BCOL RESEARCH REPORT 15.02

Industrial Engineering & Operations Research
University of California, Berkeley, CA 94720-1777

Forthcoming in *Networks*

Three-partition Flow Cover Inequalities for Constant Capacity Fixed-charge Network Flow Problems

Alper Atamtürk, Andrés Gómez

Department of Industrial Engineering & Operations Research, University of California, Berkeley, CA 94720

atamturk@berkeley.edu, a.gomez@berkeley.edu

Simge Küçükyavuz

Department of Integrated Systems Engineering, Ohio State University, Columbus, OH 43210

kucukyavuz.2@osu.edu

April 2015; February 2016

Abstract

Flow cover inequalities are among the most effective valid inequalities for capacitated fixed-charge network flow problems. These valid inequalities are based on implications for the flow quantity on the cut arcs of a two-partitioning of the network, depending on whether some of the cut arcs are open or closed. As the implications are only on the cut arcs, flow cover inequalities can be obtained by collapsing a subset of nodes into a single node. In this paper we derive new valid inequalities for the capacitated fixed-charge network flow problem by exploiting additional information from the network. In particular, the new inequalities are based on a three-partitioning of the nodes. The new three-partition flow cover inequalities include the flow cover inequalities as a special case. We discuss the constant capacity case and give a polynomial separation algorithm for the inequalities. Finally, we report computational results with the new inequalities for networks with different characteristics.

Keywords: integer programming, lifting, superadditivity, fixed-charge network flow, three-partition, facets.

1 Introduction

Many logistics, supply chain, and telecommunications problems are modeled as *capacitated fixed-charge network flow* problems (CFNF). A CFNF is defined on a directed graph, with given supply or demand on the nodes of the graph, and capacity, fixed and variable costs of flow on the arcs of the graph. The problem is to choose a subset of the arcs and route the flow on the chosen arcs while satisfying the supply, demand and capacity constraints, so that the sum of fixed and variable costs is minimized.

Given a digraph $G = (V, A)$, let y_a be the flow on arc $a \in A$ and let $x_a = 1$ if the arc a is used, 0 otherwise. Then, the CFNF problem can be formulated as

$$\begin{aligned}
\min \ & \sum_{a \in A} (p_a y_a + q_a x_a) & & \\
\text{s.t.} \ & \sum_{a \in \delta(i)^+} y_a - \sum_{a \in \delta(i)^-} y_a = d_i, & \forall i \in V & \quad (1) \\
& y_a \le c_a x_a, & \forall a \in A & \quad (2) \\
& x \in \mathbb{B}^n, y \in \mathbb{R}^n_+, & &
\end{aligned}$$

where $\delta(i)^+$ is the set of incoming arcs to $i \in V$, $\delta(i)^-$ is the set of outgoing arcs, $\sum_{i \in V} d_i = 0$, and p_a and q_a are the variable and fixed cost of flow on arc $a \in A$, respectively. The flow conservation constraints (1) ensure that the demand is met at node i (when $d_i > 0$) and supply is not exceeded (when $d_i < 0$), and the upper bound constraints (2) ensure that the arc capacity c_a, $a \in A$, is not exceeded.

Almost all previous work on valid inequalities for CFNF is based on cut-set relaxations. Given a subgraph, a cut refers to the arcs that have one end in the subgraph, the other outside. A cut set refers to the relaxation of the feasible set of CFNF given by the constraints for the cut arcs and the aggregation of the supply/demand constraints for the nodes in the subgraph defining the cut. Simply, valid inequalities based on a cut set are implications for the flow quantity on the cut arcs depending on whether some of these arcs are open or closed. The flow cover inequalities (e.g., [5, 10, 11, 13]) and the flow pack inequalities described in [2] belong to this category of inequalities. Additionally, Gu et al. [7] show how to strengthen the existing inequalities using superadditive lifting. These cut-set inequalities are based on a two-partitioning of the network and ignore the network structure in each of the partitions, essentially collapsing the corresponding subgraphs into a single node.

Valid inequalities that consider structures other than a two-partitioning of the network have received less attention in the literature. Path inequalities are studied in [3] and [12]. The submodular inequalities given in [15] are very general, but they are not given in closed form since the coefficients of the inequalities are expressed as solutions to optimization problems. Valid inequalities based on a three-partitioning of the network have been studied in the constant capacity network design problem (e.g., [1, 4, 8, 9]), but they have not been considered for CFNF problems.

In this work we consider three-partition relaxations for the constant capacity CFNF (i.e., $c_a = c$ for all $a \in A$). We present new valid inequalities for this problem, which include as a special case the flow cover inequalities. We discuss their separation and implementation as cutting planes to solve the constant capacity CFNF, and we present computational experiments in networks with different characteristics.

The paper is organized as follows. In Section 2 we review the flow cover inequalities for the constant capacity CFNF. In Section 3 we study the three-partition relaxation and provide valid inequalities for this problem. In Section 4 we discuss the implementation and separation problem for the three-partition flow cover inequalities. In Section 5 we give computational results, and in Section 6 we conclude the paper. Throughout the paper, for a vector $v \in \mathbb{R}^n$, let $v(S) = \sum_{i \in S} v_i$, $i^+ = \max\{0, i\}$ and $i^- = \min\{i, 0\}$.

2 Flow cover inequalities

Given disjoint sets N^+ and N^-, let $N := N^+ \cup N^-$, $n := |N|$ and $c > 0$, let

$$X := \left\{ (x, y) \in \mathbb{B}^n \times \mathbb{R}^n_+ : y(N^+) - y(N^-) = d,\ y_j \leq c x_j,\ j \in N \right\},$$

and let $X_{\leq}$ be the relaxation where the equality is replaced with an inequality. The set X arises in CFNF when several nodes are aggregated into one (i.e., adding the corresponding flow conservation constraints), implicitly defining a two-partitioning of the network. For a subset of vertices $V' \subseteq V$, $d = d(V')$ is the aggregate demand on the vertices of V', $N^+ = \{(i,j) \in E : i \in V \setminus V',\ j \in V'\}$ is the set of incoming arcs to V', $N^- = \{(i,j) \in E : i \in V',\ j \in V \setminus V'\}$ is the set of outgoing arcs from V' and c is the (common) capacity of the arcs. Figure 1 depicts a typical two-partitioning of a network.

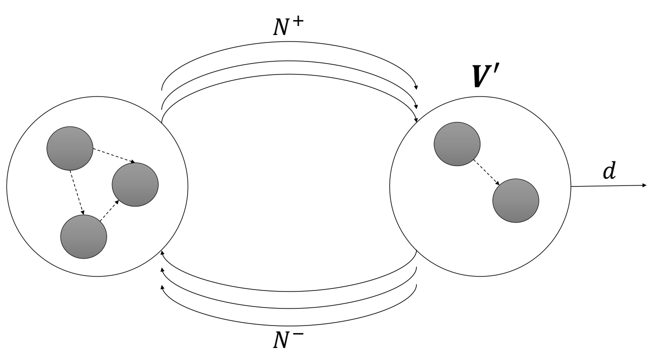

Figure 1: Two-partitioning of the network.

Padberg et al. [10] studied the convex hull of X and $X_{\leq}$. A set $S^+ \subseteq N^+$ is called a *flow cover* if $\lambda = c(S^+) - d > 0$. Moreover, when $\lambda < c$ we say that the flow cover is *minimal.* Note that $\lambda = \lceil \frac{d}{c} \rceil c - d$ in a minimal flow cover. Given a flow cover S^+, the *flow cover inequality*

$$\sum_{j \in S^+} (y_j + \rho(1 - x_j)) - \sum_{j \in N^-} \min\{y_j, (c - \rho)x_j\} \leq d, \tag{3}$$

where $\rho = (c - \lambda)^+$, is valid for $X_\leq$ and X. Moreover, the *lifted flow cover inequality*

$$\sum_{j \in S^+} (y_j + \rho(1 - x_j)) - \sum_{j \in N^-} \min\{y_j, (c - \rho)x_j\} + \sum_{j \in N^+ \setminus S^+} \max\{y_j - \rho x_j, 0\} \leq d \tag{4}$$

is also valid for $X_\leq$ and X. Moreover, inequalities (4) together with the flow conservation and bound constraints completely describe the convex hull of X. When applying inequalities (3)–(4) to an aggregation of many nodes collapsed into one, only the arcs between the two partitions are considered in the inequality, and the internal network structure of each of the partitions is not taken into account in the inequality.

Note that optimizing a linear function over X is easy.

Proposition 1. *There is an $O(n)$ algorithm to solve the optimization problem*

$$\min\{py + qx : (x, y) \in X\}. \tag{5}$$

Proof. By complementing variables in N^-, that is, defining $\bar{x}_i = 1 - x_i$ and $\bar{y}_i = c - y_i$, $i \in N^-$, we write the equivalent problem

$$\begin{aligned} cp(N^-) + q(N^-) + \min_{(x,y) \in \mathbb{B}^n \times \mathbb{R}^n} \; & \sum_{i \in N^+} (p_i y_i + q_i x_i) - \sum_{i \in N^-} (p_i \bar{y}_i + q_i \bar{x}_i) \\ \text{s.t.} \; & y(N^+) + \bar{y}(N^-) = d + c(N^-) \\ & 0 \leq y_i \leq c x_i, && \forall i \in N^+ \\ & c\bar{x}_i \leq \bar{y}_i \leq c, && \forall i \in N^-. \end{aligned}$$

Observe that if $q_i < 0$ for $i \in N^+$, then $x_i = 1$ in any optimal solution, and if $q_i < 0$ for $i \in N^-$, then $\bar{x}_i = 0$ in any optimal solution. Therefore, we assume without loss of generality that $q \geq 0$. Denote by ξ_i the contribution to the objective function of arc $i \in N$ if it is at full capacity, i.e., $\xi_i = cp_i + q_i$ for $i \in N^+$ and $\xi_i = -cp_i - q_i$ for $i \in N^-$. Assume N is sorted in nondecreasing order of ξ_i's, and let N_0 denote the first $\left\lfloor \frac{d + c(N^-)}{c} \right\rfloor$ elements of N in this order.

If $\lambda = 0$, then N_0 gives the optimal solution. Otherwise, note that there exists an optimal solution where there is only a single index k with corresponding flow equal to $c - \lambda$, where $\lambda = \lceil \frac{d}{c} \rceil c - d$, and $\left\lfloor \frac{d + c(N^-)}{c} \right\rfloor$ other arcs are at full capacity. There are two possibilities:

- $k \notin N_0$: Then set all arcs in N_0 at full capacity, and let k be the best choice between $\arg\min_{i \in N^+ \setminus N_0} (c - \lambda)p_i + q_i$ and $\arg\min_{i \in N^- \setminus N_0} -(c - \lambda)p_i$.
- $k \in N_0$: Then set the first $\left\lceil \frac{d + c(N^-)}{c} \right\rceil$ elements of N at full capacity except for k, which is the best choice between $\arg\min_{i \in N^+ \cap N_0} -\lambda p_i$ and $\arg\min_{i \in N^- \cap N_0} \lambda p_i + q_i$.

Note that N need not be sorted; we only require the first $\left\lfloor \frac{d+c(N^-)}{c} \right\rfloor$ arcs (in any order), which can be done in $O(n)$ time with quickselect. Therefore, the complexity of the algorithm is $O(n)$. □

We now show how to update an optimal solution after a small change in d. We will use this result later in Section 3.

Remark 1. Suppose we have an optimal solution to problem (5), and the demand is changed by $\pm c$. We can compute a new optimal solution from the previous one by doing exactly one of the following:

- Add one arc at full capacity.
- Remove one arc at full capacity.
- Complete the flow on the non-saturated arc, and increase the flow to $c - \lambda$ in one of the arcs with no flow.
- Make the flow on the non-saturated arc 0, and decrease the flow to $c - \lambda$ in one of the saturated arcs.

If we keep two sorted lists, one with the contributions of the arcs at full capacity and one with the contributions of the arcs at partial capacity, then these operations are done in $O(1)$ time.

3 Three-partition analysis

3.1 Preliminaries

In this section we study the constant capacity three-partition polytope T. Figure 2 shows a graphical representation of T, where node 0 (implicit) is a supply node and nodes 1 and 2 are demand nodes, and each node represents a partition in the original graph. Let $N_i^+, i = 1, 2$, be the set of arcs going from node 0 to node i, let $N_i^-, i = 1, 2$, be the set of arcs going from node i to node 0, and let N_{12} and N_{21} be the sets of arcs going from node 1 to node 2, and from node 2 to node 1, respectively. Define $N^+ = N_1^+ \cup N_2^+$, $N^- = N_1^- \cup N_2^-$, $N = N^+ \cup N^- \cup N_{12} \cup N_{21}$ and $n = |N|$. Finally, let $d_i, i = 1, 2$, be the demand at node i, and let $d_{12} = d_1 + d_2$. T is defined as the convex hull of

$$y(N_1^+) - y(N_1^-) - y(N_{12}) + y(N_{21}) = d_1 \quad (6)$$

$$y(N_2^+) - y(N_2^-) + y(N_{12}) - y(N_{21}) = d_2 \quad (7)$$

$$y_j \leq cx_j, \quad \forall j \in N$$

$$x \in \mathbb{B}^n, y \in \mathbb{R}_+^n.$$

Moreover let $T_\leq$ be the relaxation of T where equalities in (6) and (7) are replaced by inequalities.

