibility cuts and must be included in the projection of NLP_{GD} onto the space of (λ, y) variables. Then, the mathematical model (NLP_{PRO}) in the space of λ and y variables is

$$\begin{aligned}
& \min \sum_{e \in E} \sum_{l \in L} p_e^l y_e^l \\
\text{s.t.} \quad & \sum_{z=1}^m a_z^{st} \lambda_z^e \geq 0 \quad \forall e \in E, \forall (s, t) \in Q \\
& \sum_{e \in \delta(S)} \left(\sum_{z=1}^m a_z^{st} \lambda_z^e \right) \geq \begin{cases} 1 & \text{if } s \in S, t \in T \\ -1 & \text{if } s \in T, t \in S \\ 0 & \text{otherwise} \end{cases} \quad \forall (s, t) \in Q, S \subset V, T = V \setminus S \\
& (4), (8), (10)
\end{aligned}$$

Clearly, NLP_{PRO} is a very general model since it can be used for any polyhedral demand definition with $|L|$ facility alternatives. In the sequel, we will first consider the well-known Hose Model of Duffield et al. [18] as the demand polyhedron, and present results on the polyhedral analysis of the problem for this uncertainty definition and multiple facility types.

4 The Network Loading Problem under Hose Demand Uncertainty

The Hose Model of Duffield et al. [18] was initially proposed in order to carry out flexible resource management in Virtual Private Networks (VPN). It finds acceptance especially from the telecommunication community since the number of end-points on a network is growing with more complicated communication patterns, and the hose model does not require a detailed traffic matrix estimate. Rather than the point-to-point demand estimations, it uses the traffic bandwidth of some special nodes called *VPN terminals* to characterize the feasible demand matrix realizations. Hence, the hose model specifies the polyhedra of demands as

$$D = \{d \in \mathbb{R}^{|Q|} : \sum_{t:(s,t) \in Q} d_{st} + \sum_{t:(t,s) \in Q} d_{ts} \leq b_s \quad \forall s \in W, \quad d_{st} \geq 0 \quad \forall (s, t) \in Q\} \quad (17)$$

where $W \subseteq V$ is the set of VPN terminals, i.e., $W = \{s \in V : \exists (s, t) \in Q \text{ or } (t, s) \in Q\}$ and b_s is the bandwidth capacity of the terminal node $s \in W$. The importance of the hose model can be demonstrated by returning to the simple example of the Introduction on Figure 1, where we consider a single facility type with unit capacity and the number on each edge is the capacity installation cost for that edge. Recall that the optimal capacity allocation would be as shown in Figure 1b with a total cost of 4 when the demands are assumed to be known. On the other hand, in the corresponding hose model each node would have a bandwidth of 4 units since this is the total flow incident to each node for the given demand forecast. Then the optimal design for the hose polyhedron is as shown in Figure 2 again with the same total cost. The important point here is that although the total design costs are the same for the two cases, the polyhedral design is more robust to fluctuations in demand. Under the scenario, considered in the Introduction, that the demand from node A to nodes B and C are realized to be 0.999 and 1.001, the deterministic design becomes unusable although the robust one remains operational. This example illustrates a very important advantage of using the hose uncertainty model. Using the hose specification, the network service provider would have a design flexible to accommodate

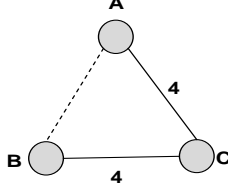


Figure 2: Minimum cost design for the hose model.

any arbitrary distribution of the total traffic provided that the total bandwidth capacity is not violated. In other words, it enables the transfer of unused capacity for a pairwise demand to another demand which goes beyond its estimation. Hence, the hose model yields a more flexible design than the deterministic one. In addition, the size of access links for the hose model can be less than required by the point-to-point pipes due to statistical multiplexing. Finally, from the customers' perspective, the system is easier to specify since the only necessary information is the aggregate bandwidth capacities rather than individual pairwise demands, which gets more and more difficult to predict as the number of endpoints and connectivity dynamics increase ([18], [23], [25]). In addition to the provisioning advantages the hose model provides, its prevalence would make the current work more acceptable especially for the telecommunication community.

Proposition 4.1. *The projection of NLP_{GD} on to the space of (λ, y) variables for the hose model (NLP_{hose}) is as follows:*

$$\begin{aligned} & \min \sum_{e \in E} \sum_{l \in L} p_e^l y_e^l \\ \text{s.t. } & \sum_{s \in W} b_s \lambda_s^e \leq \sum_{l \in L} C^l y_e^l \quad \forall e \in E \end{aligned} \quad (18)$$

$$\sum_{e \in \delta(S)} (\lambda_s^e + \lambda_t^e) \geq 1 \quad \forall (s, t) \in Q, S \subset V : s \in S, t \in V \setminus S \quad (19)$$

$$(4), (10)$$

Proof. The valid inequalities implied by the Arc Form extreme rays reduce to

$$\lambda_s^e + \lambda_t^e \geq 0 \quad \forall e \in E, (s, t) \in Q. \quad (20)$$

Moreover, the Cut Form valid inequalities are

$$\sum_{e \in \delta(S)} (\lambda_s^e + \lambda_t^e) \geq \begin{cases} 1 & \text{if } s \in S, t \in T \\ -1 & \text{if } s \in T, t \in S \\ 0 & \text{otherwise} \end{cases} \quad \forall S \subset V, T = V \setminus S. \quad (21)$$

However, the nonnegativity constraints (10) dominate (20) and the last two inequalities in (21). \square

5 Polyhedral Analysis

In this section we present results on the facets of the polyhedron associated with the network loading problem under Hose uncertainty NLP_{hose} .

In the sequel, we assume that

- i. C^l is a positive integer for $l \in L$,
- ii. for l_1 and l_2 in L such that $l_1 < l_2$ we have $C^{l_1} < C^{l_2}$.

Let $F = \{(\lambda, y) \in \mathbb{R}_+^{|W||E|} \times \mathbb{Z}_+^{|E||L|} : (18) \text{ and } (19)\}$ and $P = \text{conv}(F)$. Observe that adding constraints

$$\lambda_s^e \leq 1 \quad \forall s \in W, e \in E \quad (22)$$

does not change the validity of the model (see Karasan et al. [24]). Let $F' = F \cap \{(\lambda, y) \in \mathbb{R}_+^{|W||E|} \times \mathbb{Z}_+^{|E||L|} : (22)\}$ and $P' = \text{conv}(F')$.

For $s \in W$ and $e \in E$, let u_s^e be a unit vector of dimension $|W||E|$ where the entry corresponding to e and s is 1 and other entries are zero. For $e \in E$ and $l \in L$, let v_e^l be a unit vector of dimension $|E||L|$ where the entry corresponding to e and l is one and other entries are zero. Let M be a very large integer.

First, we investigate the dimension of the polyhedra P and P' .

Proposition 5.1. *Dimension of P and P' is $(|W| + |L|)|E|$.*

Proof. Suppose that all points in P (resp. P') satisfy $\alpha\lambda + \beta y = \gamma$. Let $e \in E$, $l \in L$ and $(\lambda, y) \in P$ (resp. $(\lambda, y) \in P'$). Observe that the vector $(\lambda, y + v_e^l)$ is also in P (resp. in P'). Hence $\beta_e^l = 0$. Let $s \in W$ and $e \in E$. Consider the vectors (λ, y) where $\lambda = \sum_{s' \in W} \sum_{e' \in E} u_{s'}^{e'} - u_s^e$ and $y = \sum_{l \in L} \sum_{e' \in E} M v_{e'}^l$, and $(\lambda + u_s^e, y)$. Since these two vectors are in P (resp. in P'), we have $\alpha_s^e = 0$. As there is no equality that is satisfied by all points in P (resp. in P'), polyhedron P (resp. P') is full-dimensional. \square

Let $F_\lambda = \text{Proj}_\lambda(F) = \{\lambda \in \mathbb{R}_+^{|W||E|} : (19)\}$ and $F'_\lambda = \text{Proj}_\lambda(F') = F_\lambda \cap \{\lambda \in \mathbb{R}_+^{|W||E|} : (22)\}$. Now, we relate facet defining inequalities of F_λ and F'_λ with those of P and P' .

