Title

Submodular Inequalities for the Path Structures of the Capacitated Fixed-Charge Network Flow Problems

Author

Tezel, Birce

Publication Date

2017

Peer reviewed|Thesis/dissertation

Submodular Inequalities for the Path Structures of the Capacitated Fixed-Charge Network Flow Problems

By

Birce Tezel

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering – Industrial Engineering and Operations Research

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Alper Atamtürk, Chair
Professor Philip Kaminsky
Professor Satish Rao

Fall 2017

Abstract

Submodular Inequalities for the Path Structures of the Capacitated Fixed-Charge Network Flow Problems

by

Birce Tezel

Doctor of Philosophy in Industrial Engineering and Operations Research

University of California, Berkeley

Professor Alper Atamtürk, Chair

Capacitated fixed-charge network flow problems (CFCNF) are used to model a variety of problems in telecommunication, facility location, production planning and supply chain management. We model CFCNF as a linear mixed-integer program and study the polyhedral structure of various path networks.

We investigate capacitated path substructures and derive strong and easy-to-compute path cover and path pack inequalities. These inequalities are based on an explicit characterization of the submodular inequalities through a fast computation of parametric minimum cuts on a path, and they generalize the well-known flow cover and flow pack inequalities for the single-node relaxations of fixed-charge flow models. Computational results demonstrate the effectiveness of the inequalities when used as cuts in a branch-and-cut algorithm.

Moreover, we consider single item lot-sizing problems with backlogging and inventory bounds and fixed costs (LSBIB). Using the underlying fixed-charge network structure of LSBIB, we derive explicit path pack inequalities that are proposed in this thesis. These inequalities are generalizations of the valid inequalities proposed by Atamtürk and Küçükyavuz (2005) for lot-sizing problems under the existence of backlogging. Furthermore, we propose extensions of these inequalities where the binary variables for inventory and backlogging are lifted. Finally, we present computational results suggest that show the effectiveness of both path pack and extended path pack inequalities when used in a branch-and-cut algorithm.

Dedication

This thesis is dedicated to my parents who loved and supported me through it all.

Acknowledgements

I am thankful to my advisor, Alper Atamtürk for his guidance. His research vision always directed me to the right things at the right time and my thesis would not have come together without him. I am also very grateful to my co-author Simge Küçükyavuz for all her feedback and contributions to our paper together. Moreover, I would like to thank my committee members Philip Kaminsky and Satish Rao for their feedback and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Given a graph with demand/supply nodes and arcs with upper bounds on flow, the capacitated fixed-charge network flow problem (CFCNF) aims to find a subset of arcs such that the flow balance on each node is conserved and flow capacities are not exceeded. In this thesis, we model CFCNF as a mixed-integer program. We propose valid inequalities and analyze their strength both theoretically and computationally.

The research in this thesis has roots in the seminal paper by Wolsey (1989). In this paper, the author proposes submodular inequalities (see Section 1.4) that are valid for any CFCNF. While submodular inequalities are quite general and make no assumptions on the network structure, computing their coefficients require solving many maximum flow problems. Due to their implicit coefficients, using them in a branch-and-cut algorithm requires a significant amount of computational effort. As a result, the strength of these inequalities have not been analyzed for general network structures.

In this thesis, we focus on path networks and propose a linear-time algorithm to express submodular inequalities explicitly. All of the chapters are concluded with an extensive computational study. In Chapter 2, we study on simple paths and propose path pack inequalities. Then, we generalize path pack inequalities by a simultaneous lifting procedure. In Chapter 3, we generalize the simple path by adding arcs $(j+1, j)$ to the consecutive nodes in the path. In addition to the path cover inequalities for these path structures, we propose a second class of submodular inequalities and refer to them as path pack inequalities. We show that path cover and path pack inequalities reduce to flow cover and flow pack inequalities when the underlying network consists of a single node. In Chapter 4, we extend the research of Atamtürk and Küçükyavuz (2005) on lot-sizing problems with inventory bounds by considering backlogging arcs. Using the path structure of this problem, we propose eleven classes of path pack inequalities parametrically. Then, we extend these inequalities by incorporating the binary fixed-charge variables associated with inventory and backlogging arcs. The computational results prove that path cover and path pack inequalities and their extensions are quite useful while solving fixed-charge network flow problems.

In the remainder of this chapter, we give a brief background information on polyhedral analysis using the definitions from Wolsey and Nemhauser (1999) and Wolsey (1998). Following the basics in Section 1.1, we introduce a mixed-integer programming formulation for the capacitated fixed-charge networks in Section 1.2. In Sections 1.3 and 1.4, we introduce the existing flow cover, flow pack and submodular inequalities which are referred to often in this thesis. In Section 1.5, we go over the lifting process of valid inequalities.

1.1 Polyhedral analysis terminology

We first start by describing *mixed-integer programming* (MIP) as

$$\max\{cx + hy : \ Ax + Gy \leq b, \ x \in \mathbb{Z}^n, \ y \in \mathbb{R}^p\}$$

where $\mathbb{Z}^n$ is the set of integer n-dimensional vectors and $\mathbb{R}^p$ is the set of real p-dimensional vectors. In a MIP A, G and b are the parameter set and x, y are the decision variables. The feasible set of MIP is defined by

$$S = \{x \in \mathbb{Z}^n, \ y \in \mathbb{R}^p \ : \ Ax + Gy \leq b\}$$

and we call the set

$$S_{LP} = \{x \in \mathbb{R}^n, \ y \in \mathbb{R}^p \ : \ Ax + Gy \leq b\}$$

the *linear programming (LP) relaxation* of the feasible set S.

Following definitions are directly cited from Wolsey and Nemhauser (1999).

Definition *Convex hull* of S is the set of all points that are convex combinations of points in S.

Definition A *polyhedron* $P \subseteq \mathbb{R}^n$ is the set of points that satisfy a number of linear inequalities; that is, $P = \{x \in \mathbb{R}^n \ : \ Ax \leq b\}$, where (A, b) is an $m \times (n+1)$ matrix.

Definition $x \in P$ is an *extreme point* of P if there do not exist $x^1, x^2 \in P$, $x^1 \neq x^2$ such that $x = \frac{1}{2}x^1 + \frac{1}{2}x^2$.

Definition A set of points $x^1, \ldots, x^k \in \mathbb{R}^n$ is *affinely independent* if the unique solution of $\sum_{i=1}^k \alpha_i x^i = 0, \ \sum_{i=1}^k \alpha_i = 0$ is $\alpha_i = 0$ for $i = 1, \ldots, k$.

Definition A polyhedron P is of *dimension* k if the maximum number of affinely independent points in P is $k+1$ and it is represented by $\dim(P) = k$.

Definition A polyhedron $P \subseteq \mathbb{R}^n$ is *full-dimensional* if $\dim(P) = n$.

Definition The inequality $\pi x \leq \pi_0$ (or (π, π_0)) is called a *valid inequality* for P if it is satisfied by all points in P.

Definition If (π, π_0) is a valid inequality for P, and $F = \{x \in P : \ \pi x = \pi_0\}$, F is called a *face* of P.

Definition A face F of the polyhedron P is called a *facet* of P if $\dim(F) = \dim(P) - 1$. The inequality describing F is then a *facet-defining inequality* or a *facet*.

1.2 Capacitated fixed-charge network flow problems

Let $G = (N, A)$ be a directed graph with nodes N and arcs A and $|N| = n$, $|A| = a$. Let $E_j^+ = \{(i,j) \in A : i \in N\}$ and $E_j^- = \{(j,i) \in A : i \in N\}$ for all $j \in N$. Let the demand value at node $j \in N$ be d_j (where $d_j < 0$ indicates j is a supply node). Define the binary variable x_t to be 1 if arc $t \in A$ is used and the variable y_t to be the flow through arc t, which has a capacity of c_t.

The mathematical programming representation of capacitated fixed-charge network problems can be represented by the following feasibility set:

$$
(\mathcal{P}_G) \qquad \begin{aligned}
&\sum_{t \in E_j^+} y_t - \sum_{t \in E_j^-} y_t \leq d_j, \quad j \in N, \\
&0 \leq y_t \leq c_t x_t, \quad t \in A, \\
&x_t \in \{0, 1\}, \quad t \in A.
\end{aligned}
$$

We represent the feasible set of the capacitated fixed charge network flow problem on underlying network G by $\mathcal{P}_G$.

1.3 Flow cover and flow pack inequalities

In this section, we introduce the well known flow cover inequalities of Padberg et al. (1985) and flow pack inequalities of Atamtürk (2001). Consider a single-node capacitated fixed charge network flow problem with the feasible set

$$
\mathcal{P} = \Big\{ y \in \mathbb{R}_+^n, x \in \mathbb{B}^n : \ \sum_{t \in E^+} y_t - \sum_{t \in E^-} y_t \leq d, \ y_t \leq c_t x_t \text{ for } t \in E^+ \cup E^- \Big\}.
$$

See Figure (1.1) for a graph representation of $\mathcal{P}$. Let $S^+ \subseteq E^+$, $S^- \subseteq E^-$ and $L^- \subseteq E^- \setminus S^-$. The set pair (S^+, S^-) is called a *flow cover* if $\sum_{t \in S^+} c_t - \sum_{t \in S^-} c_t = d + \lambda$ and $\lambda > 0$. Similarly, the pair (S^+, S^-) is called a *flow pack* if $\sum_{t \in S^+} c_t - \sum_{t \in S^-} c_t = d - \mu$ and $\mu > 0$.

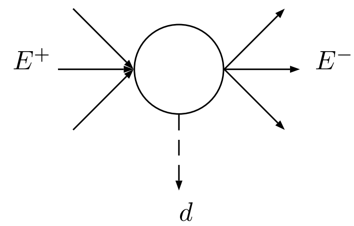


Figure 1.1: A single node network

Let (S^+, S^-) be a flow cover, then the inequality

$$\sum_{t \in S^+} \big(y_t + (c_t - \lambda)^+(1 - x_t)\big) \le d + \sum_{t \in S^-} c_t + \sum_{t \in L^-} \lambda x_t + \sum_{t \in E^- \setminus (L^- \cup S^-)} y_t \tag{1.1}$$

is valid for $\mathcal{P}$. Inequality (1.1) is referred to as a *generalized flow cover* inequality. van Roy and Wolsey (1986) show that if $d > 0$, $\max_{t \in S^+} c_t > \lambda$ and $c_t > \lambda$ for $t \in L^-$ and $S^- = \emptyset$, then inequality (1.1) is a facet of the convex hull of $\mathcal{P}$.