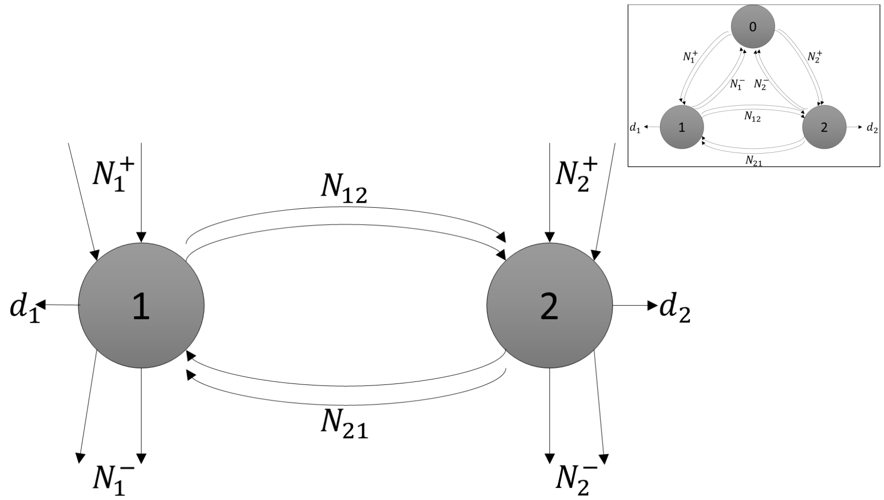

Figure 2: Three-partition graph.

Proposition 2. *For any network, after aggregation, we may assume without loss of generality that* $d_1 \geq 0$ *and* $d_2 \geq 0$ *in equalities* (6)-(7).

Proof. Suppose that in a three-partitioning there are two supply nodes and one demand node. In other words, $d_1 \leq 0$, $d_2 \leq 0$ and $d_0 = -d_1 - d_2$. Letting $\bar{N}_i^+ = N_i^-$, $\bar{N}_i^- = N_i^+$, $i = 1, 2$ and $\bar{N}_{12} = N_{21}$, $\bar{N}_{21} = N_{12}$, and $\bar{d}_i = -d_i$, $i = 1, 2$, we get an equivalent model with a single supply node and two demand nodes. □

By Proposition 2 we assume throughout the rest of the paper that $d_1 \geq 0$ and $d_2 \geq 0$.

Proposition 3. *There is an* $O(n \log n)$ *algorithm to solve the optimization problem*

$$\min\{py + qx : (x, y) \in T\}.$$

Proof. There exists an optimal solution in which at most two arcs have flow strictly between 0 and c. In particular, there exists a pair of nodes such that the flow between them is a multiple of c. Keep sorted lists for $N_1^+ \cup N_1^-$, $N_2^+ \cup N_2^-$ and $N_{12} \cup N_{21}$ as described in Remark 1.

Suppose $y(N_{12}) - y(N_{21}) = kc$, where $k = -|N_{21}|, \ldots, 0, \ldots, |N_{12}|$. Set $k = 0$, and the problem decomposes into two single node constant capacity fixed-charge network flow problems, solvable in $O(n)$ time. Then, changing k one unit at a time, we can compute optimal solutions for all possible values of k, each one in O(1) time (Remark 1). Choose the one that results in the best objective function value.

The cases when $y(N_1^+) - y(N_1^-) = kc$ and $y(N_2^+) - y(N_2^-) = kc$ are handled similarly in $O(n)$ time. The complexity is then given by sorting, which is done in $O(n \log n)$ time.

□

The existence of polynomial time algorithm for optimization over T implies that the separation problem over T is also polynomial time solvable, as shown in [6].

We now characterize strong valid inequalities for T and $T_{\leq}$.

3.2 Valid inequalities

Definition 1 (Three-partition flow cover)**.** For $S_1^+ \subseteq N_1^+$, $S_2^+ \subseteq N_2^+$ and $S_{12} \subseteq N_{12}$, we say that the set $\mathcal{S} = S_1^+ \cup S_2^+ \cup S_{12}$ is a *three-partition flow cover* if

1. $\lambda_1 := c(S_1^+) - d_1 > 0$.
2. $\lambda_2 := c(S_2^+ \cup S_{12}) - d_2 > 0$.
3. $\lambda := c(S_1^+ \cup S_2^+) - d_{12} > 0$.

Furthermore, we say that the three-partition flow cover $\mathcal{S}$ is *minimal* if

4. $\lambda_2 < c$ and $\lambda < c$.

Conditions 1, 2 and 3 imply that the arcs in $\mathcal{S}$ are sufficient to satisfy the demand in nodes 1 and 2. Condition 4 implies that if any arc is removed from $\mathcal{S}$, then the demand can no longer be met. Note that if the cover is minimal, then $\lambda_2 = \lceil \frac{d_2}{c} \rceil c - d_2$ and $\lambda = \lceil \frac{d_{12}}{c} \rceil c - d_{12}$.

Given a minimal three-partition flow cover $\mathcal{S} = (S_1^+, S_2^+, S_{12})$, consider the *three-partition flow cover inequalities*

$$\begin{aligned} &\sum_{i=1,2} \sum_{j \in S_i^+} (y_j + \rho_i(1 - x_j)) - \sum_{i=1,2} \sum_{j \in N_i^-} \min\{y_j, (c - \rho_i)x_j\} \\ &+ \sum_{i=1,2} \sum_{j \in N_i^+ \setminus S_i^+} \max\{y_j - \rho_i x_j, 0\} + \sum_{j \in S_{12}} (\rho_2 - \rho_1)(1 - x_j) \\ &- \sum_{j \in N_{12} \setminus S_{12}} \min\{y_j, (\rho_2 - \rho_1)x_j\} + \sum_{j \in N_{21}} \max\{0, y_j + (\rho_2 - \rho_1 - c)x_j\} \leq d_{12}, \end{aligned} \tag{8}$$

where $(\rho_1, \rho_2) = \begin{cases} (c - \lambda, c - \lambda + (\lambda - \lambda_2)^+)) & \text{(Type 1 three-partition flow cover inequalities)} \\ ((\lambda_2 - \lambda)^+, c - \lambda_2 + (\lambda_2 - \lambda)^+) & \text{(Type 2 three-partition flow cover inequalities).} \end{cases}$

Remark 2. Note that when $\lambda_2 \geq \lambda$, the type 1 three-partition flow cover inequalities reduce to the lifted flow cover inequalities for both nodes collapsed. Moreover, when $\lambda_2 \leq \lambda$, the type 2 three-partition flow cover inequalities are equivalent to the lifted flow cover inequalities for node 2 after adding the flow conservation constraint for node 1. In Section 3.3.2 we give conditions under which inequalities (8) strictly dominate the flow cover inequalities.

Before proving the validity of inequalities (8), we provide examples of both types of three-partition flow cover inequalities.

Example 1 (Type 1 three-partition flow cover inequalities)**.** Consider the network depicted in Figure 3, and assume that the arc capacity is 10. Note that $\lambda_2 = \lceil \frac{15}{10} \rceil 10 - 15 = 5$ and $\lambda = \lceil \frac{22}{10} \rceil 10 - 22 = 8$, and since $\lambda > \lambda_2$ the type 2 three-partition flow cover inequalities reduce to the flow cover inequalities for node 2.

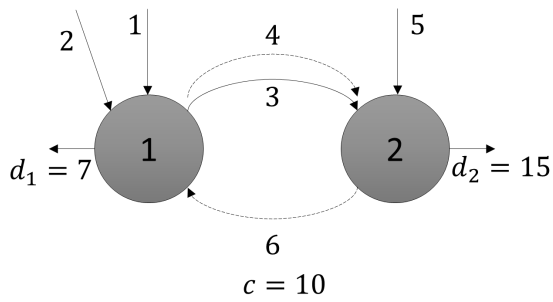

Figure 3: Type 1 three-partition example.

For this network the flow-cover inequality for both nodes aggregated with flow cover $\{1, 2, 5\}$ is

$$y_1 - 2x_1 + y_2 - 2x_2 + y_5 - 2x_5 \leq 16, \tag{9}$$

and the three-partition flow cover inequality with three-partition flow cover $\{1, 2, 3, 5\}$ is

$$y_1 - 2x_1 + y_2 - 2x_2 + y_5 - 5x_5 + 3(1 - x_3) + (y_6 - 7x_6)^+ - \min\{y_4, 3x_4\} \leq 13.$$

Example 2 (Type 2 three-partition flow cover inequalities)**.** Consider the network depicted in Figure 4, and assume that the arc capacity is 10. Note that $\lambda_2 = \lceil \frac{4}{10} \rceil 10 - 4 = 6$ and $\lambda = \lceil \frac{6}{10} \rceil 10 - 6 = 4$, and since $\lambda < \lambda_2$ the type 1 three-partition flow cover inequalities reduce to the flow cover inequalities for both nodes collapsed.

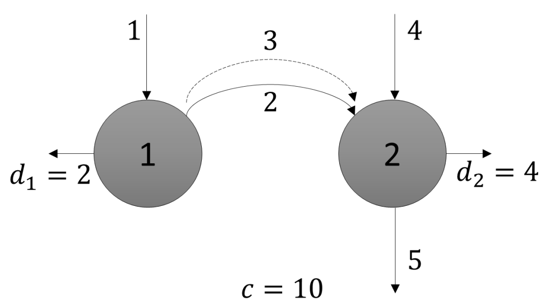

Figure 4: Type 2 three-partition example.

For this network the lifted flow-cover inequality for node 2 with flow cover $\{2\}$ is

$$y_2 - 4x_2 + (y_3 - 4x_3)^+ + (y_4 - 4x_4)^+ - \min\{y_5, 6x_5\} \leq 0,$$

and the three-partition flow cover inequality with three-partition flow cover $\{1, 2\}$ is

$$y_1 - 2x_1 - 4x_2 - \min\{y_3, 4x_3\} + (y_4 - 6x_4)^+ - \min\{y_5, 4x_5\} \leq 0.$$

We now state and prove the main result of the paper.

Theorem 1. *The three-partition flow cover inequalities* (8) *are valid for* T *and* $T_{\leq}$*.*

To prove Theorem 1, we fix some of the variables at their upper or lower bounds in order to apply the lifted flow cover inequalities of Section 2. We then use superadditive lifting to include the variables assumed to be fixed in the inequalities.

We first state two propositions on superadditive functions that are used to prove Theorem 1.

Proposition 4. *[7] The function* $g_{a,c} : \mathbb{R} \to \mathbb{R}$*, where* $0 < a < c$*, given by*

$$g_{a,c}(z) = \begin{cases} ia & \text{if } ic \leq z \leq (i+1)c - a,\ i \in \mathbb{Z} \\ z - i(c-a) & \text{if } ic - a \leq z \leq ic,\ i \in \mathbb{Z} \end{cases}$$

is superadditive on $\mathbb{R}$*.*

Figure 5 depicts function $g_{a,c}(z)$.

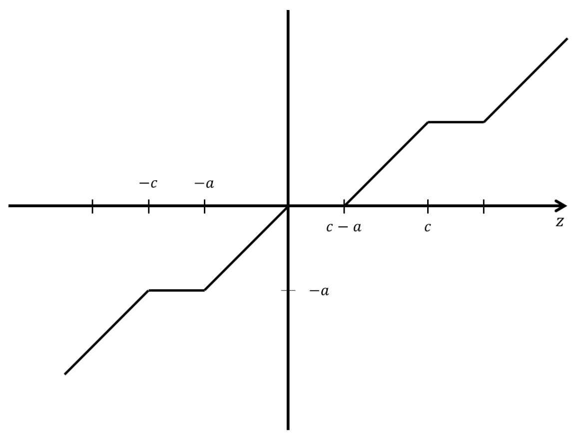

Figure 5: Function g.

Proposition 5. *Let* $g : \mathbb{R} \to \mathbb{R}$ *be a superadditive function and let* $h : \mathbb{R}^2 \to \mathbb{R}$ *be defined as* $h(x, y) = g(x + y)$*. Then, the function* h *is superadditive on* $\mathbb{R}^2$*.*

Proof. For any $(x_1, y_1), (x_2, y_2) \in \mathbb{R}^2$ we have

$$\begin{aligned} h(x_1 + x_2, y_1 + y_2) &= g((x_1 + y_1) + (x_2 + y_2)) \\ &\geq g(x_1 + y_1) + g(x_2 + y_2) \\ &= h(x_1, y_1) + h(x_2, y_2). \end{aligned}$$

□

We now proceed with the proof of Theorem 1.