Proposition 5.2. *Inequality $\sigma\lambda \geq \sigma_0$ is facet defining for P (resp. for P') if and only if it is facet defining for F_λ (resp. for F'_λ).*

Proof. If inequality $\sigma\lambda \geq \sigma_0$ is facet defining for P (resp. for P') then it is facet defining for F_λ (resp. for F'_λ) since $F_\lambda = \text{Proj}_\lambda(F)$ and both P and F_λ are full-dimensional (resp. $F'_\lambda = \text{Proj}_\lambda(F')$ and both P' and F'_λ are full-dimensional). Suppose that inequality $\sigma\lambda \geq \sigma_0$ is facet defining for F_λ (resp. for F'_λ) and all points in P (resp. in P') such that $\sigma\lambda = \sigma_0$ also satisfy $\alpha\lambda + \beta y = \gamma$. Let $e \in E$, $l \in L$ and $(\lambda, y) \in P$ (resp. in P') such that $\sigma\lambda = \sigma_0$. The vector $(\lambda, y + v_e^l)$ is also in P (resp. in P') and satisfies $\sigma\lambda = \sigma_0$. Hence $\beta_e^l = 0$. As $\sigma\lambda \geq \sigma_0$ is facet defining for F_λ (resp. for F'_λ) and $F_\lambda = \text{Proj}_\lambda(F)$ (resp. $F'_\lambda = \text{Proj}_\lambda(F')$), (α, γ) is a multiple of (σ, σ_0) . Thus inequality $\sigma\lambda \geq \sigma_0$ is facet defining for P (resp. for P'). \square

Results similar to Propositions 5.1 and 5.2, based on projection of polyhedra, are given in Labbe and Yaman [26].

For $e \in E$, define $F_e = \{(\lambda^e, y_e) \in \mathbb{R}_+^{|W|} \times \mathbb{Z}_+^{|L|} : (18)\}$, $P_e = \text{conv}(F_e)$, $F'_e = F_e \cap \{(\lambda^e, y_e) \in \mathbb{R}_+^{|W|} \times \mathbb{Z}_+^{|L|} : (22)\}$, and $P'_e = \text{conv}(F'_e)$. Observe that if $\delta(S) \setminus \{e\} \neq \emptyset$ for every $S \subset V$ such that there exists $(s, t) \in Q$ with $s \in S$ and $t \in V \setminus S$, then $F_e = \text{Proj}_{(\lambda^e, y_e)}(F)$ and $F'_e = \text{Proj}_{(\lambda^e, y_e)}(F')$. In the following proposition, we investigate how the facet defining inequalities of P_e and P'_e are related to those of P and P' .

Theorem 1. *Let $e \in E$ be such that $\delta(S) \setminus \{e\} \neq \emptyset$ for every $S \subset V$ such that there exists $(s, t) \in Q$ with $s \in S$ and $t \in V \setminus S$. Inequality $\alpha\lambda^e + \beta y_e \geq \gamma$ is facet defining for P_e (resp. for P'_e) if and only if it is facet defining for P (resp. for P').*

Proof. Let $e \in E$ be such that $\delta(S) \setminus \{e\} \neq \emptyset$ for every $S \subset V$ such that there exists $(s, t) \in Q$ with $s \in S$ and $t \in V \setminus S$. If inequality $\alpha\lambda^e + \beta y_e \geq \gamma$ is facet defining for P (resp. for P') then it defines a facet of P_e (resp. for P'_e) as $F_e = \text{Proj}_{(\lambda^e, y_e)}(F)$ and both P_e and P are full-dimensional (resp. $F'_e = \text{Proj}_{(\lambda^e, y_e)}(F')$ and P'_e and P' are full-dimensional).

Suppose that inequality $\alpha\lambda^e + \beta y_e \geq \gamma$ is facet defining for P_e (resp. for P'_e). Suppose also that all points in P (resp. in P') such that $\alpha\lambda^e + \beta y_e = \gamma$ also satisfy $\bar{\alpha}\lambda + \bar{\beta}y = \bar{\gamma}$. Let

$e' \in E \setminus \{e\}$, $l \in L$, and $(\lambda, y) \in P$ (resp. in P') be such that $\alpha\lambda^e + \beta y_e = \gamma$. The vector (λ, \bar{y}) where $\bar{y} = y + v_e^l$ is also in P (resp. in P') and satisfies $\alpha\lambda^e + \beta\bar{y}_e = \gamma$. Hence $\bar{\beta}_{e'}^l = 0$.

Let $s' \in W$ and $e' \in E \setminus \{e\}$. Let (λ^e, y_e) be a vector in P_e (resp. in P'_e) such that $\alpha\lambda^e + \beta y_e = \gamma$. Consider $(\bar{\lambda}, \bar{y})$ where $\bar{\lambda} = \sum_{s \in W} \lambda_s^e u_s^e + \sum_{s \in W} \sum_{e'' \in E \setminus \{e\}} u_{s'}^{e''} - u_{s'}^{e'}$ and $\bar{y} = \sum_{l \in L} \sum_{e'' \in E \setminus \{e\}} M v_{e''}^l + \sum_{l \in L} y_e^l v_e^l$. Since $\delta(S) \setminus \{e\} \neq \emptyset$ for every $S \subset V$ such that $(s, t) \in Q$ with $s \in S$ and $t \in V \setminus S$, this vector is in P (resp. in P') and satisfies $\alpha\bar{\lambda}^e + \beta\bar{y}_e = \gamma$. Consider also $(\hat{\lambda}, \hat{y})$ where $\hat{\lambda} = \bar{\lambda} + u_{s'}^{e'}$ and $\hat{y} = \bar{y}$. This vector is also in P (resp. in P') and satisfies $\alpha\hat{\lambda}^e + \beta\hat{y}_e = \gamma$. Hence $\bar{\alpha}_{s'}^{e'} = 0$.

Now we can conclude that $\bar{\alpha}\lambda + \bar{\beta}y = \bar{\gamma}$ is a multiple of $\alpha\lambda^e + \beta y_e = \gamma$ as $F_e = \text{Proj}_{(\lambda^e, y_e)}(F)$ (resp. $F'_e = \text{Proj}_{(\lambda^e, y_e)}(F')$). \square

For $S \subseteq V$, define $b(S) = \sum_{s \in S \cap W} b_s$ and $B(S) = \min\{b(S), b(V \setminus S)\}$. Notice that in the worst case all terminals in $S \subset V$ would want to use all of their bandwidths to exchange traffic with the nodes in $V \setminus S$. As a result, the worst case traffic on the cut $\delta(S)$ would be the minimum of these requirements, i.e., $B(S)$.

Let $S \subset V$ be such that the subgraphs induced by S and $V \setminus S$ are both connected. Let $y_{\delta(S)}$ be the restriction of the vector y to edges $e \in \delta(S)$, $F(S) = \{y_{\delta(S)} \in \mathbb{Z}_+^{|\delta(S)||L|} : \sum_{l \in L} \sum_{e \in \delta(S)} C^l y_e^l \geq B(S)\}$ and $P(S) = \text{conv}(F(S))$.

Proposition 5.3. *Let $S \subset V$ be such that the subgraphs induced by S and $V \setminus S$ are both connected and $B(S) > 0$. $F(S) = \text{Proj}_{y_{\delta(S)}}(F) = \text{Proj}_{y_{\delta(S)}}(F')$.*

Proof. Suppose that $S \subset V$ such that the subgraphs induced by S and $V \setminus S$ are both connected and $B(S) > 0$. Let $y_{\delta(S)} \in F(S)$ and assume that $B(S) = b(S)$. The proof for the other case is similar. Define $(\hat{\lambda}, \hat{y})$ as follows: $\hat{\lambda}_s^e = 1$ for $e \in E \setminus \delta(S)$ and $s \in W$, $\hat{\lambda}_s^e = 0$ for $e \in \delta(S)$ and $s \in W \setminus S$, $\hat{y} = \sum_{l \in L} \sum_{e \in \delta(S)} y_e^l v_e^l + \sum_{l \in L} \sum_{e \in E \setminus \delta(S)} M v_e^l$. Notice that since the subgraphs induced by S and $V \setminus S$ are both connected, for $S' \subset V$ such that $S' \neq S$, $S' \neq V \setminus S$, and $\delta(S') \setminus \delta(S) \neq \emptyset$. Hence, if $(\hat{\lambda}, \hat{y})$ satisfies

$$\sum_{e \in \delta(S)} \hat{\lambda}_s^e = 1 \quad \forall s \in S \cap W \quad (23)$$

$$\sum_{s \in S \cap W} b_s \hat{\lambda}_s^e \leq \sum_{l \in L} C^l \hat{y}_e^l \quad \forall e \in \delta(S) \quad (24)$$

$$\hat{\lambda}_s^e \geq 0 \quad \forall s \in S \cap W, e \in \delta(S) \quad (25)$$

then $(\hat{\lambda}, \hat{y})$ is in P and P' . By Farkas Lemma, this system is feasible if and only if

$$\sum_{s \in S \cap W} \alpha_s + \sum_{e \in \delta(S)} \sum_{l \in L} C^l \hat{y}_e^l \beta_e \geq 0 \quad (26)$$

for all (α, β) such that

$$\begin{aligned} \alpha_s + b_s \beta_e &\geq 0 & \forall s \in S \cap W, e \in \delta(S) \\ \beta_e &\geq 0 & \forall e \in \delta(S). \end{aligned}$$

Since $\sum_{l \in L} C^l \hat{y}_e^l \geq 0$ for all $e \in \delta(S)$, we can limit ourselves to (α, β) such that $\beta_e = \max_{s \in S \cap W} \left(\frac{-\alpha_s}{b_s} \right)^+$ for all $e \in \delta(S)$. Now, we can rewrite the left hand side of (26) as

$$\sum_{s \in S \cap W} \alpha_s + \sum_{e \in \delta(S)} \sum_{l \in L} C^l \hat{y}_e^l \max_{s \in S \cap W} \left(\frac{-\alpha_s}{b_s} \right)^+. \quad (27)$$