Let (S^+, S^-) be a flow pack and let $L^+ \subseteq E^+ \setminus S^+$, then the inequality

$$\sum_{t \in S^+} y_t + \sum_{t \in L^+} \big(y_t - \min\{c_t, \mu\} x_t\big) \le c_t + \sum_{t \in S^-} (c_t - \mu)(1 - x_t) + \sum_{t \in E^- \setminus S^-} y_t \tag{1.2}$$

is valid for $\mathcal{P}$ and is referred to as a *flow pack* inequality.

1.4 Submodular inequalities

Definition A set function v on a set $S = \{1, \dots, s\}$ is *submodular* if

$$v(A) + v(B) \ge v(A \cup B) + v(A \cap B)$$

for all $A, B \subseteq S$.

Proposition 1.1. Let $\rho_j(A) = v(A \cup \{j\}) - v(A)$. The function v is submodular if and only if

$$\rho_j(A) \ge \rho_j(B)$$

for all $A \subseteq B \subseteq S \setminus \{j\}$.

Next, we summarize some of the main results in Wolsey (1989). Let $G(V, A)$ be a capacitated fixed-charge network and let $N \subseteq V$ be a subset of nodes and define $E^+ = \{(i,j) \in A : i \notin N, j \in N\}$, $E^- = \{(i,j) \in A : i \in N, j \notin N\}$ and let $E := E^+ \cup E^-$.

Let $S^+ \subseteq E^+$ and $L^- \subseteq E^-$. Define the set function $v(S^+, L^-)$ (or $v(C)$ in short notation) by the following optimization problem:

$$\begin{aligned} v(S^+, L^-) = \max \quad & \sum_{t \in E} a_t y_t \\ \text{s.t.} \quad & \sum_{t \in E_j^+} y_t - \sum_{t \in E_j^-} y_t \le d_j, \quad j \in N, \\ & 0 \le y_t \le c_t, \quad t \in E, \\ & y_t = 0, \quad t \in (N^+ \setminus S^+) \cup L^-, \end{aligned}$$

where $a_t \in \{0, 1\}$ for $t \in E^+$ and $a_t \in \{0, -1\}$ for $t \in E^-$. Let $\rho_t(S^+, L^-)$ be

$$\rho_t(S^+ \setminus \{t\}, L^-) = v(S^+, L^-) - v(S^+ \setminus \{t\}, L^-)$$

for $t \in S^+$ and

$$\rho_t(S^+, L^- \setminus \{t\}) = v(S^+, L^-) - v(S^+, L^- \setminus \{t\})$$

for $t \in L^-$. For convenience, we represent the sets (S^+, L^-) as a single let:

$$C := S^+ \cup L^-$$

and

$$\rho_t(C) := v(C \cup \{t\}) - v(C).$$

Wolsey (1989) proves that $v(S^+, L^-)$ is submodular on the set E and shows that the inequalities

$$\begin{aligned} \sum_{t \in E} a_t y_t \le v(C) - \sum_{t \in S^+} \rho_t(E \setminus \{t\})(1 - x_t) - \sum_{t \in L^-} \rho_t(E \setminus \{t\}) x_t \\ + \sum_{t \in E^+ \setminus S^+} \rho_t(C) x_t + \sum_{t \in E^- \setminus L^-} \rho_t(C)(1 - x_t) \end{aligned}$$

and

$$\begin{aligned} \sum_{t \in E} a_t y_t \le v(C) - \sum_{t \in S^+} \rho_t(C \setminus \{t\})(1 - x_t) - \sum_{t \in L^-} \rho_t(C \setminus \{t\}) x_t \\ + \sum_{t \in N^+ \setminus S^+} \rho_t(\emptyset) x_t + \sum_{t \in E^- \setminus L^-} \rho_t(\emptyset)(1 - x_t) \end{aligned}$$

are valid for the capacitated fixed-charge network feasible set $\mathcal{P}_G$.

1.5 Lifting valid inequalities

Given a valid inequality (π, π_0) for polyhedron P, lifting is the procedure of extending this inequality to higher dimensions. The notion of lifting was first introduced by Gomory (1969) and then generalized and formally described in Wolsey (1976), Zemel (1978) and Balas and Zemel (1978). Lifting proved to be very useful in solving integer programs more efficiently; however, in most cases, the coefficients of the lifted valid inequalities depend on the sequence of the variables being lifted. Obtaining the lifting coefficients require solving a separate optimization problem. As a result, sequence dependence creates a computational burden for the lifting procedure. For binary integer programs, Wolsey (1977) showed that if the lifting function is super-additive then the coefficients are independent from the sequence of lifting. This result was generalized for mixed binary integer programs by Gu (1994) and Gu et al. (2000), and for general mixed integer programs by Atamtürk (2004).

Gu et al. (1999) provide the analytical form of the lifting function of flow cover inequalities, introduce a valid super additive lifting function and present computational results to show the effectiveness of lifting. In the next section, we introduce a procedure to obtain a valid super additive lifting function for submodular path inequalities. Next, we describe the general lifting procedure for mixed binary integer programs.

Let $X = \{x \in \mathbb{B}^p,\ y \in \mathbb{R}^q :\ Ax + Gy \leq d;\ l_j \leq y_j \leq u_j,\ j \in J\}$ be the feasible set of a mixed binary integer program where $\mathbb{B} = \{0, 1\}$ and index sets I and J contain binary and continuous variables respectively. Size of d is $m \times 1$, $A = \{a_{ij}\}_{i=1,\ldots,m,\ j=1,\ldots,p}$ and $G = {g_{ij}}_{i=1,\ldots,m,\ j=1,\ldots,q}$. Let C^i, $i = 0, \ldots, t$ be a partition of the variables $I \cup J$. We define a restriction of X by setting a subset of the variables to one of their bound values b_j. Note that, $b_j \in \{0, 1\}$ for $j \in I$ and $b_j \in \{l_j, u_j\}$ for $j \in J$. Let the restricted feasible set associated with partition j be represented as X^j.

$$X^0 = \{(x, y) \in X :\ x_j = b_j,\ j \in I \setminus C^0,\ y_j = b_j,\ j \in J \setminus C^0\}.$$

Let

$$\sum_{j \in I \cap C^0} \alpha_j x_j + \sum_{j \in J \cap C^0} \beta_j y_j \leq \pi \tag{1.3}$$

be a valid inequality for X^0. We would like to extend this valid inequality by adding the variables in $(I \cup J) \setminus C^0$ and obtain an inequality of the form

$$\sum_{j \in I \cap C^0} \alpha_j x_j + \sum_{j \in J \cap C^0} \beta_j y_j + \sum_{j \in I \setminus C^0} \alpha_j (x_j - b_j^x) + \sum_{j \in J \setminus C^0} \beta_j (y_j - b_j^y) \leq \pi.$$

that is valid for X.

In order to reach this inequality, lifting is carried out by steps. At each step, a partition of variables C^i are lifted simultaneously. Unfortunately, the resulting coefficients are usually dependent on the sequence in which these partitions are lifted. Without loss of generality, we assume variables in C^i are lifted in ith order. Then the valid inequality obtained after ith lifting sequence is valid for the restricted feasible set

$$X^i = \{(x,y) \in X : x_j = b_j^x,\ j \in I \setminus (\cup_{k=0}^{i} C^k),\ y_j = b_j^y,\ j \in J \setminus (\cup_{k=0}^{i} C^k)\}.$$

Given the valid inequality for set X^{i-1}:

$$\sum_{j \in I \cap C^0} \alpha_j x_j + \sum_{j \in J \cap C^0} \beta_j y_j + \sum_{j \in I \cap (\cup_{k=1}^{i-1} C^k)} \alpha_j (x_j - b_j^x) + \sum_{j \in J \cap (\cup_{k=1}^{i-1} C^k)} \beta_j (y_j - b_j^y) \le \pi$$

lifting problem for set C^i aims to find coefficients $(\alpha_j, \beta_j),\ j \in C^i$ such that

$$\sum_{j \in I \cap C^0} \alpha_j x_j + \sum_{j \in J \cap C^0} \beta_j y_j + \sum_{j \in I \cap (\cup_{k=1}^{i} C^k)} \alpha_j (x_j - b_j^x) + \sum_{j \in J \cap (\cup_{k=1}^{i} C^k)} \beta_j (y_j - b_j^y) \le \pi \quad (1.4)$$

is valid for X^i. We also define the ith *lifting function* associated with inequality as:

$$\begin{aligned} f_i(z) = \min \quad & \pi - \Big[\sum_{j \in I \cap C^0} \alpha_j x_j + \sum_{j \in J \cap C^0} \beta_j y_j \\ & \quad + \sum_{j \in I \cap (\cup_{k=1}^{i-1} C^k)} \alpha_j (x_j - b_j^x) + \sum_{j \in J \cap (\cup_{k=1}^{i-1} C^k)} \beta_j (y_j - b_j^y) \Big] \\ \text{s.t.} \quad & \sum_{j \in I \cap (\cup_{k=0}^{i-1} C^k)} a_{lj} (x_j - b_j^x) + \sum_{j \in J \cap (\cup_{k=0}^{i-1} C^k)} g_{lj} (y_j - b_j^y) \le d_l' - z_l, \\ & \qquad l = 1, \dots, m, \\ & l_j \le y_j \le u_j, \quad j \in J, \\ & x_j \in \{0,1\}, \quad j \in I, \\ & x_j = b_j^x, \quad j \in I \setminus (\cup_{k=0}^{i-1} C^k), \\ & y_j = b_j^y, \quad j \in J \setminus (\cup_{k=0}^{i-1} C^k). \end{aligned}$$

where $d_l' = d_l - \sum_{j \in I \setminus C^0} a_{lj} b_j^x - \sum_{j \in J \setminus C^0} g_{lj} b_j^y$ and $z \in Z^i$ with

$$Z^i = \Bigg\{ z \in \mathbb{R}^m :\ \exists (x,y) \in X^i :\ \sum_{j \in I \cap C^i} a_{lj}(x_j - b_j) + \sum_{j \in J \cap C^i} g_{lj}(y_j - b_j) = z_l \quad l = 1, \dots, m$$
$$\sum_{j \in I \cap (\cup_{k=0}^{i-1} C^k)} a_{lj} (x_j - b_j^x) + \sum_{j \in J \cap (\cup_{k=0}^{i-1} C^k)} g_{lj} (y_j - b_j^y) \le d_l' - z_l \quad l = 1, \dots, m \Bigg\}.$$

Let $Z = Z^1 \times \ldots \times Z^t$.