Validity of type 1 inequalities. We prove the validity of type 1 three-partition flow cover inequalities when $\lambda > \lambda_2$. Let $\mathcal{S}$ be a minimal three-partition flow cover, and assume $y_j = c$ for all $j \in S_{12}$ (i.e., $y(S_{12}) = c(S_{12}) = \lambda_1 + \lambda_2 - \lambda$), $x_j = 0$ for $j \in (N_2^+ \setminus S_2^+) \cup N_2^- \cup (N_{12} \setminus S_{12}) \cup N_{21}$, and $x_j = 1$ for $j \in S_2^+$. Under these conditions the lifted flow cover inequality for nodes 1 and 2 collapsed, and with flow cover $S_1^+ \cup S_2^+$ is

$$y(S_1^+ \cup S_2^+) + \sum_{j \in S_1^+} (c - \lambda)(1 - x_j) + \sum_{j \in N_1^+ \setminus S_1^+} (y_j - (c - \lambda)x_j)^+ - \sum_{j \in N_1^-} \min\{y_j, \lambda x_j\} \leq d_{12}. \quad (10)$$

To obtain a valid inequality for $T_{\leq}$, we lift inequality (10) first with the variables assumed to be fixed in $N_2^+ \cup N_2^-$, and then with the variables in $N_{12} \cup N_{21}$.

The lifting function associated with simultaneously lifting the inequality with the variables x_j for $j \in S_2^+$ and pairs (x_j, y_j) for $j \in (N_2^+ \setminus S_2^+) \cup N_2^-$ is given by

$$\begin{aligned} f_2(z_2, w_2) = \quad & d_{12} + \min \Big\{ -y(S_1^+ \cup S_2^+) - \sum_{j \in S_1^+} (c - \lambda)(1 - x_j) \\ & - \sum_{j \in N_1^+ \setminus S_1^+} (y_j - (c - \lambda)x_j)^+ + \sum_{j \in N_1^-} \min\{y_j, \lambda x_j\} \Big\} \\ \text{s.t. } & y(N_1^+) - y(N_1^-) \leq d_1 + \lambda_1 + \lambda_2 - \lambda \\ & y(S_2^+) \leq d_2 - \lambda_1 - \lambda_2 + \lambda + w_2 \\ & y(S_1^+) \leq d_1 + \lambda_1 \\ & y(S_2^+) \leq d_2 + \lambda - \lambda_1 - z_2 \\ & 0 \leq y_j \leq c x_j, x_j \in \{0, 1\}, \quad j \in N_1^+ \cup N_1^-, \end{aligned}$$

where z_2 is a nonnegative multiple of the capacity c and stands for the capacity closed on arcs in S_2^+, $w_2 > 0$ stands for the flow on arcs in N_2^- and $w_2 < 0$ stands for the flow on arcs in $N_2^+ \setminus S_2^+$.

Note that the problem decomposes for the arcs in $N_1^+ \cup N_1^-$ and the arcs in S_2^+. The objective function is nonincreasing in $y(N_1^+ \cup N_1^-)$, and therefore, there exists an optimal solution where $y(N_1^+) - y(N_1^-) = d_1 + \lambda_1 + \lambda_2 - \lambda$. The optimal choice for parameters is to set $y(N_1^-) = x(N_1^-) = 0$, $y(N_1^+ \setminus S_1^+) = 0$ and $y(S_1^+) = d_1 + \lambda_1 - \lambda + \lambda_2 > c(S_1^+) - \lambda$, so $x_j = 1$ for all $j \in S_1^+$. Similarly, in an optimal solution we have $y(S_2^+) = d_2 + \lambda - \lambda_1 + \min\{-z_2, w_2 - \lambda_2\}$.

Replacing these values in the objective, we get a closed form solution for the lifting function,

$$f_2(z_2, w_2) = -w_2 + \begin{cases} 0 & \text{if } z_2 + w_2 \leq \lambda_2 \\ z_2 + w_2 - \lambda_2 & \text{otherwise.} \end{cases}$$

The exact lifting function f_2 is not superadditive in $\mathbb{R}^2$. Note however that f_2 and the superadditive function $g_{c-\lambda_2,c}$ of Proposition 4 are closely related, as shown by Figure 6. In particular, we show that $\psi(z_2, w_2) := -w_2 + g_{c-\lambda_2,c}(z_2 + w_2)$ is a *superadditive valid lifting function* for f_2.

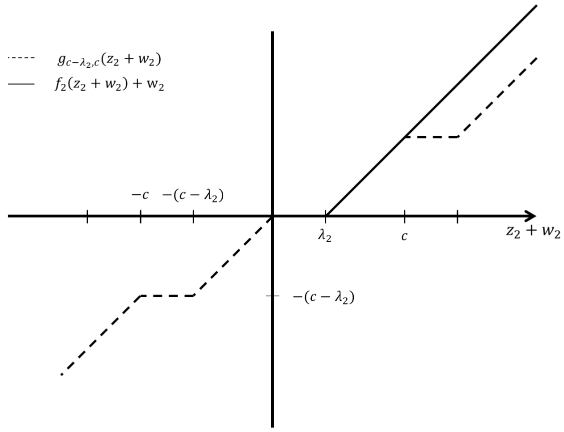

Figure 6: Functions $f_2(z_2+w_2)+w_2$ and $g_{c-\lambda_2,c}(z_2+w_2)$.

Proposition 6. *The function ψ is a superadditive lower bound of f_2.*

Proof. The function $\psi_1(z_2, w_2) = -w_2$ is linear and therefore superadditive. The function $\psi_2(z_2, w_2) = g_{c-\lambda_2,c}(z_2+w_2)$ is superadditive by Proposition 5. We have that $\psi = \psi_1 + \psi_2$ is a sum of superadditive functions, and is therefore superadditive itself.

Moreover, since ψ_2 is a lower bound of $f_2 + w_2$, ψ is a lower bound of f_2.

□

Using the lifting function ψ of Proposition 6, we obtain the inequality

$$\begin{aligned} y(S_1^+ \cup S_2^+) &+ \sum_{j\in S_1^+}(c-\lambda)(1-x_j) + \sum_{j\in N_1^+\setminus S_1^+}(y_j-(c-\lambda)x_j)^+ - \sum_{j\in N_1^-}\min\{y_j,\lambda x_j\} \\ &+ \sum_{j\in S_2^+}(c-\lambda_2)(1-x_j) + \sum_{j\in N_2^+\setminus S_2^+}(y_j-(c-\lambda_2)x_j)^+ - \sum_{j\in N_2^-}\min\{y_j,\lambda_2 x_j\} \le d_{12}. \end{aligned} \tag{11}$$

Inequality (11) assumes that $y_j = c$ for $j \in S_{12}$ and $x_j = 0$ for $j \in (N_{12} \setminus S_{12}) \cup N_{21}$. To obtain a valid inequality for $T_{\le}$, we lift inequality (11) with the variable pairs (x_j, y_j) for $j \in N_{12} \cup N_{21}$. The corresponding lifting function is

$$
\begin{aligned}
f_{12}(w_{12}) = \quad & d_{12} + \min\Big\{ -y(S_1^+ \cup S_2^+) - \sum_{j\in S_1^+}(c-\lambda)(1-x_j) - \sum_{j\in S_2^+}(c-\lambda_2)(1-x_j) \\
& - \sum_{j\in N_1^+\setminus S_1^+}(y_j-(c-\lambda)x_j)^+ - \sum_{j\in N_2^+\setminus S_2^+}(y_j-(c-\lambda_2)x_j)^+ \\
& + \sum_{j\in N_1^-}\min\{y_j,\lambda x_j\} + \sum_{j\in N_2^-}\min\{y_j,\lambda_2 x_j\}\Big\} \\
\text{s.t. } & y(N_1^+) - y(N_1^-) \le d_1 + \lambda_1 + \lambda_2 - \lambda - w_{12} \qquad (12) \\
& y(N_2^+) - y(N_2^-) \le d_2 - \lambda_1 - \lambda_2 + \lambda + w_{12} \qquad (13) \\
& y(S_1^+) \le d_1 + \lambda_1 \\
& y(S_2^+) \le d_2 + \lambda - \lambda_1 \\
& 0 \le y_j \le cx_j, x_j \in \{0,1\}, \quad j \in N_i^+ \cup N_i^-, i = 1,2,
\end{aligned}
$$

where $w_{12} > 0$ stands for the flow on arcs in N_{21} plus the unused capacity on S_{12} (i.e., $y(N_{21}) + \sum_{j\in S_{12}}(c_j - y_j)$) and $w_{12} < 0$ stands for the flow on arcs in $N_{12} \setminus S_{12}$.

The problem decomposes again and since the objective function is nonincreasing in $y(N_i^+) - y(N_i^-)$, $i = 1,2$, we can set these values to their upper bounds, so that constraints (12) and (13) are binding.

Note that for $\lambda_2 - \lambda \le w_{12} \le \lambda_2$, we have $c(S_1^+) - \lambda \le y(N_1^+) - y(N_1^-) \le c(S_1^+)$. Therefore, in this range there exists an optimal solution with $y(S_1^+) = d_1 + \lambda_1 + \lambda_2 - \lambda - w_{12}$ and $y(N_1^- \cup N_1^+ \setminus S_1^+) = 0$. For $w_{12} \ge \lambda_2$ we need to sequentially close arcs in S_1^+ or open arcs in N_1^-, and for $w_{12} \le \lambda_2 - \lambda$ we need to sequentially open arcs in $N_1^+ \setminus S_1^+$.

Moreover, for $0 \le w_{12} \le \lambda_2$, we have $c(S_2^+) - \lambda_2 \le y(N_2^+) - y(N_2^-) \le c(S_2^+)$. Therefore, in this range there exists an optimal solution with $y(S_2^+) = d_2 - \lambda_1 - \lambda_2 + \lambda + w_{12}$ and $y(N_2^- \cup N_2^+ \setminus S_2^+) = 0$. For $w_{12} \le 0$ we need to sequentially close arcs in S_2^+ or open arcs in N_2^-, and for $w_{12} \ge \lambda_2$ we need to sequentially open arcs in $N_2^+ \setminus S_2^+$.

The lifting function, obtained by subtracting from d_{12} the contributions of arcs in $N_i^+ \cup N_i^-$, $i = 1,2$, is then

$$
f_{12}(w_{12}) = \begin{cases} i(\lambda - \lambda_2) & \text{if } ic \le w_{12} \le (i+1)c - \lambda + \lambda_2,\ i \in \mathbb{Z} \\ w_{12} - ic + i(\lambda - \lambda_2) & \text{if } ic - \lambda + \lambda_2 \le w_{12} \le ic,\ i \in \mathbb{Z}. \end{cases}
$$

Function f_{12} is of the form $g_{\lambda-\lambda_2,c}$ defined in Proposition 4, and is, therefore, superadditive in $\mathbb{R}$. Using f_{12}, we obtain the type 1 three-partition flow cover inequality

$$\begin{aligned}
&y(S_1^+ \cup S_2^+) + \sum_{j\in S_1^+} (c-\lambda)(1-x_j) + \sum_{j\in N_1^+\setminus S_1^+} (y_j - (c-\lambda)x_j)^+ - \sum_{j\in N_1^-} \min\{y_j, \lambda x_j\} \\
&+ \sum_{j\in S_2^+} (c-\lambda_2)(1-x_j) + \sum_{j\in N_2^+\setminus S_2^+} (y_j - (c-\lambda_2)x_j)^+ - \sum_{j\in N_2^-} \min\{y_j, \lambda_2 x_j\} + \sum_{j\in S_{12}} (\lambda-\lambda_2)(1-x_j) \\
&+ \sum_{j\in N_{21}} \max\{0, y_j + (\lambda - \lambda_2 - c)x_j\} - \sum_{j\in N_{12}\setminus S_{12}} \min\{y_j, (\lambda-\lambda_2)x_j\} \le d_{12}.
\end{aligned} \tag{14}$$

Validity of type 2 inequalities. The validity of the type 2 three-partition flow cover inequalities when $\lambda_2 > \lambda$ is proved similarly: We assume $x_j = 1$ for $j \in S_1^+ \cup S_2^+ \cup S_{12}$ and $x_j = 0$ for $j \in (N_i^+ \setminus S_i^+) \cup N_i^-$, $i = 1, 2$. Under those assumptions, the lifted flow cover inequality for node 2 yields

$$y(S_2^+ \cup S_{12}) + \sum_{j\in S_{12}} (c-\lambda_2)(1-x_j) + \sum_{j\in N_{12}\setminus S_{12}} \max\{y_j - (c-\lambda_2)x_j, 0\} - \sum_{j\in N_{21}} \min\{y_j, \lambda_2 x_j\} \le d_2.$$

We then lift first variables in $N_2^+ \cup N_2^-$ and then $N_1^+ \cup N_1^-$ to get a valid inequality. The complete proof is in Appendix A.

□

We now give an alternative characterization of inequalities (8).

Remark 3. Note that we can rewrite inequalities (8) as

$$\begin{aligned}
&\sum_{i=1,2} \sum_{j\in S_i^+} (y_j - \rho_i x_j) - \sum_{i=1,2} \sum_{j\in N_i^-} \min\{y_j, (c-\rho_i)x_j\} + \sum_{i=1,2} \sum_{j\in N_i^+\setminus S_i^+} \max\{y_j - \rho_i x_j, 0\} \\
&- \sum_{j\in S_{12}} (\rho_2 - \rho_1)x_j - \sum_{j\in N_{12}\setminus S_{12}} \min\{y_j, (\rho_2-\rho_1)x_j\} + \sum_{j\in N_{21}} \max(0, y_j + (\rho_2 - \rho_1 - c)x_j) \\
&\le d_{12} - \sum_{i=1,2} \rho_i \left|S_i^+\right| - (\rho_2 - \rho_1)\left|S_{12}\right|.
\end{aligned} \tag{15}$$

Furthermore, from the conditions defining a minimal three-partition flow cover, we infer that

Condition 1: $\left|S_1^+\right| \ge \lceil \frac{d_1}{c} \rceil$.