If $\max_{s \in S \cap W} \left(\frac{-\alpha_s}{b_s} \right)^+ = 0$, then $\alpha_s \geq 0$ for all $s \in S \cap W$, and (27) is nonnegative. Suppose now that $\max_{s \in S \cap W} \left(\frac{-\alpha_s}{b_s} \right)^+ = \frac{-\alpha_{s'}}{b_{s'}} > 0$ for some $s' \in S \cap W$. Then (27) is

$$\sum_{s \in S \cap W} \alpha_s - \frac{\alpha_{s'}}{b_{s'}} \sum_{e \in \delta(S)} \sum_{l \in L} C^l \hat{y}_e^l. \quad (28)$$

Now as $\sum_{e \in \delta(S)} \sum_{l \in L} C^l \hat{y}_e^l \geq b(S)$ and $\frac{-\alpha_{s'}}{b_{s'}} > 0$, (28) is greater than or equal to

$$\sum_{s \in S \cap W} \alpha_s - \frac{\alpha_{s'}}{b_{s'}} b(S) = \sum_{s \in S \cap W} \left(\alpha_s - \frac{\alpha_{s'}}{b_{s'}} b_s \right).$$

As $\frac{\alpha_s}{b_s} \geq \frac{\alpha_{s'}}{b_{s'}}$ for all $s \in S \cap W$, $\sum_{s \in S \cap W} \left(\alpha_s - \frac{\alpha_{s'}}{b_{s'}} b_s \right) \geq 0$. So the system (23)-(25) always has a solution and $F(S) \subseteq \text{Proj}_{y_{\delta(S)}}(F)$ and $F(S) \subseteq \text{Proj}_{y_{\delta(S)}}(F')$. As $\sum_{l \in L} \sum_{e \in \delta(S)} C^l y_e^l \geq B(S)$ is a valid inequality for F and F' , $\text{Proj}_{y_{\delta(S)}}(F) \subseteq F(S)$ and $\text{Proj}_{y_{\delta(S)}}(F') \subseteq F(S)$. \square

Now, we can relate facet defining inequalities of $P(S)$ to those of P .

Theorem 2. *Let $S \subset V$ be such that the subgraphs induced by S and $V \setminus S$ are both connected and $B(S) > 0$. If inequality $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l y_e^l \geq \beta_0$ is facet defining for $P(S)$ and for each $e' \in \delta(S)$ there exists a vector $y_{\delta(S)} \in F(S)$ such that $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l y_e^l = \beta_0$ and $\sum_{l \in L} C^l y_{e'}^l > B(S)$, then the inequality is facet defining for P .*

Proof. Let $S \subset V$ be such that the subgraphs induced by S and $V \setminus S$ are both connected and suppose that inequality $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l y_e^l \geq \beta_0$ is facet defining for $P(S)$ and for each $e' \in \delta(S)$ there exists a vector $y_{\delta(S)} \in F(S)$ such that $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l y_e^l = \beta_0$ and $\sum_{l \in L} C^l y_{e'}^l > B(S)$. Assume that $B(S) = b(S)$. The proof for the other case is similar.

Suppose that all points in P such that $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l y_e^l = \beta_0$ also satisfy $\bar{\alpha}\lambda + \bar{\beta}y = \bar{\gamma}$. Let $e' \in E \setminus \delta(S)$, $l' \in L$ and $(\lambda, y) \in P$ be such that $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l y_e^l = \beta_0$. The vector (λ, \bar{y}) where $\bar{y} = y + v_{e'}^{l'}$, is also in P and satisfies $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l \bar{y}_e^l = \beta_0$. Hence $\bar{\beta}_{e'}^{l'} = 0$.

Define $\lambda_{S \cap W}^{\delta(S)}$ to be the restriction of the vector λ to its entries with edges in $\delta(S)$ and vertices in $S \cap W$. Let $y_{\delta(S)} \in F(S)$ be such that $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l y_e^l = \beta_0$ and $\lambda_{\delta(S)}^{S \cap W}$ be a solution to the system (23)-(25) with $y_{\delta(S)}$. Let $s' \in W$ and $e' \in E \setminus \delta(S)$. Let $\bar{\lambda} = \sum_{s \in W} \sum_{e \in E \setminus \delta(S)} u_s^e - u_{s'}^{e'} + \sum_{s \in S \cap W} \sum_{e \in \delta(S)} \lambda_s^e u_s^e$ and $\bar{y} = \sum_{e \in \delta(S)} y_e^l v_e^l + \sum_{e \in E \setminus \delta(S)} M v_e^l$. The vector $(\bar{\lambda}, \bar{y})$ is in P and satisfies $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l \bar{y}_e^l = \beta_0$. Consider also $(\hat{\lambda}, \bar{y})$ where $\hat{\lambda} = \bar{\lambda} + u_{s'}^{e'}$. This latter vector is also in P (resp. in P') and satisfies $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l \bar{y}_e^l = \beta_0$. Hence $\bar{\alpha}_{s'}^{e'} = 0$.

Let $s' \in W$, $e' \in \delta(S)$ and $y_{\delta(S)} \in F(S)$ be such that $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l y_e^l = \beta_0$ and $\sum_{l \in L} C^l y_{e'}^l > b(S)$. Let $\lambda_{\delta(S)}^{S \cap W}$ be a solution to the system (23)-(25) with $y_{\delta(S)}$. Consider the vector $(\bar{\lambda}, \bar{y})$ where $\bar{\lambda} = \sum_{s \in W} \sum_{e \in E \setminus \delta(S)} u_s^e + \sum_{s \in S \cap W} \sum_{e \in \delta(S)} \lambda_s^e u_s^e$ and $\bar{y} = \sum_{e \in \delta(S)} y_e^l v_e^l + \sum_{e \in E \setminus \delta(S)} M v_e^l$. This vector is in P and satisfies $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l \bar{y}_e^l = \beta_0$. Now let $\epsilon > 0$ be very small and consider also the vector $(\hat{\lambda}, \bar{y})$ where $\hat{\lambda} = \bar{\lambda} + \epsilon u_{s'}^{e'}$. This vector is also in P since $\sum_{l \in L} C^l y_{e'}^l > b(S)$ and satisfies $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l \bar{y}_e^l = \beta_0$. Thus $\bar{\alpha}_{s'}^{e'} = 0$.

As $F(S) = \text{Proj}_{y_{\delta(S)}}(F)$ and as $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l y_e^l \geq \beta_0$ is facet defining for $P(S)$, $\bar{\alpha}\lambda + \bar{\beta}y = \bar{\gamma}$ is a multiple of $\sum_{l \in L} \sum_{e \in \delta(S)} \beta_e^l y_e^l = \beta_0$. \square

For $S \subset V$ and $l \in L$, let

$$Y^l(S) = \sum_{e \in \delta(S)} y_e^l,$$

$$r^l(S) = b(S) - \left\lfloor \frac{b(S)}{C^l} \right\rfloor C^l,$$

and

$$R^l(S) = \lceil B(S) \rceil - \left\lfloor \frac{\lceil B(S) \rceil}{C^l} \right\rfloor C^l.$$

For l_1 and l_2 in L , let

$$f(l_1, l_2) = C^{l_1} - \left\lfloor \frac{C^{l_1}}{C^{l_2}} \right\rfloor C^{l_2}.$$

Now, we modify two well-known families of valid inequalities for the NLP to render them valid for our problem. These inequalities are the cutset inequalities and arc residual capacity inequalities (see e.g. Magnanti et al. [28]). Both inequalities can be generated as Mixed Integer Rounding (MIR) inequalities. Let $X = \{(x_1, x_2) \in \mathbb{R}_+ \times \mathbb{Z} : x_1 + x_2 \geq \mu\}$. The MIR inequality $x_1 \geq (\mu - \lfloor \mu \rfloor)(\lceil \mu \rceil - x_2)$ is valid for X (see Wolsey [38]).

The special cases of the cutset and residual capacity inequalities for the network loading problem under Hose uncertainty with a single facility type are presented and used in Karaşan et al. [24] in order to strengthen the LP relaxation bound. However, they neither give any polyhedral results nor implement a Branch-and-Cut algorithm.

The set $F(S)$ is an integer knapsack cover set and the polyhedral properties of $P(S)$ are studied in Yaman [39]. A family of valid inequalities, called “lifted rounding inequalities”, are generated using sequence independent lifting. These inequalities generalize the cutset inequalities. As they are valid for $P(S)$, they are also valid for P and P' .