Proposition 1.2 (Gu et al. (1999))**.** Inequality (1.4) is valid for X^i for a choice of $(\alpha_j, \beta_j),\ j \in C^i$ if and only if $h_i(z) \leq f_i(z)$ for any $z \in Z^i$ where

$$\begin{aligned} h_i(z) = \max \quad & \sum_{j \in I \cap C^i} \alpha_j (x_j - b_j) + \sum_{j \in J \cap C^i} \beta_j (y_j - b_j) \\ \text{s.t.} \quad & \sum_{j \in I \cap C^i} a_{lj} (x_j - b_j) + \sum_{j \in J \cap C^i} g_{lj} (y_j - b_j) = z_l \quad l = 1, \ldots, m \\ & x_j \in \{0,1\} \quad j \in I \cap C^i \\ & l_j \leq y_j \leq u_j \quad j \in J \cap C^i. \end{aligned}$$

Definition *Lifting function* $f(z)$ of the valid inequality (1.3) is defined as $f(z) = f_1(z)$ for all $z \in Z$.

Definition Lifting of valid inequality (1.3) is *sequence independent* if $f(z) = f_i(z)$ for $i = 2, \ldots, t$.

Definition A function f is super-additive on Z if

$$f(z_1) + f(z_2) \leq f(z_1 + z_2)$$

for all $z_1, z_2, z_1 + z_2 \in Z$.

It is shown in the literature that, if $f(z)$ is super additive on Z, then lifting is sequence independent. Unfortunately, super-additivity of $f(z)$ is very uncommon.

Definition Function g is called *valid super additive lifting function* if it is super additive and $g(z) \leq f(z)$ for all $z \in Z$.

Proposition 1.3. If g is a valid super additive lifting function and (α_j, β_j) for $j \in C^i$ ensure that $h_i(z) \leq g(z)$ for all $z \in Z$ and $i = 1, \ldots, t$, then inequality (1.4) is valid for X.

Chapter 2

Submodular Path Inequalities for the Capacitated Fixed-Charge Network Flow Problem

Given a directed graph with demand and supply on the nodes, and capacity, fixed and variable cost of flow on the arcs, the capacitated fixed-charge network flow (CFNF) 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. There are numerous polyhedral studies on the fixed-charge network flow problem. However, few give explicit valid inequalities that simultaneously make use of the path substructures of the network as well as the arc capacities, which is the goal of the current chapter.

For the *uncapacitated* fixed-charge network flow problem, van Roy and Wolsey (1985) give flow path inequalities that are based on path substructures. Rardin and Wolsey (1993) introduce a new family of dicut inequalities and show that they describe the projection of extended multicommodity formulation onto the original variables of fixed-charge network flow problem. Ortega and Wolsey (2003) present a computational study on the performance of path and cut-set (dicut) inequalities. For the *capacitated* fixed-charge network flow problem, almost all known valid inequalities are based on single-node relaxations. Padberg et al. (1985), van Roy and Wolsey (1986) and Gu et al. (1999) give flow cover, generalized flow cover and lifted flow cover inequalities. Stallaert (1997) introduces a complementary class of flow cover inequalities; Atamtürk (2001) describes lifted flow pack inequalities. Both uncapacitated path inequalities and capacitated flow cover inequalities are highly valuable in solving a host of practical problems and are part of the suite of cutting planes implemented in modern mixed-integer programming solvers.

The path structure arises naturally in network models of the lot-sizing problem. Atamtürk and Muñoz (2004) introduce valid inequalities for the capacitated lot-sizing problems with infinite inventory capacities. Atamtürk and Küçükyavuz (2005) give valid inequalities for the lot-sizing problems with finite inventory and infinite production capacities. Van Vyve (2013) introduces valid inequalities for uncapacitated fixed charge transportation problems. Van Vyve and Ortega (2004) and Gade and Küçükyavuz (2011) give valid inequalities and extended formulations for uncapacitated lot-sizing with fixed charges on stocks.

In this chapter we consider a generic path relaxation, with supply and/or demand nodes and capacities on all incoming and outgoing arcs. First, by exploiting the path substructure of the network, we provide an explicit description of the submodular inequalities introduced by Wolsey (1998). An important consequence of the explicit derivation is that the coefficients of these *submodular path inequalities* can be computed efficiently. In particular, we show that *all* coefficients of an inequality can be computed by solving max-flow/min-cut problems over the path in linear time. For a path with a single node, the inequalities reduce to the flow cover inequalities introduced by Padberg et al. (1985). We give sufficient and necessary facet-defining conditions. Then we generalize the inequalities further by superadditive lifting using an approximate multi-dimensional lifting function. Finally, we demonstrate the effectiveness of the proposed inequalities when used as cuts in a branch-and-cut algorithm.

Outline

The remainder of this chapter is organized as follows: In Section 2.1, we describe the CFNF problem on a path, its formulation and the assumptions we make. In Section 2.2, we review the submodular inequalities of Wolsey (1998) and introduce the submodular path inequalities. In Section 2.3, we generalize the submodular path inequalities by superadditive lifting. In Section 2.4, we present some computational results depicting the effectiveness of the valid inequalities proposed.

2.1 Capacitated fixed-charge network flow on a path

Let $G = (N', A)$ be a directed path with nodes N' partitioned into sets N and $\{s_N, t_N\}$, where s_N and t_N are source and sink nodes and V are the path nodes (see Figure 2.1). Let $N = \{k, \ldots, \ell\}$ and the *path arcs* be $I := \{(k, k+1), (k+1, k+2), \ldots, (\ell-1, \ell)\}$. Define $a := |A|$ and $n := |N|$. Without loss of generality, we assume that node k has an incoming path arc $(k-1, k)$ and node ℓ has an outgoing path arc $(\ell, \ell+1)$ with zero capacity (represented by dotted lines in Figure 2.1).

For each node $j \in N$, let $E_j^+ = \{(s_N, i) \in A : i = j\}$ and $E_j^- = \{(i, t_N) \in A : i = j\}$. Let the union of such sets be denoted by $E^+ := \{(s_N, i) \in A : i \in N\}$ and $E^- := \{(i, t_N) \in A : i \in N\}$. Finally, let $E := E^+ \cup E^-$ with $e := |E|$.

In this chapter, we consider simple directed paths where sets E_j^+ and E_j^- for all $j \in N$ are mutually exclusive. In other words, the arcs in I form the unique directed

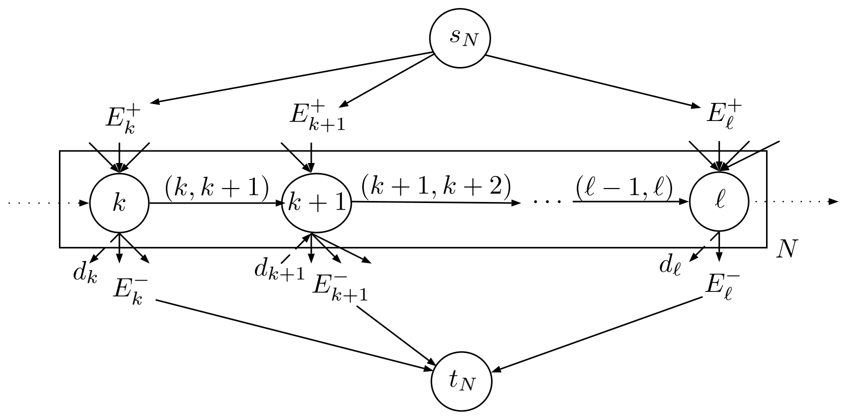


Figure 2.1: Nodes and arcs of a path set.

path from node k to node ℓ. For any graph, if need be, one can construct such a relaxation by duplicating arcs $t \in E_i^+ \cap E_j^-$ as $t^+ \in E_i^+$ and $t^- \in E_j^-$ for $i, j \in N$.

Let the demand at node $j \in N$ be d_j. We call a node $j \in N$ a demand node if $d_j \geq 0$ and a supply node if $d_j < 0$. Define the binary variable x_t to be 1 if arc $t \in E$ is open, 0 otherwise; and the variable y_t to be the flow through arc t, with capacity of c_t. We refer to the flow of path arc $(j, j+1)$ as i_j, and denote the corresponding capacity by u_j. We let $[k,\ell] := \{t \in \mathbb{Z} : k \leq t \leq \ell\}$ and let $d(T) = \sum_{j \in T} d_j$ for a given $T \subseteq V$. For notational convenience, we refer to $d([k,\ell])$ as $d_{k\ell}$. Let $c(S) = \sum_{t\in S} c_t$, $y(S) = \sum_{t \in S} y_t$, $x(S) = \sum_{t\in S} x_t$ and $(a)^+ = \max\{0, a\}$. Moreover, let us denote the convex hull of a formulation P as $\text{conv}(P)$.

Let the *path-set* relaxation $(\mathcal{P}_G)$ of CFNF problem on G be

$$
(\mathcal{P}_G) \qquad
\begin{aligned}
& i_{j-1} + y(E_j^+) - y(E_j^-) - i_j \leq d_j, && j \in N, && (2.1)\\
& 0 \leq i_j \leq u_j, && j \in N, && (2.2)\\
& 0 \leq y_t \leq c_t x_t, && t \in E, && (2.3)\\
& x_t \in \{0,1\}, && t \in E. && (2.4)
\end{aligned}
$$

In order to avoid trivial cases we make the following assumptions:

(A.1) $u_j > 0$ for $j \in I$ (otherwise, $i_t = 0$ and arc t can be removed from the graph),

(A.2) $c_t > 0$ for $t \in E$ (otherwise $y_t = 0$ and arc t can be removed from the graph and the problem decomposes into two subproblems defined on two simple paths),

(A.3) for all $t \in E$, $G' = (V, A \setminus \{t\})$ with corresponding feasible set $\mathcal{P}_{G'} \neq \emptyset$ (otherwise $x_t = 1$).