Conditions 2 & 4: $\left|S_2^+\right| + |S_{12}| = \lceil \frac{d_2}{c} \rceil$.

Conditions 3 & 4: $\left|S_1^+\right| + \left|S_2^+\right| = \lceil \frac{d_{12}}{c} \rceil$.

Conditions 2, 3 & 4: $|S_{12}| - \left|S_1^+\right| = |S_{12}| + \left|S_2^+\right| - \left|S_1^+\right| - \left|S_2^+\right| = \lceil \frac{d_2}{c} \rceil - \lceil \frac{d_{12}}{c} \rceil$.

Therefore, we get that inequalities (8) are equivalent to

$$\sum_{i=1,2}\sum_{j\in S_i^+}(y_j-\rho_i x_j)-\sum_{i=1,2}\sum_{j\in N_i^-}\min\{y_j,(c-\rho_i)x_j\}+\sum_{i=1,2}\sum_{j\in N_i^+\setminus S_i^+}\max\{y_j-\rho_i x_j,0\}$$
$$-\sum_{j\in S_{12}}(\rho_2-\rho_1)x_j-\sum_{j\in N_{12}\setminus S_{12}}\min\{y_j,(\rho_2-\rho_1)x_j\}+\sum_{j\in N_{21}}\max(0,y_j+(\rho_2-\rho_1-c)x_j)$$
$$\leq d_{12}+(\rho_1-\rho_2)\lceil\frac{d_2}{c}\rceil-\rho_1\lceil\frac{d_{12}}{c}\rceil. \tag{16}$$

3.3 Strength of the three-partition flow cover inequalities

In this section we study the strength of inequalities (8). We prove that, under mild conditions, they are facet defining for $T_{\leq}$. We also show that, in some cases, they dominate flow cover inequalities.

3.3.1 Facet-defining conditions

Proposition 7. *$T_{\leq}$ is full dimensional when $d_1 > 0$ and $d_2 > 0$.*

Proof. Let e_i be i-th unit vector and let $0 < \epsilon \leq \min\{d_1, d_2, c\}$. The set

$$\{0\}\cup\bigcup_{i=1}^{n}\{(x,y)=(e_i,0)\}\cup\bigcup_{i=1}^{n}\{(x,y)=(e_i,\epsilon e_i)\}$$

contains $2n+1$ affinely independent points belonging to $T_{\leq}$. □

Theorem 2. *The three-partition flow cover inequalities* (8) *are facet defining for $T_{\leq}$ when $d_1 > 0$, $d_2 > 0$, $S_2^+ \neq \emptyset$ and $S_{12} \neq \emptyset$.*

Proof. To prove that, under the conditions of the theorem, inequalities (8) are facet defining for $T_{\leq}$, we provide, for each type of three-partition flow cover inequality, $2n$ affinely independent points where the inequality holds at equality. Let $\bar{1} = (1, 1, \ldots, 1)$, $\bar{c} = (c, c, \ldots, c)$ and let e_i be the i-th unit vector. For clarity, we give each point in the format

$$\left(x^{S_1^+}, y^{S_1^+}, x^{S_2^+}, y^{S_2^+}, x^{S_{12}}, y^{S_{12}}, x^{N\setminus\mathcal{S}}, y^{N\setminus\mathcal{S}}\right),$$

where (y^S, x^S) are the (y, x) values for the arcs in the set S.

Facet proof for type 1 inequalities Let $C_i^+ \subseteq N_i^+ \setminus S_i^+$, $S_i^- \subseteq N_i^-$, $i = 1, 2$, and let $C_{12} \subseteq N_{12} \setminus S_{12}$, $S_{21} \subseteq N_{21}$, and rewrite inequality (14) as

$$\begin{aligned} & y(S_1^+ \cup S_2^+) + \sum_{j \in S_1^+} (c - \lambda)(1 - x_j) + \sum_{j \in C_1^+} (y_j - (c - \lambda)x_j) - \sum_{j \in S_1^-} \lambda x_j - \sum_{j \in N_1^- \setminus S_1^-} y_j \\ &+ \sum_{j \in S_2^+} (c - \lambda_2)(1 - x_j) + \sum_{j \in C_2^+} (y_j - (c - \lambda_2)x_j) - \sum_{j \in S_2^-} \lambda_2 x_j - \sum_{j \in N_2^- \setminus S_2^-} y_j \\ &+ \sum_{j \in S_{12}} (\lambda - \lambda_2)(1 - x_j) + \sum_{j \in S_{21}} (y_j + (\lambda - \lambda_2 - c)x_j) - \sum_{j \in C_{12}} (\lambda - \lambda_2)x_j - \sum_{j \in N_{12} \setminus (S_{12} \cup C_{12})} y_j \le d_{12}. \end{aligned}$$

Note that $\lambda_2 - \lambda < 0$ implies $c(S_{12}) = \lambda_1 + \lambda_2 - \lambda < \lambda_1 < d_1 + \lambda_1 = c(S_1^+)$. Therefore, if $S_{12} \neq \emptyset$ then we have $\left|S_1^+\right| \geq 2$. Let $k_i \in S_i^+$, $i = 1, 2$, let $k_1' \in S_1^+$ with $k_1' \neq k_1$ and let $k_{12} \in S_{12}$. Table 1 shows the affinely independent points for type 1 inequalities. Note that we provide two points for each $j \in N$ (and therefore there are $2n$ points). To check that the points are indeed affinely independent, observe that the two points corresponding to each $i \in N \setminus \mathcal{S}$ ensure that $x_i \neq 0$ and $y_i \neq kx_i$ (for some $k \in \mathbb{R}_+$).

Facet proof for type 2 inequalities Let $C_i^+ \subseteq N_i^+ \setminus S_i^+$, $S_i^- \subseteq N_i^-$, $i = 1, 2$, and let $C_{12} \subseteq N_{12} \setminus S_{12}$, $S_{21} \subseteq N_{21}$. Rewrite inequalities (8) as

$$\begin{aligned} & y(S_1^+ \cup S_2^+) + \sum_{j \in S_1^+} (\lambda_2 - \lambda)(1 - x_j) + \sum_{j \in C_1^+} (y_j - (\lambda_2 - \lambda)x_j) - \sum_{j \in S_1^-} (c - (\lambda_2 - \lambda))x_j \\ &- \sum_{j \in N_1^- \setminus S_1^-} y_j + \sum_{j \in S_2^+} (c - \lambda)(1 - x_j) + \sum_{j \in C_2^+} (y_j - (c - \lambda)x_j) - \sum_{j \in S_2^-} \lambda x_j - \sum_{j \in N_2^- \setminus S_2^-} y_j \\ &+ \sum_{j \in S_{12}} (c - \lambda_2)(1 - x_j) + \sum_{j \in S_{21}} (y_j - \lambda_2 x_j) - \sum_{j \in C_{12}} (c - \lambda_2)x_j - \sum_{j \in N_{12} \setminus (S_{12} \cup C_{12})} y_j \le d_{12}. \end{aligned}$$

Let $k_i \in S_i^+$, $i = 1, 2$, and let $k_{12} \in S_{12}$. Table 2 shows the $2n$ affinely independent points for type 2 inequalities. To check that the points are indeed affinely independent, observe that that the point corresponding to $i = k_1$ with $c_{k_1} = c$ is affinely independent from the previously introduced points since it is the first point where $y(S_1^+) \neq y(S_{12}) + d_1$, and the two points corresponding to each $i \in N \setminus \mathcal{S}$ ensure that $x_i \neq 0$ and $y_i \neq kx_i$ (for some $k \in \mathbb{R}_+$).

□

3.3.2 Comparison with the flow cover inequalities

Recall from Remark 2 that, depending on the values of λ and λ_2, either the type 1 or the type 2 three-partition flow cover inequalities reduce to lifted flow cover inequalities. We now give conditions under which inequalities (8) dominate the regular flow cover inequalities.

Proposition 8. *When $\lambda > \lambda_2$ and $S_{12} = N_{12}$, the type 1 three-partition inequalities with minimal flow cover (S_1^+, S_2^+, S_{12}) dominate the corresponding flow cover inequalities for both nodes aggregated with flow cover $S_1^+ \cup S_2^+$.*

Table 1: Affinely independent points for Type 1 inequalities.

Condition	S_1^+ y	S_1^+ x	S_2^+ y	S_2^+ x	S_{12} y	S_{12} x	$N \setminus \mathcal{S}$ y	$N \setminus \mathcal{S}$ x
$i \in S_1^+$	$\bar{c} - \lambda e_i$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	0	0
	$\bar{c} - c e_i$	$\bar{1} - e_i$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	0	0
$i \in S_1^-$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	λe_i	e_i
	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	$c e_i$	e_i
$i \in N_1^- \setminus S_1^-$	$\bar{c} - \lambda e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	0	e_i
	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	λe_i	e_i
$i \in S_{12}$	$\bar{c} - c e_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - (c - \lambda - \lambda_2) e_i$	$\bar{1}$	0	0
	$\bar{c} - c e_{k_1} - (\lambda - \lambda_2) e_{k_1'}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - c e_i$	$\bar{1} - e_i$	0	0
$i \in S_{21}$	$\bar{c} - c e_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$(c - \lambda + \lambda_2) e_i$	e_i
	$\bar{c} - c e_{k_1} - (\lambda - \lambda_2) e_{k_1'}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$c e_i$	e_i
$i \in N_{21} \setminus S_{21}$	$\bar{c} - c e_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	0	e_i
	$\bar{c} - c e_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\lambda_2 e_i$	e_i
$i \in S_2^+$	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c} - \lambda_2 e_i$	$\bar{1}$	$\bar{c}$	$\bar{1}$	0	0
	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c} - c e_i$	$\bar{1} - e_i$	$\bar{c}$	$\bar{1}$	0	0
$i \in S_2^-$	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\lambda_2 e_i$	e_i
	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} e_i$	e_i
$i \in N_2^- \setminus S_2^-$	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_2}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	0	e_i
	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\lambda_2 e_i$	e_i
$i \in C_1^+$	$\bar{c} - c e_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	$(c - \lambda) e_i$	e_i
	$\bar{c} - c e_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c} - \lambda_2 e_{k_2}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$(c - \lambda + \lambda_2) e_i$	e_i
$i \in N_1^+ \setminus (S_1^+ \cup C_1^+)$	$\bar{c} - c e_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	0	e_i
	$\bar{c} - c e_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	$(c - \lambda) e_i$	e_i
$i \in C_2^+$	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c} - c e_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c}$	$\bar{1}$	$(c - \lambda_2) e_i$	e_i
	$\bar{c} - \lambda e_{k_1}$	$\bar{1}$	$\bar{c} - c e_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	$c e_i$	e_i
$i \in N_2^+ \setminus (S_2^+ \cup C_2^+)$	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c} - c e_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c}$	$\bar{1}$	0	e_i
	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c} - c e_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c}$	$\bar{1}$	$(c - \lambda_2) e_i$	e_i
$i \in C_{12}$	$\bar{c} - \lambda e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - c e_{k_{12}}$	$\bar{1} - e_{k_{12}}$	$(c - \lambda_2) e_i$	e_i
	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_2}$	$\bar{1}$	$\bar{c} - c e_{k_{12}}$	$\bar{1} - e_{k_{12}}$	$c e_i$	e_i
$i \in N_{12} \setminus (S_{12} \cup C_{12})$	$\bar{c} - (\lambda - \lambda_2) e_{k_1}$	$\bar{1}$	$\bar{c} - c e_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c}$	$\bar{1}$	0	e_i
	$\bar{c}$	$\bar{1}$	$\bar{c} - c e_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c}$	$\bar{1}$	$(\lambda - \lambda_2) e_i$	e_i

Proof. Consider the type 1 three-partition inequalities when $S_{12} = N_{12}$

$$\begin{aligned} & y(S_1^+ \cup S_2^+) + \sum_{j \in S_1^+} (c - \lambda)(1 - x_j) - \sum_{j \in N_1^-} \min\{y_j, \lambda x_j\} \\ & + \sum_{j \in S_2^+} \boldsymbol{(c - \lambda_2)}(1 - x_j) - \sum_{j \in N_2^-} \min\{y_j, \boldsymbol{\lambda_2} x_j\} \\ & + \sum_{\boldsymbol{j \in N_{12}}} \boldsymbol{(\lambda - \lambda_2)(1 - x_j)} + \sum_{\boldsymbol{j \in N_{21}}} \boldsymbol{\max\{0, y_j + (\lambda - \lambda_2 - c) x_j\}} \leq d_{12}, \end{aligned} \tag{17}$$

Table 2: Affinely independent points for Type 2 inequalities.