Proposition 5.4. *For $S \subset V$ and $l^* \in L$ such that $R^{l^*}(S) > 0$, the cutset inequality*

$$\begin{aligned} & \sum_{l \in L: C^l < B(S)} \left(R^{l^*}(S) \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor + \min \left\{ f(l, l^*), R^{l^*}(S) \right\} \right) Y^l(S) \\ & + \sum_{l \in L: C^l \geq B(S)} R^{l^*}(S) \left\lfloor \frac{B(S)}{C^{l^*}} \right\rfloor Y^l(S) \geq R^{l^*}(S) \left\lfloor \frac{B(S)}{C^{l^*}} \right\rfloor \end{aligned} \quad (29)$$

is valid for P and P' .

Yaman [39] proves that if $C^1 = 1$, then the cutset inequality (29) for $l^* \in L$ such that $R^{l^*}(S) > 0$ is facet defining for $P(S)$. Using Theorem 2, we can state the following proposition.

Proposition 5.5. *Let $S \subset V$ be such that the subgraphs induced by S and $V \setminus S$ are both connected and $l^* \in L$ be such that $R^{l^*}(S) > 0$. If $C^1 = 1$, then the cutset inequality (29) is facet defining for P .*

Proof. Suppose that $S \subset V$ such that the subgraphs induced by S and $V \setminus S$ are both connected, $l^* \in L$ such that $R^{l^*}(S) > 0$ and $C^1 = 1$. The cutset inequality (29) is facet defining for $P(S)$ ([39]). To show that it also defines a facet of P , we need to prove, for each $e \in \delta(S)$, the existence of a vector $y_{\delta(S)} \in F(S)$ which satisfies the inequality (29) as equality and $\sum_{l \in L} C^l y_e^l > B(S)$. For edge $e \in \delta(S)$, consider the vector $y_{\delta(S)} = \left\lfloor \frac{B(S)}{C^{l^*}} \right\rfloor v_e^{l^*}$. This vector satisfies (29) as equality, and we have $\sum_{l \in L} C^l y_e^l > B(S)$ since $R^{l^*}(S) > 0$. \square

Notice that if $C^2, \dots, C^{|L|}$ are divisible by C^1 , then we can scale the b_s values and the C^l values by dividing with C^1 so that $C^1 = 1$.

If $|L| = 1$ and $R^1(S) > 0$, then the cutset inequality (29) is facet defining for P for $S \subset V$ such that the subgraphs induced by S and $V \setminus S$ are both connected.

Next, we generate residual capacity inequalities as MIR inequalities.

Proposition 5.6. Let $e \in E$, $l^* \in L$ and $S \subseteq W$ be such that $r^{l^*}(S) > 0$. The residual capacity inequality

$$\sum_{l \in L} \left(r^{l^*}(S) \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor + \min \{ f(l, l^*), r^{l^*}(S) \} \right) y_e^l + \sum_{s \in S} b_s (1 - \lambda_s^e) \geq r^{l^*}(S) \left\lfloor \frac{b(S)}{C^{l^*}} \right\rfloor \quad (30)$$

is valid for P' .

Proof. Inequality $\sum_{s \in S} b_s \lambda_s^e \leq \sum_{l \in L} C^l y_e^l$ is a valid inequality for P' . We substitute $\bar{\lambda}_s^e = 1 - \lambda_s^e$ for $s \in S$ and obtain $b(S) \leq \sum_{l \in L} C^l y_e^l + \sum_{s \in S} b_s \bar{\lambda}_s^e$. Let $l^* \in L$ and divide the last inequality with C^{l^*} . This yields $\frac{b(S)}{C^{l^*}} \leq \sum_{l \in L} \frac{C^l}{C^{l^*}} y_e^l + \sum_{s \in S} \frac{b_s}{C^{l^*}} \bar{\lambda}_s^e$. Let $L_u = \{l \in L : f(l, l^*) > r^{l^*}(S)\}$ and $L_d = L \setminus L_u$. The last inequality implies

$$\frac{b(S)}{C^{l^*}} \leq \sum_{l \in L_d} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor y_e^l + \sum_{l \in L_d} \frac{f(l, l^*)}{C^{l^*}} y_e^l + \sum_{l \in L_u} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor y_e^l + \sum_{s \in S} \frac{b_s}{C^{l^*}} \bar{\lambda}_s^e.$$

Here $\sum_{l \in L_d} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor y_e^l + \sum_{l \in L_u} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor y_e^l$ is integer and $\sum_{l \in L_d} \frac{f(l, l^*)}{C^{l^*}} y_e^l + \sum_{s \in S} \frac{b_s}{C^{l^*}} \bar{\lambda}_s^e$ is a non-negative real. Hence the MIR inequality

$$\sum_{l \in L_d} \frac{f(l, l^*)}{C^{l^*}} y_e^l + \sum_{s \in S} \frac{b_s}{C^{l^*}} \bar{\lambda}_s^e \geq \frac{r^{l^*}(S)}{C^{l^*}} \left(\left\lfloor \frac{b(S)}{C^{l^*}} \right\rfloor - \sum_{l \in L_d} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor y_e^l - \sum_{l \in L_u} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor y_e^l \right)$$

is valid for P' . Multiplying both sides with C^{l^*} and organizing the terms, we obtain

$$\sum_{l \in L_d} \left(f(l, l^*) + r^{l^*}(S) \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \right) y_e^l + \sum_{l \in L_u} r^{l^*}(S) \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor y_e^l + \sum_{s \in S} b_s \bar{\lambda}_s^e \geq r^{l^*}(S) \left\lfloor \frac{b(S)}{C^{l^*}} \right\rfloor$$

which is the same as

$$\sum_{l \in L} \left(r^{l^*}(S) \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor + \min \{ f(l, l^*), r^{l^*}(S) \} \right) y_e^l + \sum_{s \in S} b_s \bar{\lambda}_s^e \geq r^{l^*}(S) \left\lfloor \frac{b(S)}{C^{l^*}} \right\rfloor.$$

Substituting $\bar{\lambda}_s^e = 1 - \lambda_s^e$ for $s \in S$ yields inequality (30). \square

If $|L| = 1$, the residual capacity inequality becomes

$$r^1(S) y_e^1 + \sum_{s \in S} b_s (1 - \lambda_s^e) \geq r^1(S) \left\lfloor \frac{b(S)}{C^1} \right\rfloor. \quad (31)$$

Magnanti et al. [28] prove the following: If $\left\lfloor \frac{b(S)}{C^1} \right\rfloor \geq 2$, then this inequality defines a facet of P'_e . If $\left\lfloor \frac{b(S)}{C^1} \right\rfloor = 1$ then the inequality defines a facet of P'_e if $|S| = 1$. Using Theorem 1, we can prove the following:

Proposition 5.7. Let $e \in E$ be such that $\delta(S') \setminus \{e\} \neq \emptyset$ for every $S' \subset V$ such that there exists $(s, t) \in Q$ with $s \in S'$ and $t \in V \setminus S'$. Suppose that $|L| = 1$ and let $S \subseteq W$ be such that $r^1(S) > 0$. The residual capacity inequality (31) defines a facet of P' if $\left\lfloor \frac{b(S)}{C^1} \right\rfloor \geq 2$ or if $\left\lfloor \frac{b(S)}{C^1} \right\rfloor = 1$ and $|S| = 1$.

6 Branch-and-Cut Algorithm

In NLP_{hose} , we have the constraints (19), which can be exponential in number. Naturally, it is not reasonable and possible to include all such inequalities at the outset and try to solve the resulting model as it is. Therefore, we will use a branch-and-cut algorithm, which starts with a larger feasible set $\{(\lambda, y) \in \mathbb{R}_+^{|W||E|} \times \mathbb{Z}_+^{|E||L|} : (18)\}$ and adds the violated valid inequalities at each iteration. In the rest of this section, we will first explain the separation algorithms we use for the feasibility cuts (19) as well as the demand cutset (29), and residual capacity (30) inequalities. Then, we describe briefly our upper bounding procedure.

6.1 Separation of Feasibility Cuts

Inequalities (19) can be separated by solving minimum cut problems. Given a pair $(\bar{\lambda}, \bar{y})$, we construct an auxiliary graph $\tilde{G}_{st} = (V, E)$ for each origin-destination pair $(s, t) \in Q$ such that the capacity of each edge $e \in E$ is set to be $\bar{\lambda}_s^e + \bar{\lambda}_t^e$. If the capacity of the minimum cut $C(s, t)$ separating s and t is less than 1, then we have a violated inequality (19) for (s, t) . Otherwise, no inequality (19) is violated for (s, t) by the pair $(\bar{\lambda}, \bar{y})$.

6.2 Separation of Demand Cutset Inequalities

We have a heuristic separation algorithm for the demand cutset inequalities (29). For each commodity $(s, t) \in Q$, we use the cut $C(s, t)$ for which a feasibility cut (18) is violated. If the pair $(\bar{\lambda}, \bar{y})$ also violates a demand cutset inequality for $C(s, t)$, then we add the corresponding cut to the problem.

6.3 Separation of Residual Capacity Inequalities

We do not know any polynomial time algorithm to separate the residual capacity inequalities (30). But we can separate a relaxed version of these inequalities in polynomial time.