It follows from the assumptions that the dimension of $\mathcal{P}_G$ is $e + a$ (i.e., $\mathcal{P}_G$ is full dimensional).

2.2 Submodular path inequalities

In this section, we review the submodular inequalities of Wolsey (1989) and derive their explicit form for the path networks. Let $S^+ \subseteq E^+$ and $L^- \subseteq E^-$. Sets S^+ and L^- can be partitioned into sets S_j^+ and L_j^- for $j = k, \ldots, \ell$ where $S_j^+ = S^+ \cap E_j^+$ and $L_j^- = L^- \cap E_j^-$.

Now, we consider the following formulation introduced by Wolsey (1989) for the path structure G:

$$v(S^+, L^-) = \max \quad \sum_{t \in E} a_t y_t \tag{2.5}$$

$$\text{s.t.} \quad i_{j-1} + y(E_j^+) - y(E_j^-) - i_j \leq d_j, \quad j \in N, \tag{2.6}$$

$$0 \leq i_j \leq u_j, \quad j \in I, \tag{2.7}$$

$$(\mathcal{P}') \qquad 0 \leq y_t \leq c_t, \quad t \in E, \tag{2.8}$$

$$y_t = 0, \quad t \in (E^+ \setminus S^+) \cup L^-, \tag{2.9}$$

where $a_t \in \{0, 1\}$ for $t \in E^+$ and $a_t \in \{0, -1\}$ for $t \in E^-$. Define the sets $K^+ := \{t \in E^+ : a_t = 1\}$ and $K^- := \{t \in E^- : a_t = 0\}$. In this section, we choose $K^+ = S^+$ and $K^- = L^-$. We make the following additional assumption:

(A.4) The sets S^+ and L^- are selected such that the formulation $\mathcal{P}'$ is feasible.

Proposition 2.1 (Wolsey (1989))**.** The set function $v : 2^{|E^+|} \times 2^{|E^-|} \to \mathbb{R}$ defined through (2.5)-(2.9) is submodular.

It follows from Corollary 8 of Wolsey (1989) that the *submodular inequality*

$$\begin{aligned} \textstyle\sum_{t \in S^+}(y_t + \rho_t(S^+ \setminus \{t\}, L^-)(1 - x_t)) + \sum_{t \in L^-} \rho_t(S^+, L^- \setminus \{t\}) x_t \\ \leq v(S^+, L^-) + \textstyle\sum_{t \in E^- \setminus L^-} y_t, \end{aligned} \tag{2.10}$$

where

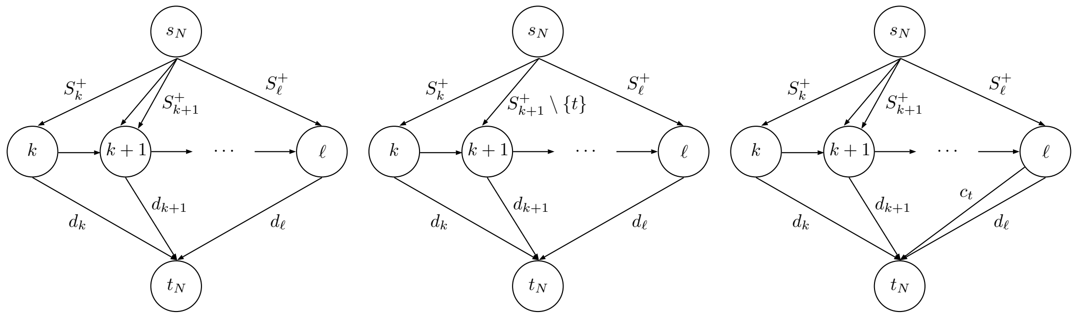
$$\rho_t(S^+ \setminus \{t\}, L^-) = v(S^+, L^-) - v(S^+ \setminus \{t\}, L^-), \quad t \in S^+$$

and

$$\rho_t(S^+, L^- \setminus \{t\}) = v(S^+, L^-) - v(S^+, L^- \setminus \{t\}), \quad t \in L^-$$

is valid for $\mathcal{P}_G$. Observe that inequality (2.10) requires computing $\rho_t(S^+ \setminus \{t\}, L^-)$ for $t \in S^+$ and $\rho_t(S^+, L^- \setminus \{t\})$ for $t \in L^-$, which can be done by solving $|S^+ \cup L^-| + 1$ optimization problems. Next, we investigate the structure of $\mathcal{P}'$ in order to obtain the explicit form of submodular inequalities.

Proposition 2.2. The optimization problem $\mathcal{P}'$ is equivalent to a maximum flow problem from source s_N to sink t_N.



(a) $v(S^+, L^-)$. (b) $v(S^+ \setminus \{t\}, L^-)$. (c) $v(S^+, L^- \setminus \{t\})$.

Figure 2.2: Equivalency of $\mathcal{P}'$ to the maximum-flow problem.

Proof. By Proposition A.1 and Remark A.1 in Appendix A, the decision variables y_t, for $t \in E^-$ such that $a_t = -1$ can be assumed to be zero in the optimization problem $\mathcal{P}'$. Then, the problem defined by (2.5)-(2.9) reduces to:

$$\max \big\{ y(S^+) \ : \ i_{j-1} + y(S_j^+) - y(L_j^-) - i_j = b_j, \ j \in N, \\ 0 \le b_j \le d_j, \ j \in N, \ (2.7) - (2.9) \big\},$$

where b_j represents the flow on the dummy demand arc of node j. □

There are numerous algorithms for solving maximum flow problems for general graphs in polynomial time. However, one can solve the maximum flow problem on path graphs in linear time. In Figure 2.2, we present three different arc sets that we solve the maximum flow on.

Definition If $v(S^+, L^-) = d_{k\ell}$, then S^+ is called a *path cover*.

Observation 1. Let S^+ be a path cover. A minimum cut associated with $v(S^+, L^-)$ is defined by the source and sink partitions $\{s_N, k, \ldots, \ell\}$ and $\{t_N\}$. Informally, this minimum cut passes below the path (see Figure 2.3a for a representation).

Obtaining an explicit form of inequality (2.10) requires finding the optimal objective value $v(S^+, L^-)$ as well as the values $v(S^+ \setminus \{t\}, L^-)$ for all $t \in S^+$ and $v(S^+, L^- \setminus \{t\})$ for all $t \in L^-$. By definition 2.2, $v(S^+, L^-)$ is equal to $d_{k\ell}$ if S^+ is a path cover. In the next lemma, we identify minimum cuts associated with values $v(S^+ \setminus \{t\}, L^-)$ for $t \in S^+$ and $v(S^+, L^- \setminus \{t\})$ for $t \in L^-$.

Lemma 2.1. Let S^+ be a path cover. At least one minimum cut associated with $v(S^+ \setminus \{t\}, L^-)$ for $t \in S^+$ and $v(S^+, L^- \setminus \{t\})$ is defined by one of the following source-sink partitions:

(i) $\{s_N, k, \ldots, \ell\}$ and $\{t_N\}$ (informally, minimum cut is the same as $v(S^+, L^-)$; see Figure 2.3a),

(ii) $\{s_N, k, \ldots, \mathtt{lb}\}$ and $\{\mathtt{lb}+1, \ldots, \ell, t_N\}$ for some $\mathtt{lb} \in [k-1, j-1]$ (informally, minimum cut goes above the path at most once through an arc $(\mathtt{lb}, \mathtt{lb}+1)$; see Figure 2.3b),

(iii) $\{s_N, k, \ldots, \mathtt{lb}, \mathtt{rb}, \ldots, \ell\}$ and $\{\mathtt{lb}+1, \ldots, \mathtt{rb}-1\}$ for some $\mathtt{rb} \in [j+1, \ell+1]$ (informally, if minimum cut goes above the path, then it goes below the path at most once through an arc $(\mathtt{rb}-1, \mathtt{rb})$; see Figure 2.3c).

Proof. For $t \in S^+$ observe that dropping an arc $t \in S_j^+$ may cause a decrease in the maximum flow from s_N to t_N (see the change from Figure 2.2a to Figure 2.2b) . If there is a positive decrease in the maximum flow, then at least one minimum cut has to pass above node j. Any cut that does not pass above node j has a value greater than or equal to $d_{k\ell}$ due to the assumption that the value of minimum cut before dropping arc t is equal to $d_{k\ell}$. If there is no decrease in the maximum flow, then minimum cut will remain unchanged with a value $d_{k\ell}$.

Now suppose that a minimum cut after dropping arc t goes above the path through more than one path arc. We refer to the nodes that are below this minimum cut as $[\mathtt{lb}_1+1, \mathtt{rb}_1-1]$ and $[\mathtt{lb}_2+1, \mathtt{rb}_2-1]$ for $\mathtt{rb}_2 > \mathtt{lb}_2 \geq \mathtt{rb}_1 > \mathtt{lb}_1$. Suppose $j \notin [\mathtt{lb}_p+1, \mathtt{rb}_p-1]$ for some $p = 1, 2$. For a minimum cut to go above the path at path arc $(\mathtt{lb}_p, \mathtt{lb}_p+1)$, it is required to have $u_{\mathtt{lb}_p} + \sum_{i \in [\mathtt{lb}_p+1, \mathtt{rb}_p-1]} c(S_i^+) < d_{\mathtt{lb}_p+1, \mathtt{rb}_p-1}$. However, the assumption of $v(S^+, L^-) = d_{k\ell}$ implies that $u_{\mathtt{lb}_p} + \sum_{i \in [\mathtt{lb}_p+1, \mathtt{rb}_p-1]} c(S_i^+) \geq d_{\mathtt{lb}_p+1, \mathtt{rb}_p-1}$. Therefore, we reach a contradiction.