Condition	S_1^+ y	S_1^+ x	S_2^+ y	S_2^+ x	S_{12} y	S_{12} x	$N \setminus \mathcal{S}$ y	$N \setminus \mathcal{S}$ x
$i \in S_{12}$	$\bar{c} - \lambda e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_i$	$\bar{1}$	0	0
	$\bar{c} - (c - \lambda_2 + \lambda)e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_i$	$\bar{1} - e_i$	0	0
$i \in S_2^+$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda e_i$	$\bar{1}$	$\bar{c} - (\lambda_2 - \lambda)e_{k_{12}}$	$\bar{1}$	0	0
	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_i$	$\bar{1} - e_i$	$\bar{c} - (\lambda_2 - \lambda)e_{k_{12}}$	$\bar{1}$	0	0
$i = k_1$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c}$	$\bar{1}$	0	0
	$\bar{c} - ce_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	0	0
$i \in S_1^+ \setminus \{k_1\}$	$\bar{c} - (c - \lambda_2 + \lambda)e_i$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	0	0
	$\bar{c} - ce_i$	$\bar{1} - e_i$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	0	0
$i \in S_{21}$	$\bar{c} - \lambda e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\lambda_2 e_i$	e_i
	$\bar{c} - (c - \lambda_2 + \lambda)e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	ce_i	e_i
$i \in N_{21} \setminus S_{21}$	$\bar{c} - \lambda e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	0	e_i
	$\bar{c} - \lambda e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	λ_2	e_i
$i \in S_2^-$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - (\lambda_2 - \lambda)e_{k_{12}}$	$\bar{1}$	λe_i	e_i
	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - (\lambda_2 - \lambda)e_{k_{12}}$	$\bar{1}$	ce_i	e_i
$i \in N_2^- \setminus S_2^-$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda e_{k_2}$	$\bar{1}$	$\bar{c} - (\lambda_2 - \lambda)e_{k_{12}}$	$\bar{1}$	0	1
	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - (\lambda_2 - \lambda)e_{k_{12}}$	$\bar{1}$	λ	1
$i \in S_1^-$	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	$(c - \lambda_2 + \lambda)e_i$	e_i
	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	ce_i	e_i
$i \in N_1^- \setminus S_1^-$	$\bar{c} - (c - \lambda_2 + \lambda)e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	0	e_i
	$\bar{c}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	$(c - \lambda_2 + \lambda)e_i$	e_i
$i \in C_{12}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	$(c - \lambda_2 + \lambda)e_i$	e_i
	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	ce_i	e_i
$i \in N_{12} \setminus (S_{12} \cup C_{12})$	$\bar{c} - \lambda e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	0	e_i
	$\bar{c} - \lambda e_{k_1}$	$\bar{1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	$(c - \lambda_2)$	e_i
$i \in C_1^+$	$\bar{c} - ce_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c} - \lambda e_{k_2}$	$\bar{1}$	$\bar{c} - (\lambda_2 - \lambda)e_{k_{12}}$	$\bar{1}$	c	e_i
	$\bar{c} - ce_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	$(\lambda_2 - \lambda)e_i$	e_i
$i \in N_1^+ \setminus (S_1^+ \cup C_1^+)$	$\bar{c} - ce_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	0	e_i
	$\bar{c} - ce_{k_1}$	$\bar{1} - e_{k_1}$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_{12}}$	$\bar{1} - e_{k_{12}}$	$(\lambda_2 - \lambda)e_i$	e_i
$i \in C_2^+$	$\bar{c} - \lambda e_{k_1}$	$\bar{1}$	$\bar{c} - ce_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c} - \lambda_2 e_{k_{12}}$	$\bar{1}$	ce_i	e_i
	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c} - (\lambda_2 - \lambda)e_{k_{12}}$	$\bar{1}$	$(c - \lambda)e_i$	e_i
$i \in N_2^+ \setminus (S_2^+ \cup C_2^+)$	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c} - (\lambda_2 - \lambda)e_{k_{12}}$	$\bar{1}$	0	e_i
	$\bar{c}$	$\bar{1}$	$\bar{c} - ce_{k_2}$	$\bar{1} - e_{k_2}$	$\bar{c} - (\lambda_2 - \lambda)e_{k_{12}}$	$\bar{1}$	$(c - \lambda)e_i$	e_i

and the flow cover inequalities

$$
\begin{aligned}
y(S_1^+ \cup S_2^+) + \sum_{j \in S_1^+} (c - \lambda)(1 - x_j) - \sum_{j \in N_1^-} \min\{y_j, \lambda x_j\} \\
+ \sum_{j \in S_2^+} \boldsymbol{(c - \lambda)}(1 - x_j) - \sum_{j \in N_2^-} \min\{y_j, \boldsymbol{\lambda} x_j\} \le d_{12}.
\end{aligned}
$$

Observe that the coefficients multiplying the terms $(1 - x_j)$ for $j \in S_2^+$ and $-x_j$ for $j \in N_2^-$ are stronger for the three-partition flow cover inequality. Moreover, inequality (17) is further strengthened by the nonnegative terms corresponding to the arcs in $N_{12} \cup N_{21}$. $\square$

Example 1 (continued)**.** Consider the network depicted in Figure 3. Note that if we delete arc 4 from the network, then the corresponding three-partition flow cover inequality with flow cover $\{1, 2, 5\}$

$$y_1 - 2x_1 + y_2 - 2x_2 + y_5 - 5x_5 + 3(1 - x_3) + (y_6 - 7x_6)^+ \leq 13$$

dominates the flow cover inequality (9).

Proposition 9. *When $\lambda < \lambda_2$ and $S_1^+ = N_1^+$, the type 2 three-partition inequalities with minimal flow cover (S_1^+, S_2^+, S_{12}) dominate the corresponding flow cover inequalities for node 2 with flow cover $S_2^+ \cup S_{12}$.*

Proof. Note that, after adding the flow conservation constraint (6) to the flow cover inequality for node 2, we get the equivalent inequality

$$\begin{aligned} y(N_1^+ \cup S_2^+) - y(N_1^-) + \sum_{j \in S_2^+} (c - \lambda_2)(1 - x_j) - \sum_{j \in N_2^-} \min\{y_j, \lambda_2 x_j\} \\ + \sum_{j \in S_{12}} (c - \lambda_2)(1 - x_j) - \sum_{j \in N_{12} \setminus S_{12}} \min\{y_j, (c - \lambda_2) x_j\} + \sum_{j \in N_{21}} (y_j - \lambda_2 x_j)^+ \leq d_{12}, \end{aligned}$$

and compare it with the three-partition flow cover inequality

$$\begin{aligned} & y(N_1^+ \cup S_2^+) - y(N_1^-) + \sum_{\boldsymbol{j \in N_1^-}} \boldsymbol{(y_j - (c - (\lambda_2 - \lambda)) x_j)^+} + \sum_{\boldsymbol{j \in N_1^+}} \boldsymbol{(\lambda_2 - \lambda)(1 - x_j)} \\ & + \sum_{j \in S_2^+} \boldsymbol{(c - \lambda)}(1 - x_j) - \sum_{j \in N_2^-} \min\{y_j, \boldsymbol{\lambda} x_j\} + \sum_{j \in S_{12}} (c - \lambda_2)(1 - x_j) \\ & - \sum_{j \in N_{12} \setminus S_{12}} \min\{y_j, (c - \lambda_2) x_j\} + \sum_{j \in N_{21}} (y_j - \lambda_2 x_j)^+ \leq d_{12}. \end{aligned} \tag{18}$$

Observe that the coefficients multiplying the terms $(1 - x_j)$ for $j \in S_2^+$ and $-x_j$ for $j \in N_2^-$ are stronger for the three-partition flow cover inequality. Moreover, inequality (18) is further strengthened by the nonnegative terms corresponding to the arcs in $N_1^+ \cup N_1^-$. □

Example 2 (continued)**.** Consider the network depicted in Figure 4. Note that if we add the flow conservation constraint of node 1 to the flow cover inequality, then we get the equivalent inequality

$$y_1 - 4x_2 - \min\{y_3, 4x_3\} - \min\{y_5, 6x_5\} \leq 2,$$

which is dominated by the three-partition flow cover inequality

$$y_1 - 2x_1 - 4x_2 - \min\{y_3, 4x_3\} - \min\{y_5, 4x_5\} \leq 0.$$

Remark 4. Note that the condition $S_{12} = N_{12}$ for the type 1 inequalities is naturally satisfied when there is a single arc between nodes 1 and 2. The observation suggests that the type 1 inequalities are particularly effective (with respect to the flow cover inequalities) when nodes

1 and 2 are single nodes (and the implicit node 0 corresponds to the aggregation of all other nodes in the CFNF). Similarly, type 2 inequalities may be particularly effective when nodes 0 and 1 are single nodes, and node 2 corresponds to the aggregation of all other nodes in the CFNF.

Finally, we close the section by noting that three-partition inequalities and flow cover inequalities are not sufficient to characterize T or $T_{\leq}$.

Example 2 (continued)**.** Consider the network depicted in Figure 4. The inequality

$$-6x_1 + 3y_1 - 16x_2 + y_2 - 16x_3 + y_3 - 12x_4 + 3y_4 - 12x_5 - y_5 \leq 0$$

is facet defining for T and $T_{\leq}$, but is neither a three-partition flow cover inequality nor a flow cover inequality.

4 Separation

Given a fractional solution, the separation problem consists of finding a three-partition flow cover inequality that cuts off that solution if there exists any. In Section 4.1 we describe an algorithm that, given a three-partitioning of the nodes, finds a most violated three-partition flow cover inequality. In Section 4.2 we provide different heuristics for choosing the three-partitions.

4.1 Choosing a minimal cover

In this section we give a polynomial separation algorithm for finding a most violated three-partition flow cover inequality given a three-partitioning of the nodes.

From inequality (16), we observe that finding a most violated inequality for a given fractional solution (y, x) consists of choosing sets $C_1^+ \subseteq N_1^+$, $C_2^+ \subseteq N_2^+$ and $S_{12} \subseteq N_{12}$ such that the left-hand-side of

$$\begin{aligned}
&\sum_{i=1,2} \sum_{j \in C_i^+} (y_j - \rho_i x_j) - \sum_{i=1,2} \sum_{j \in N_i^-} \min\{y_j, (c - \rho_i) x_j\} - \sum_{j \in S_{12}} (\rho_2 - \rho_1) x_j \\
&- \sum_{j \in N_{12} \setminus S_{12}} \min\{y_j, (\rho_2 - \rho_1) x_j\} + \sum_{j \in N_{21}} \max(0, y_j + (\rho_2 - \rho_1 - c) x_j) \\
&\leq d_{12} + (\rho_1 - \rho_2) \lceil \frac{d_2}{c} \rceil - \rho_1 \lceil \frac{d_{12}}{c} \rceil,
\end{aligned} \tag{19}$$

is maximized[1], where

$$\begin{aligned}
|C_1^+| &\geq \lceil \frac{d_1}{c} \rceil \\
|C_1^+| + |C_2^+| &\geq \lceil \frac{d_{12}}{c} \rceil \\
|C_2^+| + |S_{12}| &\geq \lceil \frac{d_2}{c} \rceil.
\end{aligned} \tag{20}$$

[1]Note that C_i^+ corresponds to S_i^+ plus the arcs in $N_i^+ \setminus S_i^+$ with nonzero terms.

Let $C_i^+ = \{j \in N_i^+ : y_j - \rho_i x_j \geq 0\}$, $i = 1, 2$, and let $S_{12} = \{j \in N_{12} : y_j \geq (\rho_2 - \rho_1)x_j\}$. If conditions (20) are satisfied by this choice of C_1^+, C_2^+, S_{12} and the left-hand-side of (19) is greater than or equal to the right-hand-side, then we have found a most violated inequality. Otherwise, we need to add more elements to the sets C_1^+, C_2^+ or S_{12} so that conditions (20) are satisfied. In order to do so, we keep three sorted lists, for the arcs in $K_i = N_i^+ \setminus C_i^+$, $i = 1, 2$ and $K_{12} = N_{12} \setminus S_{12}$ in which the elements in the list are sorted in descending order of $\varepsilon_j = y_j - \rho_i x_j$ for $j \in K_i$, $i = 1, 2$ and $\varepsilon_j = y_j - (\rho_2 - \rho_1)x_j$ for $j \in K_{12}$. For $i \in \{1, 2, 12\}$, let $K_i[0] = \max_{j \in K_i} \varepsilon_j$, and let $\texttt{first}(K_i)$ be a function that returns and removes the greatest element of K_i. The separation algorithm is described in Algorithm 1. If the left-hand-side of (19) is greater than the right-hand-side at termination of the algorithm, then we have found a violated inequality. Otherwise, there exists no violated three-partition flow cover inequality.

Algorithm 1 Separation algorithm.