Let $e \in E$, $l^* \in L$, and $S(e, l^*) \subseteq W$. Define the relaxed residual capacity inequality as

$$\sum_{l \in L} \left(C^l - (C^{l^*} - r^{l^*}(S(e, l^*))) \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \right) \bar{y}_e^l + \sum_{s \in S(e, l^*)} b_s(1 - \lambda_s^e) \geq r^{l^*}(S(e, l^*)) \left\lceil \frac{b(S(e, l^*))}{C^{l^*}} \right\rceil. \quad (32)$$

This inequality is valid for P' as it is implied by inequality (30). Moreover, it is a MIR inequality.

For a given edge $e \in E$, a facility type $l^* \in L$, and a pair $(\bar{\lambda}_e, \bar{y}_e)$, finding a violated relaxed residual capacity inequality or showing that there is no such inequality is equivalent to solving the problem

$$\phi(e, l^*) = \min_{S(e, l^*) \subseteq W} \left\{ \sum_{s \in S(e, l^*)} b_s(1 - \bar{\lambda}_s^e) - r^{l^*}(S(e, l^*)) \left(\left\lceil \frac{b(S(e, l^*))}{C^{l^*}} \right\rceil - \sum_{l \in L} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \bar{y}_e^l \right) \right\}.$$

If $\sum_{l \in L} \left(C^l - C^{l^*} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \right) \bar{y}_e^l + \phi(e, l^*) \geq 0$ then all relaxed residual capacity inequalities for $e \in E$ and $l^* \in L$ are satisfied by $(\bar{\lambda}_e, \bar{y}_e)$. Otherwise, we have a violated relaxed residual capacity inequality defined by a minimizing set $S(e, l^*)$. Since the relaxed residual capacity inequality is a MIR inequality, if $\sum_{l \in L} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \bar{y}_e^l \geq \left\lceil \frac{b(S(e, l^*))}{C^{l^*}} \right\rceil$ or $\sum_{l \in L} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \bar{y}_e^l \leq \left\lceil \frac{b(S(e, l^*))}{C^{l^*}} \right\rceil - 1$, it cannot be violated. Then using the arguments in Atamtürk and Rajan [5], we can show that the relaxed residual capacity inequalities can be separated in the following way: Let

$$S(e, l^*) = \left\{ s \in W : \bar{\lambda}_s^e > \sum_{l \in L} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \bar{y}_e^l - \left\lceil \frac{b(S(e, l^*))}{C^{l^*}} \right\rceil \right\}$$

and

$$\Psi(S(e, l^*)) = \sum_{s \in S(e, l^*)} b_s(1 - \bar{\lambda}_s^e) - r^{l^*}(S(e, l^*)) \left(\left\lceil \frac{b(S(e, l^*))}{C^{l^*}} \right\rceil - \sum_{l \in L} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \bar{y}_e^l \right).$$

If $\left\lfloor \sum_{l \in L} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \bar{y}_e^l \right\rfloor < \frac{b(S(e, l^*))}{C^{l^*}} < \left\lceil \sum_{l \in L} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \bar{y}_e^l \right\rceil$ and $\sum_{l \in L} \left(C^l - C^{l^*} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \right) \bar{y}_e^l + \Psi(S(e, l^*)) < 0$, then the relaxed residual capacity inequality for edge $e \in E$, $l^* \in L$, and the set $S(e, l^*)$ is violated. Otherwise, no inequality (32) for this choice of edge and edge type is violated. Hence separation of the relaxed residual capacity inequalities can be done in $O(|E||L||W|)$ time. Note

that we solve the separation problem for the relaxed residual capacity inequalities (32) but add the stronger inequalities (30) in case of a violation. Another alternative is to use a hybrid separation method where, for each edge e and facility type l^* , we check if any strong inequality (30) is violated for the set $S(e, l^*)$. We have implemented both methods and observed that the former method is as efficient as the latter one. Hence, we use the separation algorithm displayed in Algorithm 1 for the residual capacity inequalities (30).

Algorithm 1 Residual Capacity Inequality Separation

for all edge $e \in E$ **do**
 for all facility type $l^* \in L$ **do**
 $\bar{Y}_e^{l^*} := \sum_{l \in L} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \bar{y}_e^l$
 $S(e, l^*) := \{s \in W : \bar{\lambda}_s^e > \bar{Y}_e^{l^*} - \lfloor \bar{Y}_e^{l^*} \rfloor\}$
 if $\lfloor \bar{Y}_e^{l^*} \rfloor < \frac{b(S(e, l^*))}{C^{l^*}} < \lceil \bar{Y}_e^{l^*} \rceil$ and $\sum_{l \in L} \left(C^l - C^{l^*} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \right) \bar{y}_e^l + \sum_{s \in S(e, l^*)} b_s (1 - \bar{\lambda}_s^e) - r^{l^*}(S(e, l^*)) \left(\left\lceil \frac{b(S(e, l^*))}{C^{l^*}} \right\rceil - \sum_{l \in L} \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor \bar{y}_e^l \right) < 0$ **then**
 Add the violated residual capacity inequality
 $\sum_{l \in L} (r^{l^*}(S(e, l^*)) \left\lfloor \frac{C^l}{C^{l^*}} \right\rfloor + \min\{f(l, l^*), r^{l^*}(S(e, l^*))\}) y_e^l + \sum_{s \in S(e, l^*)} b_s (1 - \lambda_s^e) \geq r^{l^*}(S(e, l^*)) \left\lceil \frac{b(S(e, l^*))}{C^{l^*}} \right\rceil$

6.4 Heuristics

Given the difficulty of the problem, we expect it to be useful to incorporate some approximation heuristics into our B&C algorithm. These algorithms yield easy-to-compute upper bounds, useful especially for the large instances that we could not solve to optimality within our time limits.

We apply a simple rounding heuristic to get upper bounds on the optimal solution. So, at each node of the B&C tree, if we cannot find any violated inequality then we have a feasible solution for the LP relaxation of the NLP_{hose} problem. Let $(\bar{\lambda}, \bar{y})$ be the current solution. We simply generate a feasible solution $(\bar{\lambda}, \hat{y})$ such that $\hat{y}_e^l = \lceil \bar{y}_e^l \rceil$ for all $e \in E$ and $l \in L$. Bienstock et al. [16] also use a similar method and mention that it is efficient.

We have also tried a more advanced heuristic, which is an adaptation of the approximation algorithm developed in Gupta et al. [19] for designing Virtual Private Networks with continuous capacity reservation. However, the results were not promising for our problem and the simple rounding heuristic was superior. Thus, we use only the rounding heuristic in our B&C algorithm.

7 Experimental Results

In this section we report the results of a computational study for NLP_{hose} with single facility and two facilities. Let NLP_{GD}^{hose} be the NLP_{GD} model for the hose uncertainty definition, which we solve using Cplex. Then, we compare the performance of the B&C algorithm with that of Cplex on instances from the network design literature. Most of these instances, namely *polska*, *dfn*, *newyork*, *france*, *janos*, *atlanta*, *tai*, *nobel-eu*, *pioro*, *gui39*, *cost266*, *norway*, and *sun*, are from the SNDLIB web site ([36]) whereas the remaining 7 are the ones used in Altın et al. [1] for a VPN design problem. For the SNDLIB instances the average pairwise demand estimates d_{st} are available. Hence, in order to generate the corresponding hose polyhedron, we let the bandwidth of each terminal node to be the total demand incident to it. In other words, we let $b_s = \sum_{t \in W \setminus \{s\}} (d_{st} + d_{ts})$ for $s \in W$ be the terminal node bandwidths.

We have used AMPL to model the single and two facility variants of NLP_{GD}^{hose} as well as Cplex 9.1 MIP solver to solve them. The B&C algorithm is implemented in C using MINTO (Mixed INTEger Optimizer [35]) and Cplex 9.1 as LP solver. We have set a two-hour time

limit both for AMPL and MINTO. The branching rule for the B&C algorithm is to choose the integer variable with fractional part closest to 0.5 to create two subproblems at each node of the B&C tree, where the current solution is not integral. Moreover, node selection is done using the best-bound search of the B&C tree. We present our test results for single- and two-facility cases in Section 7.1 and 7.2, respectively. The entries in the tables contain the following information:

- the instance characteristics, i.e., the name of the instance as well as the numbers of nodes, links, and terminals¹,
- facility capacities, i.e., C for the single facility case, C^1 and C^2 for the two facility case,
- the best upper bound z_{cp} and CPU time t_{cp} of Cplex,
- the best upper bound $z_{B\&C}$ and CPU time $t_{B\&C}$ of the B&C algorithm,
- the number of B&C nodes for Cplex $\#_{cp}$,
- the number of B&C nodes $\#_{B\&C}$,
- the gap at termination for Cplex g_{cp} ,
- the gap at termination for B&C $g_{B\&C}$,

where gap at termination is the gap between the best upper and lower bounds at termination for each method. Given the difficulty of the problem, it is not surprising that some of the instances are not solved to optimality within two hours time limit. We indicate such cases with a * in the corresponding z column. Moreover, the gaps at termination (g) for them are provided under the corresponding time (t) columns in parenthesis. Besides, we mark those cases for which solving the NLP_{GD}^{hose} or NLP_{hose} models did not yield even a feasible solution within the time limit with *NoI* whereas *MA* indicates a termination due to insufficient memory allocation. The gap at termination for such cases are noted as INF in all the result tables. Finally, note that all solution times are in CPU seconds.