The proof follows the same steps for $t \in L^-$ (see the change from Figure 2.2a to Figure 2.2c). □

Let λ_j represent the total excess flow capacity that can be sent through arcs in I and S^+ to node j after satisfying all the demands d_i, $i \in N$.

Definition Let S^+ be a path cover. The *excess flow capacity* λ_j is the difference between the value of a minimum cut where node j is included in the sink partition (informally, a minimum cut that passes through arcs in S_j^+) and $d_{k\ell}$. The excess values λ_j can be calculated by

$$\lambda_j = \min_{(\mathtt{lb}),(\mathtt{rb}): k-1 \leq (\mathtt{lb}) < j < (\mathtt{rb}) \leq \ell+1} \{d_{k(\mathtt{lb})} + u_{(\mathtt{lb})} + c(\cup_{j=\mathtt{lb}+1}^{\mathtt{rb}-1} S_j^+) + d_{(\mathtt{rb})\ell}\} - d_{k\ell}. \quad (2.11)$$

Remark 2.1. If S^+ is a path cover, then $\lambda_j \geq 0$ for all $j \in N$.

Proposition 2.3. Let S^+ be a path cover. Then,

$$\rho_t(S^+ \setminus \{t\}, L^-) = (c_t - \lambda_j)^+, \quad t \in S^+,$$

$$\rho_t(S^+, L^- \setminus \{t\}) = -\min\{c_t, \lambda_j\}, \quad t \in L^-.$$

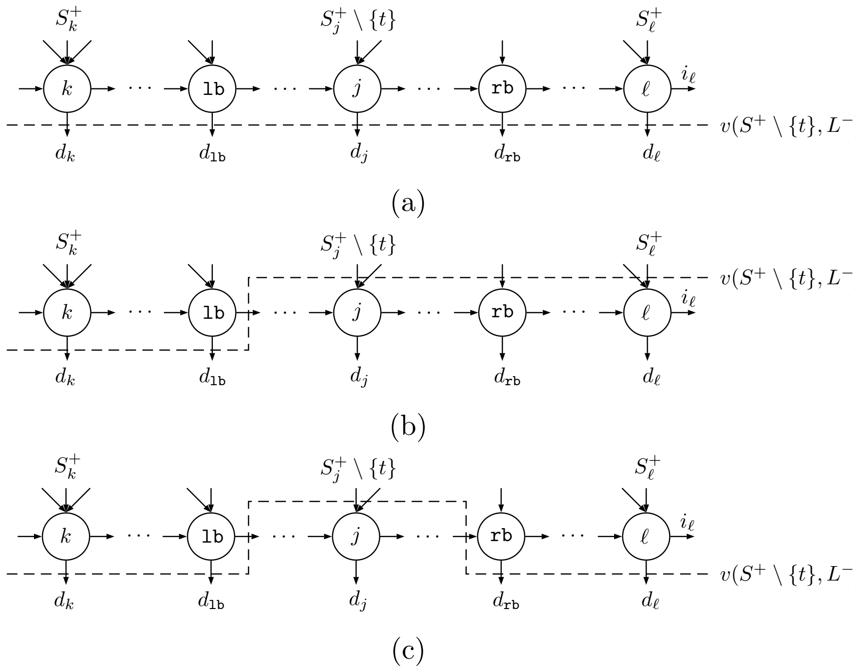


Figure 2.3: Representation of cases (i), (ii) and (iii) in Lemma 2.1.

Proof. First, notice that the value $v(S^+, L^-)$ will change if and only if at least one of the flow balance constraints (2.6) can no longer be satisfied at equality. Let $t \in S_j^+$ and suppose $c_t \leq \lambda_j$. Then, the excess flow that can be sent to node j will cover any flow that will be lost due to cancellation of arc t. Otherwise, all the excess capacity will be used up to satisfy demand d_j and the remaining unsatisfied demand will cause a decrease of $(c_t - \lambda_j)$ in the objective function (2.5). Similarly, let $t \in L_j^-$. Dropping t from L_j^- corresponds to adding an arc from node j to t_N. Now, we can potentially increase the objective function value by pushing more flow through S^+. The largest flow that can be pushed out of node j is λ_j, when merged with the capacity constraint of arc t, the increase in the objective function value becomes $\min\{\lambda_j, c_t\}$. □

Definition 2.2 and Proposition 2.3 lead to the following explicit description of the submodular inequalities for path substructures.

Corollary 2.4. Let S^+ be a path cover, then inequality (2.10) can be stated explicitly as

$$y(S^+) + \sum_{j \in N} \sum_{t \in S_j^+} (c_t - \lambda_j)^+ (1 - x_t) \leq d_{k\ell} + \sum_{j \in N} \sum_{t \in L_j^-} \min\{\lambda_j, c_t\} x_t + y(E^- \setminus L^-). \quad (2.12)$$

We refer to inequality (2.12) as the *submodular path inequality*. Note that if $c_t \leq \lambda_j$ for some $t \in L_j^-$, $j \in N$, then excluding arc t from the set L^- provides a stronger submodular path inequality since $y_t \leq c_t x_t$.

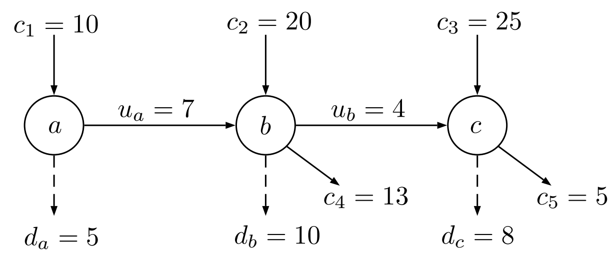


Figure 2.4: Path set example.

Remark 2.2. Observe that the flow cover inequalities of Padberg et al. (1985) are special cases of the submodular path inequalities (2.12). Suppose the path consists of a single node $N = \{j\}$ with demand $d := d_j > 0$. Then the $S^+ \subseteq E_j^+$ is a path cover if $\lambda := c(S^+) - d > 0$ and the resulting submodular path inequality

$$y(S^+) + \sum_{t \in S^+} (c_t - \lambda)^+ (1 - x_t) \le d + \lambda x(L^-) + y(E^- \setminus L^-)$$

is the flow cover inequality.

Example 1. Consider the path consisting of the three nodes in Figure 2.4. Suppose we select path $N = \{a, b, c\}$ and $S^+ = \{1, 2, 3\}$, $L^- = \{4, 5\}$. Then, $\lambda_a = 5$, $\lambda_b = 12$ and $\lambda_c = 21$ and the resulting submodular path inequality is

$$y_1 + y_2 + y_3 + 5(1 - x_1) + 8(1 - x_2) + 4(1 - x_3) \le 23 + 12x_4 + 5x_5. \tag{2.13}$$

For $N = \{b\}$, $S^+ = \{2\}$ and $L^- = \{4\}$, we have $\lambda = 10$ and the corresponding submodular path inequality is

$$y_2 + 10(1 - x_2) \le 10 + 10x_2 + i_2, \tag{2.14}$$

which is the flow cover inequality for node b with the same set selection (S^+, L^-). $\square$

Notice that calculating λ_j for some $j \in N$ using Equation (2.11) has a complexity of $O(|S^+| + |N|)$. However, we can compute *all* λ_j for $j \in N$ in the same complexity as well by using the minimum cut descriptions in Lemma 2.1. In particular, we compute all values λ_j, $j \in N$ simultaneously by using both a forward and a backward recursive formula. In a forward pass, we calculate, for all $j \in [k, \ell]$, the minimum cut value α_j for nodes $[k, j]$ that goes through S_j^+. In a backward pass, we calculate, for all $j \in [k, \ell]$, the minimum cut value β_j that goes through S_j^+ for nodes $[j, \ell]$. Then,

$$\lambda_j = \alpha_j + \beta_j - c(S_j^+) - d_{k\ell},$$

where α_j for $j \in [k, \ell]$ is

$$\alpha_j = \min\{\alpha_{j-1}, d_{k(j-1)} + u_{j-1}\} + c(S_j^+),$$

with $\alpha_{k-1} = 0$ and β_j for $j \in [k, \ell]$ is

$$\beta_j = \min\{\beta_{j+1}, d_{(j+1)\ell}\} + c(S_j^+),$$

with $\beta_{\ell+1} = 0$.

Proposition 2.5. All values λ_j, for $j \in N$, can be computed in $O(|S^+| + |V|)$ time.

Next, we provide necessary and sufficient conditions under which the submodular path inequalities are facet defining for $\text{conv}(\mathcal{P}_G)$.

Theorem 2.6. For $S^+ \subseteq E^+$ and $L^- \subseteq E^-$, let S^+ be a path cover for V. Then, the following conditions are necessary for inequality (2.12) to be facet-defining for $\text{conv}(\mathcal{P}_G)$.

1. $\rho_t(S^+ \setminus \{t\}, L^-) < c_t$ for all $t \in S^+$,
2. $\rho_t(S^+, L^- \setminus \{t\}) > -c_t$ for all $t \in L^-$,
3. if $L^- = \emptyset$, then $\max_{t \in S^+} \rho_t(S^+ \setminus \{t\}, L^-) > 0$,
4. for $p := \max\{j \in [k, \ell] \ : \ S_j^+ \cup L_j^- \neq \emptyset\}$, at least one of the following holds:
   (i) $c(S_p^+) < d_{p\ell}$ or
   (ii) $\max_{t \in S_p^+} \rho_t(S^+ \setminus \{t\}, L^-) > 0$ (i.e., $\max_{t \in S_p^+} c_t > \lambda_\ell$) or
   (iii) $L_p^- \neq \emptyset$,
5. if $\alpha_j = d_{k,j-1} + u_{j-1} + c(S_j^+)$ for some $j \in [k+1, \ell]$, then $\cup_{i=j}^{\ell} S_i^+$ is not a path cover for $\bar{V} = \{j, \ldots, \ell\}$.

Proof. See Appendix B.1. □

Theorem 2.7. For $S^+ \subseteq E^+$, $L^- \subseteq E^-$, let S^+ be a path cover for V. If $d_j \geq 0$, for all $j \in N$, then the necessary conditions in Theorem 2.6 along with the condition $\max_{t \in S_j^+} c_t > \lambda_j$ for $j \in N$ are sufficient for inequality (2.12) to be facet-defining for $\text{conv}(\mathcal{P}_G)$.