Input: C_i^+, $i = 1, 2$; S_{12}; K_i, $i \in \{1, 2, 12\}$; $\lceil \frac{d_i}{c} \rceil$,$i \in \{1, 2, 12\}$.
Output: C_i^+, $i = 1, 2$; S_{12}, the sets that maximize the violation.

while $\left|C_1^+\right| < \lceil \frac{d_1}{c} \rceil$ **do** ▷ Ensures that $\left|C_1^+\right| \geq \lceil \frac{d_1}{c} \rceil$
  $C_1^+ \leftarrow \texttt{first}(K_1)$
end while
while $\lceil \frac{d_{12}}{c} \rceil - \left|C_1^+\right| - \left|C_2^+\right| > (\lceil \frac{d_2}{c} \rceil - \left|C_2^+\right| - |S_{12}|)^+$ **do**
  if $K_1[0] \leq K_2[0]$ **then**
    $C_2^+ \leftarrow \texttt{first}(K_2)$
  else
    $C_1^+ \leftarrow \texttt{first}(K_1)$
  end if
end while
while $(\lceil \frac{d_{12}}{c} \rceil - \left|C_1^+\right| - \left|C_2^+\right|)^+ < \lceil \frac{d_2}{c} \rceil - \left|C_2^+\right| - |S_{12}|$ **do**
  if $K_{12}[0] \leq K_2[0]$ **then**
    $C_2^+ \leftarrow \texttt{first}(K_2)$
  else
    $S_{12} \leftarrow \texttt{first}(K_{12})$
  end if
end while
while $\lceil \frac{d_{12}}{c} \rceil - \left|C_1^+\right| - \left|C_2^+\right| = \lceil \frac{d_2}{c} \rceil - \left|C_2^+\right| - |S_{12}| > 0$ **do**
  if $K_1[0] + K_{12}[0] \leq K_2[0]$ **then**
    $C_2^+ \leftarrow \texttt{first}(K_2)$
  else
    $C_1^+ \leftarrow \texttt{first}(K_1)$
    $S_{12} \leftarrow \texttt{first}(K_{12})$
  end if
end while
return C_1^+, C_2^+, S_{12}

Note that sorting the lists can be done in $O(n \log n)$ time. If the lists are sorted, then each computation of $K_i[0]$ and each call of $\texttt{first}(K_i)$ can be done in constant time. Therefore, the commands inside each loop of Algorithm 1 can be done in constant time. Since an element is removed from the lists at each step, the loops finish in at most $|K_1 \cup K_2 \cup K_{12}| \leq n$ steps, and the complexity of the algorithm after sorting the arcs is $O(n)$.

Proposition 10. *Given a three-partitioning of the network, there exists an $O(n \log n)$ separation algorithm for inequalities* (8).

4.2 Choosing three-partitions

For a general CFNF, the separation algorithm described in Section 4.1 assumes a three-partitioning of the vertices has been chosen. In this section we propose strategies for finding three-partitions that may lead to violated inequalities.

4.2.1 Single nodes

When describing the implementation of flow cover inequalities, Van Roy and Wolsey [14] consider in turn each flow conservation constraint, which is equivalent to considering two-partitions in which one of the partitions is a single node. In a similar spirit, we consider all three-partitions in which two of the partitions are single nodes.

4.2.2 Spanning trees

Stallaert [11] describes a heuristic for finding two-partitions to apply flow cover inequalities. We use an adaptation to the case of three-partitions, which is described next.

Given a solution (x, y) to a CFNF defined on the graph $G = (V, A)$, define an *active arc* as an arc $j \in A$ that is neither void nor saturated, i.e., $0 < y_j < c$, and let $\bar{A}$ be the set of active arcs in the current solution. The algorithm proceeds as follows:

Step 1 Construct a maximum spanning forest with arc weights $1 - (x_j - \frac{y_j}{c})$ for $j \in \bar{A}$ and $-\infty$ otherwise.

Step 2A (Two-partition) For each arc k in the forest, let T_1^k and T_2^k be two trees that result from the deletion of arc k, with node sets $V_{T_1^k}$ and $V_{T_2^k}$. Then for each $i = 1, 2$ there is a two-partition defined by $V_{T_i^k}$ and $V \setminus V_{T_i^k}$. This is the original method reported in [11].

Step 2B For each arc k in the forest and $i = 1, 2$, define $V_{T_i^k}$ as above. Let v_i^k be the single node connected to $V_{T_i^k}$ by k. Then for each $i = 1, 2$ there is a three-partitioning defined by $V_{T_i^k}$, $\{v_i^k\}$ and $V \setminus (V_{T_i^k} \cup \{v_i^k\})$.

Step 2C For each node l in the forest with at least two adjacent nodes, let $\{T^l\}$ be the collection of trees that result from the deletion of node l. Then for each pair of elements i and j in $\{T^l\}$ there is a three-partitioning defined by $V_{T_i^l}$, $V_{T_j^l}$ and $V \setminus (V_{T_i^k} \cup V_{T_j^k})$.

4.2.3 Extension methods

The idea of the extension methods is to generate new partitions from a set of existing partitions. Given a three-partitioning P, define the characteristic function $e_P : V \to \{0, 1, 2\}$ as the function that maps each vertex to its partition (labeled as 0, 1 and 2), and define b_P as the value of the violation of a most violated three-partition flow cover inequality arising from P. We generate new partitions in the following ways:

Mixture Given partitions P_1 and P_2, generate a new partition P such that for each $v \in V$

$$e_P(v) = \begin{cases} e_{P_1}(v) & \text{if } e_{P_1}(v) = e_{P_2}(v) \text{ or } e_{P_2}(v) = 0 \\ e_{P_2}(v) & \text{if } e_{P_1}(v) = 0 \\ e_{P_m}(v) & \text{otherwise,} \end{cases}$$

where $m = 1$ if $b_{P_1} \geq b_{P_2}$ and $m = 2$ otherwise.

Modification Given a partition P_0, generate a new partition P such that $e_P(v) = e_{P_0}(v)$ for $v \in V \setminus \{l\}$, and $e_P(l) \neq e_{P_0}(l)$. The vertex l and the new value $e_P(l)$ are chosen randomly.

Given $\zeta \in \mathbb{Z}_+$, the algorithm proceeds as follows:

Step 0 Add all partitions in which two of the partitions are single nodes to a pool of partitions.

Step 1 Choose the ζ highest partitions in terms of b_P.

Step 2 For each pair of the selected partitions, generate a new partition using the `Mixture` operation. Add all new partitions to the pool.

Step 3 For each selected partition, generate a new partition using the `Modification` operation. Add all new partitions to the pool.

Step 4 If a termination criterion is met, then terminate the algorithm. Otherwise, return to Step 1.

5 Computational experiments

In this section we report computational experiments with solving CFNF of varying sizes with a cut-and-branch algorithm using CPLEX v12.6.0. All experiments are conducted on one thread of a Dell computer with a 2.2GHz Intel®Core™ i7-2670QM CPU and 8 GB main memory. We test the CPLEX branch-and-bound algorithm using the following configurations for generating cuts:

ALL Adds three-partition flow cover inequalities for all three-partitions using complete enumeration.

FC Adds flow cover inequalities for single nodes and aggregating two nodes.

TP Adds three-partition flow cover inequalities in which two of the partitions are single nodes. Note that configuration TP considers the same partitions as FC (instead of aggregating the two nodes, each node is its own partition).

FC* Adds the inequalities from FC, plus flow cover inequalities derived from the spanning tree heuristic from [11] and from an adaptation of the extension heuristics of Section 4.2.3 to two partitions.

TP* Adds three-partition flow cover inequalities using the strategies proposed in Section 4.2 for selecting the three-partitions.

CP CPLEX in default setting.

With the exception of the configuration CP, CPLEX cuts and heuristics are turned off. For FC* and TP*, we use $\zeta = 50$ for the partition combination heuristic.

All instances are randomly generated as follows. Let $\alpha \in \{40, 60, 80\}$ be a density parameter and let $\beta \in \{1.25, 2\}$ be a capacity parameter. In each instance, 40% of the nodes are demand nodes, 40% are supply nodes, and 20% are transhipment nodes. For each demand node, the demand is randomly generated between 1 and 20. The total supply, equal to the total demand, is distributed equally among the supply nodes. The capacity of the arcs is given by $c = \beta\bar{d}$, where $\bar{d}$ denotes the average demand; note that instances with high β result in weaker LP relaxations, and thus in more difficult instances. Between each pair of nodes there is an arc with probability $\alpha/100$, with fixed cost between 1 and 2000 and variable cost between 1 and 200; for this choice of parameters the fixed and variable costs of an arc at full capacity are of the same order of magnitude.

ALL vs. TP and TP*. First, to test the effectiveness of the strategies for finding three-partitions, we solve small instances with up to 14 nodes. For each instance, we compare the gap improvement obtained by using only single node three-partitions (TP), using the proposed heuristics to find additional partitions (TP*), and considering all partitions by doing complete enumeration (ALL). Table 3 presents the results. Each row represents the average over five randomly generated instances of similar characteristics. The table shows, from left to right: The number of nodes in the instance; the initial gap; the algorithm configuration; the root gap improvement, computed as $100 \times (\texttt{zroot} - \texttt{zinit})/(\texttt{zub} - \texttt{zinit})$, where `zroot` is the LP lower bound after adding the cuts, `zub` is the best integer solution found and `zinit` is the initial LP solution; the number of cuts added by CPLEX; the end gap, as reported by CPLEX; the number of nodes processed in the branch-and-bound tree; and the total time used in seconds. Configuration TP* strikes a good balance between the quality of gap improvement and the solution times. A gap improvement close to complete enumeration is achieved in only a fraction of the time.

FC vs. TP and FC* vs. TP*. Next we test the impact of the three-partition flow cover inequalities for larger instances without the interference of CPLEX cuts. To evaluate the marginal impact of adding three-partition cuts on top of the flow cover cuts, we implemented our version of the lifted flow cover inequalities and tested the versions of the algorithm without separation heurisitics (FC and TP), and using the separation heuristics (FC* and TP*). Tables 4 and 5 present the results for 60-node instances with different capacity-to-demand ratios. Each

Table 3: Heuristics compared to complete enumeration.

Nodes.	**Initial Gap.**	**Config.**	**Gap Impr.**	**Root Time**	**Cuts**	**End Gap**	**Nodes**	**Time**
		ALL	98.3%	16	96	0%	5	16
10	24.9%	TP	85.6%	0	29	0%	37	0
		TP*	98.2%	12	63	0%	8	12
		ALL	97.5%	64	140	0%	18	64
11	23.1%	TP	85.0%	0	33	0%	180	0
		TP*	97.5%	18	69	0%	30	18
		ALL	99.1%	309	121	0%	17	309
12	23.1%	TP	85.4%	0	40	0%	127	0
		TP*	99.1%	19	95	0%	6	19
		ALL	96.5%	930	130	0%	180	931
13	17.8%	TP	79.1%	0	39	0%	283	0
		TP*	96.1%	18	121	0%	163	19
		ALL	97.6%	4,599	123	0%	106	4,600
14	26.8%	TP	81.8%	0	46	0%	257	1
		TP*	97.3%	32	97	0%	74	32
		ALL	**97.8%**	**1,184**	**122**	**0%**	**65**	**1,184**
Average		**TP**	**83.4%**	**0**	**37**	**0%**	**177**	**0**
		TP*	**97.6%**	**20**	**89**	**0%**	**56**	**20**

row represents the average over five randomly generated instances of similar characteristics. We set the time limit to 7200 seconds and the memory limit to 4 GB. The tables show, from left to right: The arc density; the initial gap; the algorithm configuration; the root gap improvement; the number of user cuts and cuts added by CPLEX; the end gap, as reported by CPLEX; the number of nodes processed in the branch-and-bound tree; the total time used; the results of the five instances, where S denotes the number of instances solved to optimality, T denotes the number of instances that timed out and M denotes the number of instances that used all the available memory.

Using configurations TP and TP* results in an additional root gap improvement of 12.7% and 2.2% over configurations FC and FC*, respectively. We observe that option TP also improves over FC in terms of end gaps, resulting in reductions of 2.8% and 8.7% in the low and high capacity instances, respectively. In the low capacity instances option TP* results in a small increase of 0.2% in end gaps with respect to FC*, but in the high capacity instances TP* is clearly superior, with a decrease of 7.0% in end gaps. Overall we see that three-partition flow cover inequalities are particularly effective for the high capacity instances. Indeed, three-partition cuts increase the size of the formulation (making the LPs harder to solve), but on the harder instances the stronger formulation typically results in better overall performance.

Note that since the memory limit is reached in many of the instances, high run times indicate a better ability to prune in the branch-and-bound tree (instead of faster times to reach

Table 4: Instances with $\beta = 1.25$ and CPLEX cuts off.

Dens.	Initial Gap.	Config.	Gap Impr.	Cuts User	Cuts CPLEX	End Gap	Nodes	Time	Result S	Result T	Result M
40	24.1%	FC	67.6%	153	0	5.3%	739,986	1,592	1	0	4
		FC+TP	80.5%	232	0	4.5%	352,235	1,212	0	0	5
		FC*	86.7%	289	0	1.9%	305,280	1,212	2	0	3
		FC*+TP*	89.7%	255	0	0.6%	391,779	2,823	3	0	2
60	22.9%	FC	68.3%	164	0	6.1%	597,625	2,073	0	1	4
		FC+TP	82.1%	243	0	5.8%	238,024	1,329	1	0	4
		FC*	87.7%	273	0	1.7%	263,661	1,933	3	0	2
		FC*+TP*	90.0%	271	0	3.2%	240,076	3,682	2	0	3
80	22.9%	FC	69.7%	169	0	16.3%	195,835	723	0	0	5
		FC+TP	83.4%	284	0	9.0%	183,142	1,553	1	0	4
		FC*	88.4%	306	0	1.6%	185,477	1,954	3	0	2
		FC*+TP*	90.0%	286	0	1.9%	220,075	3,570	3	0	2
Average		**FC**	**68.4%**	**162**	**0**	**9.2%**	**511,149**	**1,462**			
		FC+TP	**81.8%**	**253**	**0**	**6.4%**	**257,800**	**1,364**			
		FC*	**87.8%**	**289**	**0**	**1.7%**	**251,473**	**1,818**			
		FC*+TP*	**90.2%**	**273**	**0**	**1.9%**	**283,977**	**3,017**			

Table 5: Instances with $\beta = 2$ and CPLEX cuts off.