7.1 Single Facility NLP_{hose}

For the single facility case, we assume that there is only one type of facility available with a capacity of C units. Then the demand cutset inequalities (29) reduce to

$$Y^1(S) \geq \left\lceil \frac{B(S)}{C} \right\rceil \quad \forall S \subset V \quad (33)$$

which ensures that the total capacity across a cut is sufficient to support the total demand between all terminal pairs whose end points are on different shores of the cut. Moreover, the residual capacity inequalities are

$$\sum_{s \in S \cap W} \frac{b_s}{C} (1 - \lambda_s^e) \geq \left(\frac{b(S)}{C} - \left\lfloor \frac{b(S)}{C} \right\rfloor \right) \left(\left\lceil \frac{b(S)}{C} \right\rceil - y_e \right) \quad \forall S \subset V, e \in E \quad (34)$$

whose separation can be done in $O(|W||E|)$ time using Algorithm 1 ([5]). Notice that the inequalities (30) and (32) are identical for the single facility case. Hence we implement an exact separation algorithm for the residual capacity inequalities (30).

The first set of results are shown in Table 1 where we compare our B&C algorithm with solving the single facility NLP_{GD}^{hose} using Cplex. At this stage, we include the demand cutset inequalities (33) and the arc residual capacity inequalities (34) together with the feasibility cuts (19) in the B&C algorithm. However, we will also provide an analysis of the effect of each family of valid inequalities on the solution quality later in this section.

¹Then the number of commodities is $|W|(|W| - 1)$.

Instance	(V,E,W,C)	z_{cp}	$t_{cp}(Ig)$	$\#_{cp}$	$z_{B\&C}$	$t_{B\&C}(Ig)$	$\#_{B\&C}$
metro	(11,42,5,24)	818	31	1092	818	19.97	913
nsflb	(14,21,10,24)	97,100	103.5	504	97,100	3.68	21
at-cep1	(15,22,6,24)	51,745	2.84	70	51,745	0.39	5
pacbell	(15,21,7,24)	11,390	12.46	304	11,390	1.76	16
bhv6c	(27,39,15,24)	840,251	110.56	4949	840,251*	(1.22%)	37,702
bhvdc	(29,36,13,24)	1.1e+6	1098.44	590	1.1e+6	3.6	3
pdh	(11,34,6,480)	1.1e+6	11.39	78	1.1e+6	0.22	1
polska	(12,18,12,155)	44,253*	(1.16%)	10,793	44,287*	(0.42%)	46,133
polska	(12,18,12,100)	7478	3713	8056	7478	3094.19	18,719
dfn	(11,47,11,155)	52,380*	(6.34%)	1159	52,416*	(3.74%)	6624
newyork	(16,49,16,1000)	1.499e+6*	(56.74%)	71	1.497e+6*	(45.4%)	21
france	(25,45,14,2500)	17,400*	(3.61%)	525	17,400*	(5.45%)	304
ny-cep2	(16,49,9,24)	7079.2*	(3.8%)	2741	7094.4*	(2.52%)	7998
atlanta	(15,22,15,1000)	4.7e+8*	(0.2%)	9594	4.72e+8*	(0.54%)	19,135
tai	(24,51,19,504k)	1.6e+12*	(99.7%)	52	3.6e+7*	(19.94%)	13
janos	(26,42,26,64)	1.29e+9*	(99.7%)	30	2.7e+6	884.31	3
nobel-eu	(28,41,28,20)	1.47e+10*	(99.7%)	25	4.3e+6*	(2.01%)	8
sun	(27,51,24,40)	6.3e+10*	(99.7%)	1	1533.34*	(20.5%)	6

Table 1: Results for the single facility problem

Out of 18 instances, Cplex and B&C could both solve 8 to optimality in 2 hours. In *bhv6c*, B&C could find the optimal solution with the rounding method but since it could not improve the lower bound, it kept on branching till the end of the time limit. Moreover, even though Cplex gives better upper bounds than B&C in *dfn*, *ny-cep2*, and *atlanta*, the gaps at termination are better for the B&C algorithm in the first two of these instances. On the other hand, B&C is clearly superior for the instances *tai*, *newyork*, *janos*, *nobel-eu*, and *sun*. Actually, the most important point to be highlighted is the significant degradation in the performance of Cplex relative to the B&C algorithm as the network size increases. The last four instances on Table 1 are very good examples of this behavior. Except *tai*, all of the nodes are demand nodes in these instances, i.e., $W = V$, and we observe that among such cases only in *dfn* and *atlanta* Cplex has performed slightly better than B&C. Actually the upper bound of Cplex is just 0.07% and 0.2% smaller than the one of B&C in *dfn* and *atlanta*, respectively. On the other hand the upper bounds we obtain with B&C are 100% better than the bounds with Cplex in *tai*, *janos*, *nobel-eu*, and *sun*. Finally, a comparison of the gaps at termination shows that the B&C algorithm is clearly superior in 8 of the 11 instances with much lower gaps for the *tai*, *nobel-eu*, and *sun* in addition to the zero gap for the *janos* instance.

An additional analysis we have performed is the comparison of the individual and joint influence of the two types of cuts on the root relaxation solution qualities and the total solution times when they are used together with the feasibility cuts. We investigate the following four cases:

- *F*: use only the feasibility cuts;
- *F+D*: use feasibility cuts and demand cutset inequalities together;
- *F+R*: use feasibility cuts and residual capacity inequalities together;
- *all*: use all three types of cuts together.

For each of these four settings, we report the percentage gap between the optimal value and the lower bound at the root node under the *gap* columns as well as the corresponding solution

times under the t columns in Table 2. We have considered 6 instances, which were solved to optimality in relatively shorter times. Based on the results for these six instances we can say that the impact of demand cutset inequalities both on root gaps and solution times is significant. Moreover, the residual capacity inequalities have also yield reasonable improvements in the root gaps. Although adding residual capacity inequalities together with demand cutset inequalities does not improve the root gaps, it improves the solution times. The root gaps and solution times are improved by 86% and 84.72%, respectively on the average as we use all cuts in our B&C algorithm.

Instance	t_F	gap_F	t_{F+D}	gap_{F+D}	t_{F+R}	gap_{F+R}	t_{all}	gap_{all}
metro	37.34	8.83%	27.72	3%	46.32	7.89%	19.97	3.02%
nsf1b	223.53	1.56%	4.34	0.34%	93.79	1.04%	3.68	0.34%
at-cep1	4.56	4.44%	0.47	0.5%	3.69	2.69%	0.39	0.5%
pacbell	68.10	3.54%	3.65	0.37%	73.59	2.87%	1.76	0.37%
bhvdc	7114.04	1.18%	15.83	0.07%	767.66	0.63%	3.6	0.02%
pdh	0.87	9.06%	0.29	0%	1.35	4.55%	0.22	0.4%

Table 2: Results with different cuts

Finally, Table 3 shows the change in both the solution times and the optimal capacity installation cost when demand uncertainty is added to the model. We believe this gives a better sense of how the original NLP is affected on the average by the inclusion of demand uncertainty. The Δ_z column shows the changes in the total reservation costs as we shift to the robust counterpart from the deterministic NLP and it is observed that the average increase is 17.62%. This figure can be interpreted as the price of robustness the network service provider should be ready to pay. On the other hand, we observe some interesting results for the solution times. When we solve the deterministic NLP and NLP_{GD}^{hose} with Cplex we see that the solution times for the polyhedral case are much shorter for the instances *pacbell*, *nsf1b*, *bhvdc*, and *bhv6c*. We believe, this is due to the nice structure of NLP_{GD} we have obtained in Proposition 2.1. Obviously, this is a very important and particular observation we can make on our models. Normally, the robust optimization problems are expected to be more difficult than their deterministic counterparts, i.e., the price for the flexibility of the robust models is paid by means of additional computational efforts. But for our case, the robust problem is even easier to solve and hence there is no technical reason why one would avoid the advantages of using the hose model.