Proof. See Appendix B.2. □

2.3 Lifting submodular path inequalities

In this section, we first generalize the submodular path inequalities and then give a sequence independent lifting procedure a la Gu et al. (1999). Let $S^+ \subseteq E^+$ and $L^- \subseteq E^-$ be defined as in Section 2.2. We select a new subset of outgoing arcs $S^- \subseteq E^- \setminus L^-$. In this section, we let $K^+ = S^+$ and $K^- := L^- \cup S^-$. In other words, we select the objective function coefficients

$$a_t = \begin{cases} 1, & t \in S^+, \\ -1, & t \in E^- \setminus (S^- \cup L^-), \\ 0, & \text{otherwise.} \end{cases}$$

For convenience, let $L^{--} := E^- \setminus (S^- \cup L^-)$.

Definition The pair (S^+, S^-) is called a *generalized path cover* if $v(S^+, L^-) = d_{k\ell} + c(S^-)$.

For a generalized path cover (S^+, S^-) the optimization problem (2.5)-(2.9) and inequality (2.10) lead to the *generalized submodular path inequality*

$$y(S^+) + \sum_{j\in N} \sum_{t\in S_j^+} (c_t - \lambda_j)^+(1 - x_t) \leq d_{k\ell} + c(S^-) + \sum_{j\in N} \lambda_j x(L_j^-) + y(L^{--}), \quad (2.15)$$

where

$$\lambda_j = \alpha_j + \beta_j - c(S_j^+) - d_{k\ell} - c(S_j^-)$$

and α_j for $j \in [k, \ell]$ is

$$\alpha_j = \min\{\alpha_{j-1}, d_{k(j-1)} + c(\cup_{i=k}^{j-1} S_i^-) + u_{j-1}\} + c(S_j^+),$$

with $\alpha_{k-1} = 0$ and β_j for $j \in [k, \ell]$ is

$$\beta_j = \min\{\beta_{j+1}, d_{(j+1)\ell} + c(\cup_{i=j+1}^{\ell} S_i^-)\} + c(S_j^+),$$

with $\beta_{\ell+1} = 0$.

Example 2. Consider the path $N = \{a, b, c\}$ in Figure 2.5. For $S^+ = \{1, 2, 3, 4\}$, $L^- = \{8\}$ and $S^- = \{5, 6, 7\}$, we have $\lambda_a = 3, \lambda_b = 6$ and $\lambda_c = 9$. Then, the corresponding generalized submodular path inequality is

$$y_1 + y_2 + y_3 + y_4 + 7(1 - x_1) + (1 - x_2) + (1 - x_4) \leq 21 + 9x_8. \quad (2.16)$$

□

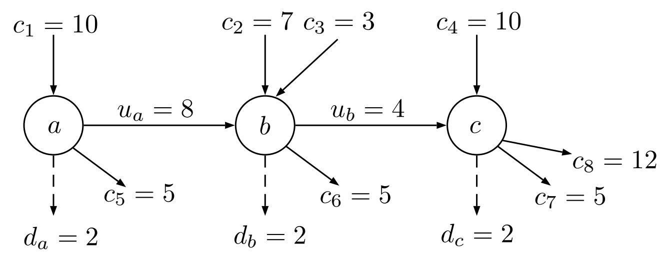


Figure 2.5: Path set example.

In order to strengthen inequality (2.15), we lift it by introducing variables x_t, $t \in S^-$ simultaneously into the submodular path inequality:

$$y(S^+) + \sum_{j\in N}\sum_{t\in S_j^+}(c_t - \lambda_j)^+(1-x_t) \le d_{k\ell} + \sum_{t\in S^-}\big(c_t + \theta_t(1-x_t)\big) + \sum_{j\in N}\lambda_j x(L_j^-) + y(L^{--}). \quad (2.17)$$

Let K be the set of variables used for simultaneous lifting. We select at most one arc from S_j^- for each $j \in N$ to include in K. Denote the arcs in K using the subscript $t_j := S_j^- \cap K$ and let $K = \{t_k, , ..., t_\ell\}$. For convenience, we refer to the capacity, flow and fixed-charge variables of arc t_j as c_j, y_j and x_j respectively.

Proposition 2.8 (Gu et al. (1999)). Let the function $f(\mathbf{z})$ be defined by (2.18)-(2.22), $h(\mathbf{z})$ be defined by (2.23)-(2.26), and set Z be defined by (2.27). If the lifting coefficients $\boldsymbol{\theta}$ are selected such that $h(\mathbf{z}) \le f(\mathbf{z})$ for any $\mathbf{z} \in Z$, and $f(\mathbf{z})$ is superadditive, then inequality (2.17) is valid for $\mathcal{P}_G$.

The function $f(\mathbf{z})$ is called the lifting function for inequality (2.17) and is defined as

$$
\begin{aligned}
f(\mathbf{z}) = \min \quad & G(\mathbf{x},\mathbf{y}) && (2.18)\\
\text{s.t.} \quad & i_{j-1} + y(S_j^+) - y(E_j^- \setminus S_j^-) - i_j \le d_j + c(S_j^-) - z_j, \quad \forall j \in N, && (2.19)\\
& 0 \le i_j \le u_j, \quad \forall j \in N, && (2.20)\\
& 0 \le y_t \le c_t x_t, \quad \forall t \in E, && (2.21)\\
& x_t \in \{0,1\}, \quad \forall t \in E, && (2.22)
\end{aligned}
$$

where

$$G(\mathbf{x},\mathbf{y}) := d_{k\ell} + c(S^-) + \sum_{j\in N}\lambda_j x(L_j^-) + y(L^{--}) - y(S^+) - \sum_{j\in N}\sum_{t\in S_j^+}(c_t-\lambda_j)^+(1-x_t)$$

is the difference between the right and the left hand sides of inequality (2.15). We call function $h(\mathbf{z})$ the *dual lifting function* and it is defined as

$$
\begin{aligned}
h(\mathbf{z}) := \max \quad & \sum_{t_j \in K} \theta_j (x_j - 1) && && (2.23)\\
\text{s.t.} \quad & y_j - c_j = -z_j, && t_j \in K, && (2.24)\\
& 0 \le y_j \le c_j x_j, && t_j \in K, && (2.25)\\
& x_j \in \{0, 1\}, && t_j \in K. && (2.26)
\end{aligned}
$$

Finally, the feasible region of the vector $\mathbf{z}$ is:

$$
Z = \big\{ z \in \mathbb{R}^v : \ \exists (x, y, i) : \ y_j - c_j = -z_j, \ t_j \in K, \\
y_t = c_t, \ t \in S^- \setminus K, \ (2.19) - (2.22) \big\}. \quad (2.27)
$$

Definition (Gu et al. (1999)) A function $\Phi(\mathbf{z})$ is a *superadditive valid lifting function* of $f(\mathbf{z})$ if $\Phi(\mathbf{z}) \le f(\mathbf{z})$ for all $\mathbf{z} \in Z$ and is superadditive.

Proposition 2.9 (Gu et al. (1999))**.** Let $\Phi(\mathbf{z})$ be a superadditive valid lifting function. If the lifting coefficients $\boldsymbol{\theta}$ satisfy $h(\mathbf{z}) \le \Phi(\mathbf{z})$ for any $\mathbf{z} \in Z$, then inequality (2.17) is valid for $\mathcal{P}_G$.

2.3.1 Superadditive valid lifting function

In this section, we provide a superadditive valid lifting function for $f(\mathbf{z})$. We tackle the multidimensionality of $f(\mathbf{z})$ by decomposing it with respect to $j \in N$. Notice that the path arc flows i_j, $j = k, \ldots, \ell - 1$ are the only terms binding constraints (2.19) together. Therefore, we first duplicate the path arc flow variables, i_j, $j = k, \ldots, \ell - 1$ into i_j^1 and i_j^2. Then, we add a constraint that enforce them to be equal. This procedure leads to the following formulation

$$
\begin{aligned}
f(\mathbf{z}) = \min \quad & G(\mathbf{x}, \mathbf{y}) \\
\text{s.t.} \quad & i_{j-1}^1 + y(S_j^+) - y(E_j^- \setminus S_j^-) - i_j^2 \le d_j + c(S_j^-) - z_j, \qquad j \in N, && (2.28)\\
& 0 \le i_j^s \le u_j, \qquad j \in N, \quad s = 1, 2, && (2.29)\\
& i_j^2 = i_j^1, \qquad j \in N \setminus \{\ell\}, && (2.30)\\
& (2.21) - (2.22).
\end{aligned}
$$

A Lagrangian relaxation

Now, constraints (2.30) are the only constraints that bind the subproblems for each node $j \in N$ together. We employ a Lagrangian relaxation w.r.t. (2.30) to decompose the problem:

$$
\begin{aligned}
f^L(\mathbf{z},\boldsymbol{\mu}) = \quad & G(\mathbf{x},\mathbf{y}) + \sum_{j=k}^{\ell-1} \mu_j (i_j^1 - i_j^2) \\
\text{s.t.} \quad & (2.21) - (2.22), (2.28) - (2.29).
\end{aligned}
$$

Notice that the relaxed formulation above is now decomposable

$$
f^L(\mathbf{z},\boldsymbol{\mu}) = \sum_{j=k}^{\ell} f_j^L(z_j, \mu_{j-1}, \mu_j),
$$

where

$$
\begin{aligned}
f_j^L(z_j,\mu_{j-1},\mu_j) = \min \quad & H_j(\mathbf{x},\mathbf{y}) & (2.31)\\
\text{s.t.} \quad & i_{j-1}^1 + y(S_j^+) - y(E_j^- \setminus S_j^-) - i_j^2 \le d_j + c(S_j^-) - z_j, & (2.32)\\
& 0 \le y_t \le c_t x_t, \qquad t \in E_j^+ \cup E_j^-, & (2.33)\\
& x_t \in \{0,1\}, \qquad t \in E_j^+ \cup E_j^-, & (2.34)\\
& i_{j-1}^1 \le u_{j-1}, & (2.35)\\
& i_j^2 \le u_j, & (2.36)
\end{aligned}
$$

and

$$
H_j(\mathbf{x},\mathbf{y}) := d_j + c(S_j^-) + \lambda_j x(L_j^-) + y(L_j^{--}) - y(S_j^+) - \sum_{t \in S_j^+} (c_t - \lambda_j)^+ (1 - x_t) + \mu_j i_j^2 - \mu_{j-1} i_{j-1}^1.
$$

From Lagrangian duality, $f^L(\mathbf{z},\boldsymbol{\mu}) \le f(\mathbf{z})$ for any $\boldsymbol{\mu}$. We pick μ_i to be 1 for all $i \in [k,\ell]$ and $\mu_{k-1} = 0$ for simplicity. Therefore, in the remainder of this chapter, we refer to $f_j^L(z_j,\mu_{j-1},\mu_j)$ as $f_j^L(z_j)$.