Dens.	Initial Gap.	Config.	Gap Impr.	Cuts User	Cuts CPLEX	End Gap	Nodes	Time	Result S	Result T	Result M
40	33.7%	FC	73.2%	166	0	11.9%	433,320	830	0	0	5
		FC+TP	83.9%	352	0	7.1%	311,385	1,441	0	0	5
		FC*	88.1%	300	0	3.1%	457,307	2,063	2	0	3
		FC*+TP*	90.0%	294	0	2.7%	465,086	3,394	2	0	3
60	33.66%	FC	73.5%	176	0	17.2%	531,709	1,279	0	0	5
		FC+TP	83.8%	302	0	14.0%	247,659	1,552	0	0	5
		FC*	87.8%	301	0	9.9%	311,026	2,064	3	0	2
		FC*+TP*	89.7%	302	0	3.7%	223,304	2,908	3	0	2
80	36.2%	FC	69.8%	160	0	40.3%	228,745	595	0	0	5
		FC+TP	83.0%	334	0	21.2%	147,863	1,504	0	0	5
		FC*	88.3%	333	0	20.2%	201,030	1,957	0	0	5
		FC*+TP*	90.6%	342	0	6.1%	234,303	4,126	0	0	5
Average		**FC**	**71.6%**	**166**	**0**	**22.4%**	**471,001**	**1,295**			
		FC+TP	**83.6%**	**330**	**0**	**13.7%**	**259,186**	**1,815**			
		FC*	**88.1%**	**311**	**0**	**11.1%**	**323,121**	**2,028**			
		FC*+TP*	**90.1%**	**320**	**0**	**4.1%**	**307,565**	**3,476**			

optimality).

CP vs. TP* Finally, we test the benefit of adding three-partition flow cover inequalities to CPLEX with default configuration. Note that CPLEX uses flow cover inequalities (among other cuts) and considers many levels of aggregation. Tables 6 and 7 present the results for different node sizes. Three-partition flow cover inequalities help to close an additional 2.9% of the root gap on top of default CPLEX, and result in a better overall performance. For the 60-node instances, configuration TP* solves 28/30 instances to optimality (as opposed to 26/30 of CP), with a reduction of 28% in solution times and 23% of the branch-and-bound tree size. For the 100-node instances, although none of the instances are solved to optimality, configuration TP* results in a reduction of 18% in the end gaps. Note that for the 100-node instances, since the memory limit is reached in many instances, high run times may indicate a better ability to prune in the branch-and-bound tree.

Table 6: 60-node instances with CPLEX cuts on.

Cap.	Dens.	Initial Gap.	Config.	Gap Impr.	Cuts User	Cuts CPLEX	End Gap	Nodes	Time	Result S	Result T	Result M
1.25	40	24.1%	CP	87.4%	0	666	0.0%	100,354	1,326	5	0	0
			CP+TP*	90.9%	365	389	0.0%	66,957	972	5	0	0
	60	22.9%	CP	87.0%	0	682	0.0%	47,993	836	5	0	0
			CP+TP*	91.0%	352	352	0.0%	34,518	665	5	0	0
	80	22.9%	CP	86.7%	0	763	0.2%	72,140	2,697	4	1	0
			CP+TP*	91.3%	374	350	0.0%	40,145	1,187	5	0	0
2.00	40	33.7%	CP	88.6%	0	781	0.0%	115,583	1,241	5	0	0
			CP+TP*	90.9%	471	456	0.0%	85,612	1,287	5	0	0
	60	33.66%	CP	87.2%	0	897	0.5%	144,517	3,693	3	2	0
			CP+TP*	90.2%	467	497	0.0%	156,200	3,922	4	1	0
	80	36.2%	CP	88.1%	0	963	0.4%	122,928	4,584	4	1	0
			CP+TP*	91.3%	439	495	0.4%	80,422	2,289	4	1	0
Average			**CP**	**87.5%**	**0**	**792**	**0.2%**	**100,586**	**2,396**			
			CP+TP*	**90.9%**	**411**	**423**	**0.1%**	**77,309**	**1,720**			

6 Conclusion

We derived new valid inequalities for CFNF problems from three-partitions of a network. The inequalities share the same spirit as flow cover inequalities, but exploit the internal network structure that aggregated flow cover inequalities ignore. We implemented the inequalities as cutting planes in a branch-and-bound approach using CPLEX, and compared the benefits of using the three-partition flow cover inequalities under different algorithm configurations.

According to our computational experiments, using three-partition flow cover inequalities results in stronger formulations, allowing additional gap improvement at the root node. The improvement often translates to lower end gaps at termination and faster solution times.

The proof technique used in this paper can in principle be used to derive valid inequalities for k-partitions, or for the general three-partition polytope with varying capacities: Fix variables in order to apply known results for polytopes with fewer partitions, and then lift the variables assumed to be fixed. Note, however, that computing the necessary lifting functions, and finding suitable superadditive lower bounds, may become more difficult as the complexity of the polytope increases.

Table 7: 100-node instances with CPLEX cuts on.

Cap.	Dens.	Initial Gap.	Config.	Gap Impr.	Cuts User	Cuts CPLEX	End Gap	Nodes	Time	Result S	Result T	Result M
1.25	40	24.7%	CP	85.5%	0	1,161	2.3%	116,824	7,200	0	5	0
			CP+TP*	88.2%	547	586	2.1%	129,288	6,552	0	3	2
	60	23.1%	CP	86.3%	0	1,127	2.0%	68,705	7,200	0	5	0
			CP+TP*	89.1%	371	524	1.6%	112,761	6,211	0	2	3
	80	23.4%	CP	85.7%	0	1,152	2.3%	51,214	7,200	0	5	0
			CP+TP*	89.4%	367	593	1.5%	95,069	6,799	0	4	1
2.00	40	35.0%	CP	88.2%	0	1,276	2.4%	118,159	6,691	0	4	1
			CP+TP*	90.3%	576	651	2.1%	126,190	6,270	0	3	2
	60	35.4%	CP	89.5%	0	1,311	2.2%	71,460	7,200	0	5	0
			CP+TP*	91.4%	464	610	1.9%	109,499	6,396	0	2	3
	80	34.4%	CP	89.8%	0	1,297	2.0%	72,192	7,200	0	5	0
			CP+TP*	91.5%	393	756	1.5%	121,486	6,630	0	3	2
Average			**CP**	**87.5%**	**0**	**1,221**	**2.2%**	**83,092**	**7,115**			
			CP+TP*	**89.9%**	**453**	**620**	**1.8%**	**115,716**	**6,476**			

Acknowledgements

Alper Atamtürk is supported, in part, by grant 0970180 from the National Science Foundation and grant FA9550-10-1-0168 from the Office of the Assistant Secretary of Defense for Research and Engineering. Simge Küçükyavuz is supported, in part, by the National Science Foundation grant 1055668.

References

[1] Y.K. Agarwal, k-partition-based facets of the network design problem, Networks 47 (2006), 123–139.

[2] A. Atamtürk, Flow pack facets of the single node fixed-charge flow polytope, Oper Res Lett 29 (2001), 107–114.

[3] A. Atamtürk and S. Küçükyavuz, Lot sizing with inventory bounds and fixed costs: Polyhedral study and computation, Oper Res 53 (2005), 711–730.

[4] D. Bienstock, S. Chopra, O. Günlük, and C.Y. Tsai, Minimum cost capacity installation for multicommodity network flows, Math Program 81 (1998), 177–199.

[5] M.X. Goemans, Valid inequalities and separation for mixed 0-1 constraints with variable upper bounds, Oper Res Lett 8 (1989), 315 – 322.

[6] M. Grötschel, L. Lovász, and A. Schrijver, The ellipsoid method and its consequences in combinatorial optimization, Combinatorica 1 (1981), 169–197.

[7] Z. Gu, G.L. Nemhauser, and M.W. Savelsbergh, Lifted flow cover inequalities for mixed 0-1 integer programs, Math Program 85 (1999), 439–467.

[8] O. Günlük, A branch-and-cut algorithm for capacitated network design problems, Math Program 86 (1999), 17–39.

[9] T. Magnanti, P. Mirchandani, and R. Vachani, The convex hull of two core capacitated network design problems, Math Program 60 (1993), 233–250.

[10] M.W. Padberg, T.J. Van Roy, and L.A. Wolsey, Valid linear inequalities for fixed charge problems, Oper Res 33 (1985), 842–861.

[11] J. Stallaert, Valid inequalities and separation for capacitated fixed charge flow problems, Discr Appl Math 98 (2000), 265 – 274.

[12] T.J. Van Roy and L.A. Wolsey, Valid inequalities and separation for uncapacitated fixed charge networks, Oper Res Lett 4 (1985), 105 – 112.

[13] T.J. Van Roy and L.A. Wolsey, Valid inequalities for mixed 0-1 programs, Discr Appl Math 14 (1986), 199 – 213.

[14] T.J. Van Roy and L.A. Wolsey, Solving mixed integer programming problems using automatic reformulation, Oper Res 35 (1987), 45–57.

[15] L.A. Wolsey, Submodularity and valid inequalities in capacitated fixed charge networks, Oper Res Lett 8 (1989), 119 – 124.

A Validity of type 2 inequalities

Let $\mathcal{S}$ be a cover, and assume $x_j = 1$ for $j \in S_1^+ \cup S_2^+ \cup S_{12}$ and $x_j = 0$ for $j \in (N_i^+ \setminus S_i^+) \cup N_i^-$, $i = 1, 2$. Under these assumptions, the lifted flow cover inequality for node 2 yields

$$y(S_2^+ \cup S_{12}) + \sum_{j \in S_{12}} (c - \lambda_2)(1 - x_j) + \sum_{j \in N_{12} \setminus S_{12}} \max\{y_j - (c - \lambda_2)x_j, 0\} - \sum_{j \in N_{21}} \min\{y_j, \lambda_2 x_j\} \leq d_2.$$

By adding the flow conservation for node 1, $y(S_1^+) - y(N_{12}) + y(N_{21}) \leq d_1$, we get

$$y(S_1^+ \cup S_2^+) + \sum_{j \in S_{12}} (c - \lambda_2)(1 - x_j) - \sum_{j \in N_{12} \setminus S_{12}} \min\{(c - \lambda_2)x_j, y_j\} + \sum_{j \in N_{21}} \max\{0, y_j - \lambda_2 x_j\} \leq d_{12}. \tag{21}$$

The lifting function associated with simultaneously lifting inequality (21) with variables x_j for $j \in S_2^+$ and pairs (y_j, x_j) for $j \in (N_1^- \cup (N_1^+ \setminus S_2^+))$ is given by

$$\begin{aligned}
\bar{f}_2(w_2, z_2) = \quad & \min \Big\{ d_{12} - y(S_1^+ \cup S_2^+) - \sum_{j \in S_{12}} (c - \lambda_2)(1 - x_j) \\
& + \sum_{j \in N_{12} \setminus S_{12}} \min\{(c - \lambda_2)x_j, y_j\} - \sum_{j \in N_{21}} \max\{0, y_j - \lambda_2 x_j\} \Big\} \\
& \text{s.t. } y(S_1^+) - y(N_{12}) + y(N_{21}) \leq d_1 & (22) \\
& \quad y(S_2^+) + y(N_{12}) - y(N_{21}) \leq d_2 + w_2 & (23) \\
& \quad y(S_1^+) \leq d_1 + \lambda_1 & (24) \\
& \quad y(S_2^+) \leq d_2 + \lambda - \lambda_1 - z_2 & (25) \\
& \quad y(S_{12}) \leq \lambda_1 + \lambda_2 - \lambda \\
& \quad 0 \leq y_j \leq x_j, x_j \in \{0, 1\} \quad j \in N_{12} \cup N_{21},
\end{aligned}$$

where z_2 is a nonnegative multiple of the capacity c and stands for the capacity closed on arcs in S_2^+, $w_2 \geq 0$ stands for the flow on arcs in N_2^-, $w_2 < 0$ stands for the flow on arcs in $N_2^+ \setminus S_2^+$. Note that either (22) or (24) is binding (otherwise we can increase $y(S_1^+)$ and obtain a better solution), and either (23) or (25) is binding. We consider then the following four cases, depending on which equations are binding:

(24) **and** (25) In this case $c(S_{12}) - \lambda_2 + \lambda = \lambda_1 \leq y(N_{12}) - y(N_{21}) \leq \lambda_1 - \lambda + w_2 + z_2$ (which implies $w_2 + z_2 \geq \lambda$). An optimal solution exists by setting $y(S_{12}) = \lambda_1$, in which case $\bar{f}_2(w_2, z_2) = -\lambda + z_2$.