Instance	t_{det}	z_{det}	t_{hose}	z_{hose}	Δ_z
metro	3.16	699	31	818	14.5%
nsf1b	785.74	70,100	103.04	97,100	27.8%
at-cep1	0.75	43,110	2.84	51,745	16.7%
pacbell	53.44	8660	12.46	11,390	24%
bhv6c	558.59	783,376	110.6	840,251	6.8%
bhvdc	3559.4	851,889	1098.44	1,095,622	22.2%
pdh	3.17	1,012,673	11.39	1,142,081	11.43%

Table 3: Deterministic vs polyhedral single facility NLP

7.2 Two Facility NLP_{hose}

In the two facility case, we distinguish between the facility types according to their transmission capacities and refer to them as low capacity facility (*LCF*) and high capacity facility (*HCF*). We

assume that the capacities of each *LCF* and *HCF* are C^1 and C^2 units, respectively. Naturally, the cost of installing each facility is different and economies of scale prevails, i.e., the cost of $\lceil \frac{C^2}{C^1} \rceil$ *LCF*'s is more than the cost of 1 *HCF*. Then, for $S \subset V$, the demand cutset inequalities (29) reduce to the following inequalities:

- *The LCF case*, i.e., $l^* = 1$, where the resulting inequalities are in either of the following forms:

$$\begin{cases} R^1(S)Y^1(S) + (R^1(S) \lceil \frac{C^2}{C^1} \rceil + \\ \min\{f(2, 1), R^1(S)\})Y^2(S) \geq R^1(S) \lceil \frac{B(S)}{C^1} \rceil & \text{if } C^1, C^2 < B(S) \\ Y^1(S) + Y^2(S) \geq 1 & \text{if } C^1, C^2 \geq B(S) \\ Y^1(S) + \lceil \frac{B(S)}{C^1} \rceil Y^2(S) \geq \lceil \frac{B(S)}{C^1} \rceil & \text{if } C^1 < B(S) \text{ and } C^2 \geq B(S) \end{cases}$$

- *The HCF case*, i.e., $l^* = 2$ where we can have:

$$\begin{cases} \min\{C^1, R^2(S)\}Y^1(S) + R^2(S)Y^2(S) \geq R^2(S) \lceil \frac{B(S)}{C^2} \rceil & \text{if } C^1, C^2 < B(S) \\ Y^1(S) + Y^2(S) \geq 1 & \text{if } C^1, C^2 \geq B(S) \\ C^1Y^1(S) + \lceil B(S) \rceil Y^2(S) \geq \lceil B(S) \rceil & \text{if } C^1 < B(S) \text{ and } C^2 \geq B(S). \end{cases}$$

On the other hand, the two types of residual capacity inequalities (30) for each edge $e \in E$ are and set for $S \subset V$

$$\begin{cases} r^1(S)y_e^1 + (r^1(S) \lceil \frac{C^2}{C^1} \rceil + \min\{f(2, 1), r^1(S)\})y_e^2 - \\ \sum_{s \in S \cap W} b_s \lambda_s^e \geq r^1(S) \lceil \frac{b(S)}{C^1} \rceil - b(S) & \text{for } l^* = 1 \\ \min\{C^1, r^2(S)\}y_e^1 + r^2(S)y_e^2 - \sum_{s \in S \cap W} b_s \lambda_s^e \geq r^2(S) \lceil \frac{b(S)}{C^2} \rceil - b(S) & \text{for } l^* = 2. \end{cases}$$

Notice that the number of residual capacity and demand cutset inequalities are doubled as we move from the single facility to the two facility case. Hence, the LP models we solve at each iteration of the B&C algorithm can get large more quickly. Thereof, we have tried the following 5 schemes for adding violated cuts:

- *HA*: add only HCF type inequalities in all nodes of the B&C tree,
- *HR*: add only HCF type inequalities only at the root node,
- *GHA*: add HCF type inequalities gradually, i.e., add a violated HCF residual capacity inequality only if no HCF demand cutset inequality is violated, in all nodes of the B&C tree,
- *GHR*: gradually add HCF type inequalities, i.e., add a violated HCF residual capacity inequality only if no HCF demand cutset inequality is violated, only at the root node of the B&C tree,
- *GAR*: gradually add all valid inequalities, i.e. add violated LCF and HCF residual inequalities only if no LCF or HCF demand cutset inequality is violated, at the root node.

These 5 different alternatives yield very similar results where we observe a slight superiority of *GHR* and *GHA*. Thus, we will provide the results for *GHA*, which we observe to be a little better.

An initial comparison of our B&C algorithm and Cplex can be made using the results in Table 4. The performance of our B&C algorithm is better than that of Cplex especially for the large instances where all nodes are demand nodes just like the single facility case. This is quite obvious especially for the 4 instances *tai*, *nobel-eu*, *pioro*, and *cost266* since the MIP solver could not find even a feasible solution by solving the two facility NLP_{GD}^{hose} problem in 2

Instance	(V, E, W, C^1, C^2)	z_{cp}	t_{cp}	$\#_{cp}$	$z_{B\&C}$	$t_{B\&C}$	$\#_{B\&C}$
metro	(11, 42, 5, 1, 24)	772	2.82	45	772	1.12	55
nsflb	(14, 21, 10, 1, 24)	96,035	2.82	33	96,035	1.86	9
at-cep1	(15, 22, 6, 1, 24)	50,968	0.88	9	50,968	0.49	11
pacbell	(15, 21, 7, 1, 24)	11,177	4.33	100	11,177	54.14	1301
bhv6c	(27, 39, 15, 1, 24)	826,136	1.94	16	829,548*	(0.5%)	36,108
bhvdc	(29, 36, 13, 1, 24)	1.09e+6	30.75	16	1.09e+6	16.12	43
pdh	(11, 34, 6, 30, 480)	8.18e+6	7.17	49	8.18e+6	3.1	87
polska	(12, 18, 12, 155, 622)	34,006*	(2.18%)	1795	34,348*	(3.11%)	50,403
dfn	(11, 47, 11, 155, 622)	44,352*	(21.37%)	896	45,316*	(12.43%)	21,736
newyork	(16, 49, 16, 1k, 4k)	NoI	MA-INF	2	1.6e+6*	(70.4%)	30
atlanta	(15, 22, 15, 1k, 4k)	1.376e+8*	(1.15%)	4326	1.383e+8*	(1.44%)	14,598
tai	(24, 51, 19, 504k, 1008k)	NoI	INF	127	3.06e+7*	(10.48%)	8702
nobel-eu	(28, 41, 28, 20, 40)	NoI	INF	50	2.66e+6*	(1.6%)	690
pioro	(40, 89, 40, 155, 622)	NoI	INF	2	1.38e+6	(2.47%)	2
norway	(27, 51, 27, 1k, 4k)	NoI	INF	2	NoI	INF	1
cost266	(37, 57, 37, 7560, 30240)	NoI	INF	2	2.42e+7*	(29.75%)	2
gui39	(39, 86, 39, 160, 320)	NoI	INF	2	NoI	INF	1

Table 4: Gradual inclusion of HCF valid inequalities at each iteration, i.e., the *GHA* setting.

hours whereas the B&C algorithm successfully produced some upper bounds. Especially, the upper bounds for *nobel-eu* and *pioro* are quite promising. Moreover, the NLP_{GD}^{hose} problem could not be solved for *newyork* instance due to insufficient memory. On the other hand, for those instances that could be solved optimally within the 2 hours time limit, we observe that B&C has worse solution times only for *pacbell* and *bhv6c*. In two cases, i.e., *norway* and *gui*, we could not find any upper bound with either of the methods². On the other hand, the B&C algorithm is better in 6 of the remaining 9 instances with much lower gaps for *dfn*, *tai*, *nobel-eu*, *pioro*, and *cost266*.

In addition, we compare the performance of the five settings in terms of the gaps at termination as shown in Figure 3. The instances for which the B&C algorithm could not find a feasible solution within the 2 hours time limit are assigned a 105% gap. Furthermore, we leave the *bhv6c* instance out of this analysis since all schemes stopped with the same gap. Consequently, we see that the average gaps at termination for these 11 instances are 32.6%, 38.5%, 31.1%, 31.2%, and 56.9% for *HA*, *HR*, *GHA*, *GHR*, and *GAR*, respectively. On the other hand, the average number of nodes in the B&C tree for these five settings are 13968, 11769, 7869, 8903, and 8629. An important point to note here is that the number of nodes is 1 for those instances terminated with no feasible solution. So, even though the highest number of such cases are observed for *GAR*, the size of the B&C tree is smaller for *GHA* on the average.

A final analysis in Table 5 is about the price of robustness measured in terms of the final design cost and solution time for the two facility case. The average increase in the optimal reservation costs of the 6 instances for which gaps could be calculated is 18.86%. *pdh* instance is not included here since the deterministic case could not be solved within 2 hours although the solution time for its robust counterpart is just 3 seconds. Similarly, the difference between the solution times is also obvious for the instances *nsflb*, *bhvdc*, and *bhv6c*. Hence, we can say that the nice structure we have obtained in Proposition 2.1, is again in effect for the two facility problem.

²As we show in Figure 3, we could get bounds for these two instances in other cut schemes *HA* and *HR*.