Let $S_j^{++} = \{t \in S_j^+ : \ c_t > \lambda_j\}$, let $S_j^{++} \cup L_j^- := \{v_1, v_2, \ldots, v_{r_j}\}$ where $c_{v_i} \ge c_{v_{i+1}}$ and $r_j := |S_j^{++} \cup L_j^-|$. In formulation (2.31)-(2.36), we observe three conditions that hold in at least one of its optimal solutions. First, y_t for $t \in E_j^+ \setminus S_j^+$ do not appear either in $H_j(\mathbf{x},\mathbf{y})$ or in constraint (2.32) and therefore, can be assumed to be zero. Second, if $t \in S_j^{++}$, then either $y_t > c_t - \lambda_j$ or $y_t = 0$. Third, if $t \in L_j^-$, then either $y_t = 0$ or $y_t = c_t$.

While optimizing $f_j^L(z_j)$, we first set $y_t = c_t$ for all $t \in S_j^{++}$ and $y_t = 0$ for all $t \in L_j^-$. If the flow balance constraint (2.32) is violated, then we decrease the flow of arcs in S_j^{++} and increase the flow of variables in L_j^- in the order of $v_1, v_2, \ldots, v_{r_j}$.

Let $mp_j = \min_{t \in S_j^{++}}\{c_t\}$ and p_j be the largest index such that $c_{v_{p_j}} = mp_j$, $M_{j,i} = \sum_{k=1}^{i} c_{v_k}$ and $M_{j,0} = 0$. Let $m_j = u_{j-1} + u_j + c(S_j^+ \setminus S_j^{++}) + c(L_j^{--})$, $ml_j = \min\{\lambda_j, m_j\}$ and $\varphi_{j,i} = \max\{0, c_{v_{i+1}} - (mp_j - \lambda_j) - ml_j\}$ for $i = p_j, \ldots, r_j - 1$.

We may assume that the path arc $(j-1, j)$ is in $S_j^+ \setminus S_j^{++}$ and path arc $(j, j+1)$ is in L_j^{--} since they appear in formulation (2.31)-(2.36) with same coefficients. The variables $c_t - y_t$ for $t \in L_j^{--}$ (thus $u_j - i_j^2$ included) and y_t for $t \in S_j^+ \setminus S_j^{++}$ (thus i_{j-1}^1 included) appear together in formulation (2.31)-(2.36). Therefore, without loss of generality, we can merge these variables into a new variable w and assume that the corresponding arcs are always open:

$$w = i_{j-1}^1 + (u_j - i_j^2) + y(S_j^+ \setminus S_j^{++}) + c(L_j^{--}) - y(L_j^{--}).$$

Then, the formulation (2.31)-(2.36) can be simplified as:

$$f_j^L(z_j) = \min \quad \hat{d}_j - w - c(L_j^-) + \lambda_j x(L_j^-) - y(S_j^{++}) - \sum_{t \in S_j^{++}} (c_t - \lambda_j)(1 - x_t) \tag{2.37}$$

$$\text{s.t.} \quad w + y(S_j^{++}) + c(L_j^-) - y(L_j^-) \leq \hat{d}_j - z_j, \tag{2.38}$$

$$0 \leq y_t \leq c_t x_t, \qquad t \in S_j^{++} \cup L_j^-, \tag{2.39}$$

$$x_t \in \{0, 1\}, \qquad t \in S_j^{++} \cup L_j^-, \tag{2.40}$$

$$0 \leq w \leq m_j, \tag{2.41}$$

where $\hat{d}_j = d_j + u_j + c(E_j^-)$. Moreover, letting

$$\bar{\lambda}_j := u_{j-1} + c(S^+) - \big(d_j + c(S_j^-)\big), \qquad j \in N \tag{2.42}$$

we observe that

$$\hat{d}_j = m_j + M_{j,r_j} - \bar{\lambda}_j.$$

Theorem 2.10. The function $f_j^L(z)$ can be expressed as:

$$f_j^L(z) = \begin{cases} z + i\lambda_j - M_{j,i}, & \text{if} \quad M_{j,i} - \bar{\lambda}_j \leq z \leq M_{j,i} - \bar{\lambda}_j + \lambda_j, \quad i \in [0, p_j - 1] \\ (i+1)\lambda_j - \bar{\lambda}_j, & \text{if} \quad M_{j,i} + \lambda_j - \bar{\lambda}_j \leq z \leq M_{j,i+1} - \bar{\lambda}_j, \quad i \in [0, p_j - 1] \\ z + i\lambda_j - M_{j,i}, & \text{if} \quad M_{j,i} - \bar{\lambda}_j \leq z \leq M_{j,i} - \bar{\lambda}_j + ml_j, \quad i \in [p_j, r_j - 1] \\ z + (i+1)\lambda_j - M_{j,i} - \varphi_{j,i} - ml_j, & \text{if} \\ \qquad M_{j,i} - \bar{\lambda}_j + ml_j < z \leq M_{j,i} - \bar{\lambda}_j + ml_j + \varphi_{j,i}, & i \in [p_j, r_j - 1] \\ (i+1)\lambda_j - \bar{\lambda}_j, & \text{if} \\ \qquad M_{j,i} - \bar{\lambda}_j + ml_j + \varphi_{j,i} \leq z \leq M_{i+1} - \bar{\lambda}_j, & i \in [p_j, r_j - 1] \\ z - M_{j,r_j} + r_j\lambda_j, & \text{if} \quad M_{j,r_j} - \bar{\lambda}_j \leq z \leq \hat{d}_j. \end{cases} \tag{2.43}$$

Proof. See Appendix B.3. □

Remark 2.3. If $\lambda_j = \bar{\lambda}_j$, then $f_j^L(z)$ is the same as the lifting function of the flow cover inequality provided in Gu et al. (1999).

Remark 2.4. In equation (2.42), the incoming flow from up-stream nodes $\{k,\dots,j-1\}$ is $i_{j-1}=u_{j-1}$ and there is no flow being pulled from down-stream nodes $\{j+1,\dots,\ell\}$. Therefore, $\bar{\lambda}_j \geq \lambda_j$ and the domain of $f_j^L(z)$ includes $[0,\infty)$.

The function $f_j^L(z_j)$ is not necessarily superadditive and does not satisfy the conditions in Proposition 2.9. Therefore, we construct a lower-bound on f_j^L that is superadditive on $\mathbb{R}_+ \cup \{0\}$. Gu et al. (1999) propose a slightly different version of the following superadditive valid lifting function for flow cover inequalities:

$$\psi_j(z) = \begin{cases} z + i\lambda_j - M_{j,i} & M_{j,i} - \bar{\lambda}_j \leq z \leq M_{j,i} - \bar{\lambda}_j + \lambda_j, \quad i \in [0, p_j - 1] \\ (i+1)\lambda_j - \bar{\lambda}_j & M_{j,i} + \lambda_j - \bar{\lambda}_j \leq z \leq M_{j,i+1} - \bar{\lambda}_j, \quad i \in [0, p_j - 1] \\ z + i\lambda_j - M_{j,i} & M_{j,i} - \bar{\lambda}_j \leq z \leq M_{j,i} - \bar{\lambda}_j + ml_j + \varphi_{j,i}, \quad i \in [p_j, r_j - 1] \\ (i+1)\lambda_j - \bar{\lambda}_j & M_{j,i} - \bar{\lambda}_j + ml_j + \varphi_{j,i} < z \leq M_{j,i+1} - \bar{\lambda}_j, \quad i \in [p_j, r_j - 1] \\ z - M_{j,r_j} + r_j\lambda_j & M_{j,r_j} - \bar{\lambda}_j \leq z \leq \hat{d}_j. \end{cases} \tag{2.44}$$

If the path consists of a single node as in flow cover inequalities, then $\psi_j(z)$ is superadditive. However, when there are multiple nodes in V, the superadditivity of both f_j^L and ψ_j depends on where the $z=0$ case lies. Therefore, we examine the function f_j^L case by case. Let the index T_j be $T_j = \min\{1 \leq i \leq r_j \;:\; M_{j,i} - \bar{\lambda}_j \geq 0\}$ if $M_{j,r_j} - \bar{\lambda}_j \geq 0$, and $T_j = r_j$ otherwise.

Theorem 2.11. The function $\psi_j(z) \leq f_j^L(z)$ is superadditive if $T_j\lambda_j - \bar{\lambda}_j \leq 0$.

Proof. See Appendix B.4. □

Note that if $T_j\lambda_j - \bar{\lambda}_j > 0$, then ψ_j is not superadditive since $\psi_j(0) > 0$. Under this case, we obtain a different superadditive lower bound. Using Lemma 2.2, we find the largest convex lower-bound of $f_j^L(z)$, $\phi_j(z)$, such that $\phi_j(0) \leq 0$ is the superadditive lower estimation.

Lemma 2.2 (Hille and Phillips (1957) p. 237)**.** A convex function $\phi : \mathbb{R}_+ \cup \{0\} \to \mathbb{R}$ with $\phi(0) \leq 0$ is superadditive.