(24) **and** (23) In this case an optimal solution exists where $y(S_2^+)$ is as high as possible and $y(N_{12}) - y(N_{21})$ is as low as possible. Therefore, we have $y(S_{12}) = \lambda_1$ and $y(S_2^+) = d_2 - \lambda_1 + w_2$ (which implies $w_2 + z_2 \leq \lambda$ and is only feasible if $d_2 - \lambda_1 + w_2 \geq 0$). In this case $\bar{f}_2(w_2, z_2) = -w_2$.

(22) **and** (25) In this case $y(N_{12})-y(N_{21}) \leq \lambda_1-\lambda+w_2+z_2$ and $y(S_1^+) \leq d_1+\lambda_1-\lambda+w_2+z_2$. An optimal solution exists where $y(S_1^+)$ is as high as possible and, therefore, we have $y(S_1^+) = d_1 + \lambda_1 - \lambda + w_2 + z_2$ (which implies $w_2 + z_2 \leq \lambda$) and $y(N_{12}) - y(N_{21}) = \lambda_1 - \lambda + w_2 + z_2$. Note that if $z_2 + w_2 \leq 0$ then it may be optimal to sequentially open arcs in $N_{21} \cup (N_{12} \setminus S_{12})$ or close arcs in S_{12}, but for $z_2+w_2 \geq 0$ we have $\bar{f}_2(w_2, z_2) = -w_2$.

(22) **and** (23) Since $y(S_1^+ \cup S_2^+) = d_{12} + w_2$ for all values $y(N_{12}) - y(N_{21})$, an optimal solution exists where $y(N_{12}) - y(N_{21})$ is as low as possible. This value is obtained when $y(S_2^+)$ is maximal, and this case reduces to case (22) and (25).

Combining the different cases, we get the lifting function

$$\bar{f}_2(z_2, w_2) = -w_2 + \begin{cases} -\lambda + z_2 + w_2 & \text{if } \lambda \leq w_2 + z_2 \\ 0 & \text{if } 0 \leq w_2 + z_2 \leq \lambda \\ i\lambda_2 & \text{if } -ic \leq z_2 + w_2 \leq -ic + \lambda_2,\ i \in \mathbb{Z}_+ \\ -(i-1)c - z_2 - w_2 + (i-1)\lambda_2 & \text{if } -ic + \lambda_2 \leq z_2 + w_2 \leq -(i-1)c,\ i \in \mathbb{Z}_+. \end{cases}$$

The exact lifting function $\bar{f}_2$ is not superadditive in $\mathbb{R}^2$, but we can use the superadditive valid lifting function $\bar{\psi}(z_2, w_2) = -w_2 + g_{c-\lambda,c}(z_2 + w_2)$ where $g_{c-\lambda,c}$ is the superadditive function of Proposition 4. Figure 7 shows $g_{c-\lambda,c}(z_2 + w_2)$ and $\bar{f}_2(z_2, w_2) + w_2$. The proof that $\bar{\psi}$ is superadditive is analogous to the proof of Proposition 6.

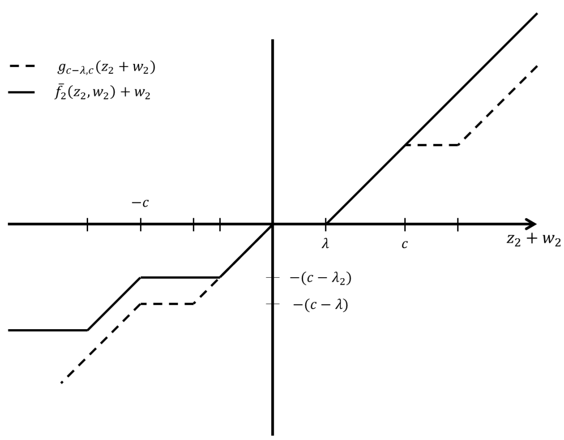

Figure 7: Functions $g_{c-\lambda,c}(z_2 + w_2)$ and $\bar{f}_2(z_2, w_2) + w_2$.

Using $\bar{\psi}$, we get the valid inequality

$$y(S_1^+ \cup S_2^+) + \sum_{j \in S_2^+} (c-\lambda)(1-x_j) + \sum_{j \in N_2^+ \setminus S_2^+} (y_j - (c-\lambda)x_j)^+ - \sum_{j \in N_2^-} \min\{y_j, \lambda x_j\}$$
$$+ \sum_{j \in S_{12}} (c-\lambda_2)(1-x_j) - \sum_{j \in N_{12} \setminus S_{12}} \min\{(c-\lambda_2)x_j, y_j\} + \sum_{j \in N_{21}} \max\{0, y_j - \lambda_2 x_j\} \leq d_{12}. \tag{26}$$

Inequality (26) still assumes $x_j = 1$ for $j \in S_1^+$ and $y_j = 0$ for $j \in (N_1^+ \setminus S_1^+) \cup N_1^-$.

The lifting function associated with simultaneously lifting (26) with variables x_j for $j \in S_1^+$ and pairs (y_j, x_j) for $j \in (N_1^+ \setminus S_1^+) \cup N_1^-$ is given by

$$\begin{aligned}\bar{f}_1(z_1, w_1) = \quad & \min \Big\{ d_{12} - y(S_1^+ \cup S_2^+) - \sum_{j \in S_2^+} (c - \lambda)(1 - x_j) - \sum_{j \in N_2^+ \setminus S_2^+} (y_j - (c - \lambda)x_j)^+ \\ & + \sum_{j \in N_2^-} \min\{y_j, \lambda x_j\} - \sum_{j \in S_{12}} (c - \lambda_2)(1 - x_j) + \sum_{j \in N_{12} \setminus S_{12}} \min\{(c - \lambda_2)x_j, y_j\} \\ & - \sum_{j \in N_{21}} \max\{0, y_j - \lambda_2 x_j\} \Big\} \\ \text{s.t. } & y(S_1^+) - y(N_{12}) + y(N_{21}) \le d_1 + w_1 \\ & y(N_2^+) - y(N_2^-) + y(N_{12}) - y(N_{21}) \le d_2 \\ & y(S_1^+) \le d_1 + \lambda_1 - z_1 \\ & y(S_2^+) \le d_2 + \lambda - \lambda_1 \\ & y(S_{12}) \le \lambda_1 + \lambda_2 - \lambda \\ & 0 \le y_j \le x_j, x_j \in \{0, 1\} \quad j \in N_{12} \cup N_{21} \cup N_2^+ \cup N_2^-,\end{aligned}$$

where z_1 is a nonnegative multiple of the capacity and stands for the capacity closed on S_1^+, $w_1 \ge 0$ stands for $y(N_1^-)$ and $w_1 < 0$ stands for $y(N_1^+ \setminus S_1^+)$.

Let

$$\begin{aligned}\gamma_{12}(a) = \quad & \min \Big\{ - \sum_{j \in S_{12}} (c - \lambda_2)(1 - x_j) + \sum_{j \in N_{12} \setminus S_{12}} \min\{(c - \lambda_2)x_j, y_j\} \\ & - \sum_{j \in N_{21}} \max\{0, y_j - \lambda_2 x_j\} \Big\} \\ \text{s.t. } & y(N_{12}) - y(N_{21}) = a\end{aligned}$$

be the contribution of the arcs in $N_{12} \cup N_{21}$ to the objective of the lifting function, given that the flow on these arcs is a. Note that for $\lambda_1 - \lambda \le a \le \lambda_1 + \lambda_2 - \lambda$ there exists an optimal solution where $y(S_{12}) = a$, $x(N_{21} \cup N_{12} \setminus S_{12}) = 0$, and $\gamma_{12}(a) = 0$. If $a > \lambda_1 + \lambda_2 - \lambda$ then we need to sequentially open arcs in $N_{12} \setminus S_{12}$, and for $a < \lambda_1 - \lambda$ it is optimal to sequentially open arcs in N_{21} or close arcs in S_{12}. We find that

$$\gamma_{12}(a) = \begin{cases} i(c - \lambda_2) & \text{if } \lambda_1 - \lambda + ic \le a \le \lambda_1 - \lambda + ic + \lambda_2,\ i \in \mathbb{Z} \\ a - \lambda_1 + \lambda - ic - \lambda_2 + i(c - \lambda_2) & \text{if } \lambda_1 - \lambda + ic + \lambda_2 \le a \le \lambda_1 - \lambda + (i+1)c,\ i \in \mathbb{Z}. \end{cases}$$

Moreover let

$$\gamma_2(a) = \quad \min\Big\{ -y(S_2^+) - \sum_{j\in S_2^+}(c-\lambda)(1-x_j) - \sum_{j\in N_2^+\setminus S_2^+}(y_j-(c-\lambda)x_j)^+ \\ + \sum_{j\in N_2^-}\min\{y_j,\lambda x_j\}\Big\} \\ \text{s.t. } y(N_2^+) - y(N_2^-) = a$$

be the contribution of the arcs in $N_2^+ \cup N_2^-$ to the objective of the lifting function, given that the flow on these arcs is a. Note that for $d_2 - \lambda_1 \le a \le d_2 - \lambda_1 + \lambda$ there exists an optimal solution where $y(S_2^+) = a$, $x(N_2^- \cup N_2^+ \setminus S_2^+) = 0$, and $\gamma_2(a) = -a$. If $a > d_2 - \lambda_1 + \lambda$ then we need to sequentially open arcs in $N_2^+ \setminus S_2^+$, and for $a < d_2 - \lambda_1$ it is optimal to sequentially open arcs in N_2^- or close arcs in S_2^+. Therefore

$$\gamma_2(a) = -(d_2-\lambda_1) + \begin{cases} -(a - d_2 + \lambda_1 - ic) - i\lambda & \text{if } d_2 - \lambda_1 + ic \le a \le d_2 - \lambda_1 + ic + \lambda,\ i \in \mathbb{Z} \\ -(i+1)\lambda & \text{if } d_2 - \lambda_1 + ic + \lambda \le a \le d_2 - \lambda_1 + (i+1)c,\ i \in \mathbb{Z}. \end{cases}$$

Now in the lifting problem, for a fixed value of $y(S_1^+) = y$, an optimal solution exists where $y(N_{12}) - y(N_{21})$ is as low as possible ($y(N_{12}) - y(N_{21}) = y - d_1 - w_1$) and $y(N_2^+) - y(N_2^-)$ is as high as possible ($y(N_2^+) - y(N_2^-) = d_{12} + w_1 - y$). In this case,

$$\begin{aligned} \bar{f}_1(z_1,w_1) &= d_{12} + \min_{0\le y\le d_1+\lambda_1-z_1} \{-y + \gamma_{12}(y - d_1 - w_1) + \gamma_2(d_{12} + w_1 - y)\} \\ &= d_{12} + \min_{0\le y\le d_1+\lambda_1-z_1} \begin{cases} i(\lambda-\lambda_2) - d_{12} - w_1 & \text{if } \lambda_2 - \lambda + (i-1)c \le y - d_1 - w_1 - \lambda_1 \le ic \\ -y + ic + i(\lambda - \lambda_2) - d_2 + \lambda_1 & \text{if } ic \le y - d_1 - w_1 - \lambda_1 \le ic + \lambda_2 - \lambda, \end{cases} \end{aligned}$$

where $i \in \mathbb{Z}$. The inner function is nonincreasing in y and, therefore, the minimum is attained at $y = d_1 + \lambda_1 - z_1$. We get

$$\bar{f}_1(z_1,w_1) = -w_1 + \begin{cases} i(\lambda_2 - \lambda) & \text{if } ic \le z_1 + w_1 \le \lambda - \lambda_2 + (i+1)c \\ z_1 + w_1 - ic + i(\lambda_2 - \lambda) & \text{if } ic + \lambda - \lambda_2 \le z_1 + w_1 \le ic, \end{cases}$$

which is of the form $-w_1 + g_{\lambda_2-\lambda,c}(z_1 + w_1)$ and is superadditive in $\mathbb{R}^2$. Using $\bar{f}_1$, we get the three-partition flow cover inequalities

$$\begin{aligned} & y(S_1^+ \cup S_2^+) + \sum_{j\in S_1^+}(\lambda_2-\lambda)(1-x_j) + \sum_{j\in N_1^+\setminus S_1^+}(y_j - (\lambda_2-\lambda)x_j)^+ - \sum_{j\in N_1^-}\min\{y_j,(c-(\lambda_2-\lambda))x_j\} \\ & + \sum_{j\in S_2^+}(c-\lambda)(1-x_j) + \sum_{j\in N_2^+\setminus S_2^+}(y_j-(c-\lambda)x_j)^+ - \sum_{j\in N_2^-}\min\{y_j,\lambda x_j\} \\ & + \sum_{j\in S_{12}}(c-\lambda_2)(1-x_j) + \sum_{j\in N_{21}}\max\{0, y_j - \lambda_2 x_j\} - \sum_{j\in N_{12}\setminus S_{12}}\min\{y_j,(c-\lambda_2)x_j\} \le d_{12}. \end{aligned} \tag{27}$$