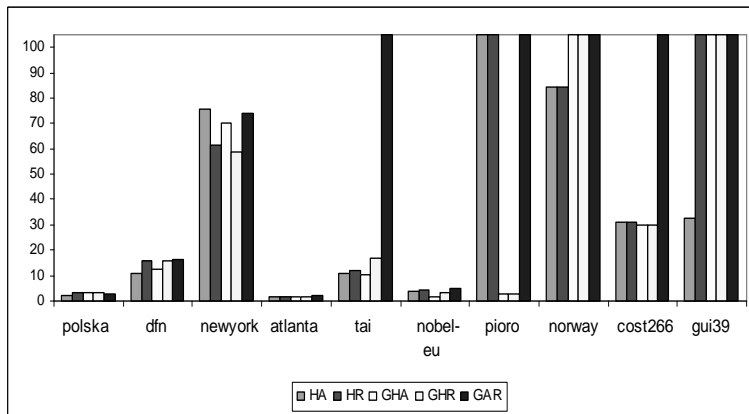


Figure 3: Percent gaps at termination for each scheme

Instance	t_{det}	z_{det}	t_{pol}	z_{pol}	Δ_z
metro	0.86	662	1.12	772	14.2%
nsflb	7.14	69,216	1.86	96,035	27.9%
at-cep1	0.74	42,578	0.49	50,968	16.5%
pacbell	2.47	8354	4.33	11,177	25.3%
bhv6c	16.4	770,185	1.94	826,136	6.8%
bhvdc	204.49	843,206	16.12	1,088,422	22.5%

Table 5: Deterministic vs polyhedral two facility NLP

8 Conclusion

In this paper we studied the Network Loading Problem where the pairwise traffic demands are not assumed to be known in advance. We used a polyhedral definition of traffic demands and sought to design a network which is capable to support infinitely many demand realizations. Based on a compact formulation and a decomposition property we gave a detailed polyhedral analysis for a specific demand uncertainty description, the hose model. The polyhedral analysis formed the basis of an efficient Branch-and-Cut algorithm equipped with a heuristic for computing upper bounds. Our computational results revealed that projecting out the flow variables and the use of Branch-and-Cut algorithm is quite effective for both single and two-facility problem types. An important question is whether similar developments can be expected for uncertainty polyhedron descriptions other than the hose model. We will answer this question in subsequent papers.

Acknowledgments

The authors would like to thank A. Ridha Mahjoub and Hervé Kerivin for helpful discussions.

References

- [1] A. Altın, A. Amaldi, P. Belotti, and M. Ç. Pınar, Provisioning virtual private networks under traffic uncertainty, *Networks*, 2007, 20, 1, pp. 100-115.
- [2] A. Altın, P. Belotti, and M.Ç. Pınar, OSPF routing with optimal oblivious performance ratio under polyhedral demand uncertainty, *Technical report*, Bilkent University, 2006.

- [3] A. Atamtürk, Strong reformulations of robust mixed 0-1 programming, *Mathematical Programming*, 2006, 108, pp. 235-250.
- [4] A. Atamtürk and O. Günlük, Network design arc set with variable upper bounds, *Networks*, to appear.
- [5] A. Atamtürk and D. Rajan, On splittable and unsplittable capacitated network design arc-set polyhedra. *Mathematical Programming*, 2002, 92, pp. 315-333.
- [6] A. Atamtürk and M. Zhang, Two-stage robust network flow and design under demand uncertainty, *Operations Research*, to appear.
- [7] P. Avella, S. Mattia, and A. Sassano, Metric inequalities and the network loading problem. *Discrete Optimization*, 2007, 4, pp. 103-114.
- [8] W. Ben-Ameur and H. Kerivin, Routing of uncertain demands, *Optimization and Engineering*, 2005, 3, pp. 283-313.
- [9] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski, Adjustable robust solutions of uncertain linear programs, *Mathematical Programming*, 2004, Ser. A 99, pp. 351-376.
- [10] A. Ben-Tal and A. Nemirovski, Robust convex optimization, *Mathematics of Operations Research*, 1998, 23(4), pp. 769-805.
- [11] A. Ben-Tal and A. Nemirovski, Robust solutions of uncertain linear programs, *Operations Research Letters*, 1999, 25, pp. 1-13.
- [12] A. Ben-Tal and A. Nemirovski, Selected topics in robust convex optimization, *Mathematical Programming*, to appear.
- [13] D. Bertsimas and M. Sim, Robust discrete optimization and network flows, *Mathematical Programming*, 2003, B 98, pp. 49-71.
- [14] D. Bertsimas and M. Sim, The price of robustness, *Operations Research*, 2004, 52, pp. 35-53.
- [15] D. Berger, B. Gendron, J. Potvin, S. Raghavan, and P. Soriano, Tabu search for a network loading problem with multiple facilities, *Journal of Heuristics*, 2000, 6, pp. 253-267.
- [16] D. Bienstock, S. Chopra, O. Günlük, and C.T. Tsai, Minimum cost capacity installation for multicommodity network flows, *Mathematical Programming*, 1998, 81, pp. 177-199.
- [17] D. Bienstock and O. Günlük, Capacitated network design - polyhedral structure and computation, *INFORMS Journal on Computing*, 1996, 8, pp. 243-259.
- [18] N. Duffield, P. Goyal, A. Greenberg, P. Mishra, K. Ramakrishnan, and J.E. van der Merive, A flexible model for resource management in virtual private networks, *Proceedings of ACM SIGCOMM*, 1999, pp. 95-108.
- [19] A. Gupta, J. Kleinberg, A. Kumar, R. Ratogi, and B. Yener, Provisioning a virtual private network : A network design problem for multicommodity flow, *Proc ACM Symposium on Theory of Computing (STOC)*, Crete, Greece, 2001, pp. 389-398.
- [20] O. Günlük, A Branch-and-Cut algorithm for capacitated network design problems, *Mathematical Programming*, 1999, 86, pp. 17-39.
- [21] O. Günlük, B. Brochmuller, and L. Wolsey, Designing private line networks - polyhedral analysis and computation, *Transactions on Operational Research*, 2004, 16, pp. 7-24.

- [22] S. P.M. van Hoesel, A. M.C.A. Koster, R. L.M.J. van de Leensel, and M. W.P. Savelsbergh, Polyhedral results for the edge capacity polytope, *Mathematical Programming*, 2002, Ser. A 92, pp. 335-358.
- [23] G. Italiano, R. Rastogi, and B. Yener, Restoration algorithms for virtual private networks in the hose model, *IEEE INFOCOM*, 2002, pp. 131-139.
- [24] O. Kardeşan, M. Ç. Pınar, and H. Yaman, Robust DWDM routing and provisioning under polyhedral demand uncertainty, *Technical report*, Bilkent University, 2006,.
- [25] A. Kumar, R. Rastogi, A. Silberschatz, and B. Yener, Algorithms for provisioning virtual private networks in the hose model, *SIGCOMM'01*, August 27-31, 2001, San Diego, California, USA.
- [26] M. Labbé and H. Yaman, Solving the hub location problem in a star-star network, *Networks*, to appear.
- [27] T. L. Magnanti and P. Mirchandani, Shortest paths single origin-destination network design, and associated polyhedra, *Networks*, 1993, Vol. 23, pp. 103-121.
- [28] T. L. Magnanti, P. Mirchandani, and R. Vachani, The convex hull of two core capacitated network design problems, *Mathematical Programming*, 1993, 60, pp. 233-250.
- [29] T. L. Magnanti, P. Mirchandani, and R. Vachani, Modeling and solving the two-facility capacitated network loading problem, *Operations Research*, 1995, 43(1), pp. 142-157.
- [30] P. Mirchandani, Projections of the network loading problem, *European Journal of Operational Research*, 2000, 122, pp. 534-560.
- [31] S. Mudchanatongsuk, F. Ordoñez, and J. Liu, Robust solutions for network design under transportation cost and demand uncertainty, *Technical Report*, University of Southern California, 2005.
- [32] K. Onaga and O. Kakusho, On feasibility conditions of multicommodity flows in Networks, *IEEE Transactions on Circuit Theory*, 1971, 18(4), pp. 425-429.
- [33] F. Ordoñez and J. Zhao, Robust capacity expansion of network flows, *Technical Report*, University of Southern California, 2004.
- [34] R.L. Rardin and L.A. Wolsey, Valid inequalities and projecting the multicommodity extended formulation for uncapacitated fixed charge network flow problems, *European Journal of Operational Research*, 1993, 71, pp. 95-109.
- [35] M.W.P. Savelsbergh, G.C. Sigismondi, G.L. Nemhauser. A functional description of MINTO, a Mixed INTEger Optimizer. *OR Letters*, 1994, 15, pp. 47-58.
- [36] <http://sndlib.zib.de/home.action>
- [37] A.L. Soyster, Convex programming with set-inclusive constraints and applications to inexact linear programming, *Operations Research*, 1973, 21, pp. 1154-1157.
- [38] L. A. Wolsey, *Integer Programming*, Wiley-Interscience, NY, 1998.
- [39] H. Yaman, The integer knapsack cover polyhedron, *SIAM Journal on Discrete Mathematics*, to appear.
- [40] H. Yaman, O.E. Kardeşan, and M.Ç. Pınar, Restricted robust uniform matroid maximization under interval uncertainty, *Mathematical Programming*, 2007, 110(2), pp. 431-441.