The generic form of the largest convex lower-bound of $f_j^L(z)$ that has a non-positive value at zero is

$$\phi_j(z) = \begin{cases} \frac{\tau\lambda_j - \bar{\lambda}_j - \Gamma}{M_{j,\tau} - \bar{\lambda}_j} z + \Gamma, & 0 \leq z \leq M_{j,\tau} - \bar{\lambda}_j, \\ \frac{\lambda_j}{c_{j_{i+1}}}(z - M_{j,i} + \bar{\lambda}_j) + i\lambda_j - \bar{\lambda}_j, & M_{j,i} - \bar{\lambda}_j \leq z \leq M_{j,i+1} - \bar{\lambda}_j, \quad i \in [\tau, r_j - 1] \\ z - M_{j,r_j} + r_j\lambda_j, & M_{j,r_j} - \bar{\lambda}_j \leq z \leq \hat{d}_j, \end{cases} \tag{2.45}$$

where Γ is the value of $\phi_j(0) = \min\{0, f_j^L(0)\}$ and τ is the smallest index that ensures the slope of linear function connecting points $(0, \Gamma)$ and $(M_{j,\tau} - \bar{\lambda}_j, \phi_j(M_{j,\tau} - \bar{\lambda}_j))$ is the smallest compared to the slope of next linear pieces of ϕ_j. The values of Γ and τ depend on the parameters of the problem (see Appendix B.6 for explicit descriptions).

Let $\Phi_j(z)$ be the superadditive lower bound of f_j^L. In summary, $\Phi_j(z)$ takes two different forms and it has the following description:

$$\Phi_j(z) = \begin{cases} \psi_j(z) & \text{if} \quad T_j\lambda_j - \bar{\lambda}_j \leq 0, \\ \phi_j(z) & \text{otherwise.} \end{cases}$$

For each node $j \in N$, the lower bounding function $\Phi_j(z_j)$ is superadditive. Therefore, $\sum_{j=k}^{\ell} \Phi_j(z_j)$ is superadditive as well. Furhermore, since inequality (2.15) is valid for $\mathcal{P}_G$, $f(\mathbf{0}) \geq 0$. On the contrary, the approximation $\sum_{j=k}^{\ell} \Phi_j(0)$ can be less than zero. Then, we can construct a tighter superadditive valid lifting function $\Phi(\mathbf{z})$ by:

$$\Phi(\mathbf{z}) = \max\left\{ \sum_{j=k}^{\ell} \Phi_j(z_j), 0 \right\} \leq f(\mathbf{z}).$$

Note that since $\sum_{j=k}^{\ell} \Phi_j(z_j)$ is superadditive, $\max\{\sum_{j=k}^{\ell} \Phi_j(z_j), 0\}$ is superadditive as well, due to Observation 5 in Appendix B.5. **Example 2 (cont).** Recall that path $N = \{a, b, c\}$ for the set selection $S^+ = \{1, 2, 3, 4\}$, $L^- = \emptyset$ and $S^- = \{5, 6, 7\}$ have $\lambda_a = 3$, $\lambda_b = 6$ and $\lambda_c = 9$. However note that the upper bounds $\bar{\lambda}_a = 3$, $\bar{\lambda}_b = 11$ and $\bar{\lambda}_c = 11$. The sets $S_j^{++} \cup L_j^-$ are $S_a^{++} \cup L_a^- = \{1\}$, $S_b^{++} \cup L_b^- = \{2\}$ and $S_c^{++} \cup L_c^- = \{4, 8\}$ with $v_{a,1} = 1, v_{b,1} = 2$ and $v_{c,1} = 8, v_{c,2} = 4$ since $c_8 > c_4$. The cardinalities of sets $S_j^{++} \cup L_j^-$ are $r_a = r_b = 1$ and $r_c = 2$ and the cumulative capacities are $M_{a,v_{a,1}} = 10, M_{b,v_{b,1}} = 7$ and $M_{c,v_{c,1}} = 12, M_{c,v_{c,2}} = 12 + 10 = 22$. We also have $p_a = 1, p_b = 1$ and $p_c = 2$. The values m_j for $j = a, b, c$ are $m_a = 8, m_b = 8 + 8 + 3 = 19$ and $m_c = 8$. Finally, the modified demand values are $\hat{d}_a = \hat{d}_b = 2 + 8 + 5 = 15$ and $\hat{d}_c = 2 + 5 + 12 = 19$.

Let us now examine the decomposed functions $f_j^L(z)$ for Example 2. Using the values calculated in Section 2.3.1, we observe that

$$f_a^L(z) = \begin{cases} 0 & 0 \leq z \leq 7 \\ z - 7 & 7 \leq z \leq 15 \end{cases}, \quad f_b^L(z) = \begin{cases} z - 1 & 0 \leq z \leq 15, \end{cases} \quad \text{and}$$

$$f_c^L(z) = \begin{cases} -2 & 0 \leq z \leq 1 \\ z - 3 & 1 \leq z \leq 10 \\ 7 & 10 \leq z \leq 11 \\ z - 4 & 11 \leq z \leq 19. \end{cases}$$

Note that $T_a = T_b = T_c = 1$ and $T_a\lambda_a - \bar{\lambda}_a = 0 \leq 0$, $T_b\lambda_b - \bar{\lambda}_b = -5 \leq 0$ and $T_c\lambda_c - \bar{\lambda}_c = -2 \leq 0$. Therefore, $\psi_a(z), \psi_b(z)$ and $\psi_c(z)$ are all superadditive. We also observe that under this parameter set, $f_j^L(z) = \psi_j(z)$ for $j = a, b, c$. We then have $\Phi_j(z) = \psi_j(z)$ for $j = a, b, c$. □

2.3.2 Dual lifting function

In this section, we investigate the explicit form of the dual lifting function $h(\mathbf{z})$. First, notice that the optimization problem defined by (2.24)-(2.26) is decomposable with respect to each arc $t_j \in K$. Therefore, we write it as a sum of single dimensional functions $h_j(z_j)$:

$$h(\mathbf{z}) = \sum_{t_j \in K} h_j(z_j),$$

where

$$\begin{aligned} h_j(z_j) = \max \quad & \theta_j(x_j - 1) \\ \text{s.t.} \quad & y_j - c_j = -z_j, \\ & 0 \leq y_j \leq c_j x_j, \\ & x_j \in \{0, 1\}. \end{aligned}$$

Remark 2.5. The function $h(\mathbf{z})$ is decomposable with respect to $j \in N$ as well, by the construction of the lifting set K.

Functions $h_j(z_j)$ are the same as the dual lifting function of single node flow cover inequalities. Therefore, we directly use the result of Gu et al. (1999) to analytically evaluate it.

Theorem 2.12 (Gu et al. (1999) Theorem 4)**.**

$$h_j(z_j) = \begin{cases} 0 & 0 \leq z_j < c_j, \\ -\theta_j & z_j = c_j, \end{cases}$$

with $\theta_j \leq 0$.

2.3.3 Lifting coefficients

Using Proposition 2.9, we construct the lifting coefficients $\boldsymbol{\theta}$ such that

$$\sum_{j=k}^{\ell} h_j(z_j) \leq \max\left\{ \sum_{j=k}^{\ell} \Phi_j(z_j), 0 \right\}. \tag{2.46}$$

Theorem 2.12 shows that the function takes a nonzero value only if $z_j = c_j$. Observe that if $z_j \neq c_j$ for all $j \in N$, then inequality (2.46) implies $0 \leq \max\big\{\sum_{j=k}^{\ell} \Phi_j(z_j), 0\big\}$ and does not provide any information on the lifting coefficients $\boldsymbol{\theta}$. Let us now examine the case where $z_i \neq c_i$ for $i \neq j$ and $z_j = c_j$ for some $j \in N$. Using Theorem 2.12, inequality (2.46) can be written as

$$\sum_{j=k}^{\ell} h_j(z_j) = -\theta_j \leq \max\big\{0, \Phi_j(c_j) + \sum_{m \neq j} \Phi_k(z_m)\big\}, \qquad \mathbf{z} : z_j = c_j, z_m \neq c_m, m \neq j, \tag{2.47}$$

for each $j \in N$. Inequality (2.47) implies that $-\theta_j$ should be less than or equal to all possible values of the right hand side, equivalently:

$$-\theta_j \leq \min_{z_j = c_j, z_m \neq c_m,\ m \neq j} \left\{ \max\big\{0, \Phi_j(c_j) + \sum_{m \neq j} \Phi_m(z_m)\big\} \right\}. \tag{2.48}$$

Since $\Phi_i(z_i)$ is a non-decreasing function, inequality (2.48) can be simplified as

$$-\theta_j \leq \max\big\{0, \Phi_j(c_j) + \sum_{m \neq j} \Phi_m(0)\big\}. \tag{2.49}$$

Let us now examine the vector $\mathbf{z}$ where $z_i = c_i$ and $z_j = c_j$ and $z_m \neq c_m$ for any $m \neq i$ and $m \neq j$. Then, using Theorem 2.12, inequality (2.46) becomes:

$$\begin{aligned} -\theta_j - \theta_i \leq \max\big\{0, \Phi_i(c_i) + \Phi_j(c_j) + \sum_{m \neq i,j} \Phi_m(z_m)\big\}, \\ \forall \mathbf{z} : z_i = c_i, z_j = c_j, z_m \neq c_m, m \neq i, j. \end{aligned} \tag{2.50}$$

Since $\Phi_m(z_m)$ is a non-decreasing function, we can replace the right hand side of inequality (2.50) with its smallest value:

$$-\theta_j - \theta_i \leq \max\big\{0, \Phi_i(c_i) + \Phi_j(c_j) + \sum_{m \neq i,j} \Phi_m(0)\big\}. \tag{2.51}$$

Proposition 2.13. Inequality (2.51) provides a smaller lower bound on $\theta_i + \theta_j$ than bounds of θ_i and θ_j in inequality (2.49) summed. Equivalently,

$$\begin{aligned} \max\big\{0, \Phi_i(c_i) + \Phi_j(c_j) + \sum_{m \neq i,j} \Phi_m(0)\big\} \geq \max\big\{0, \Phi_j(c_j) + \sum_{m \neq j} \Phi_m(0)\big\} \\ + \max\big\{0, \Phi_i(c_i) + \sum_{m \neq i} \Phi_m(0)\big\}. \end{aligned} \tag{2.52}$$

Proof. Recall that $\Phi_j(0) \leq 0$ and $\Phi_j(z_j)$ is a non-decreasing function. Hence,

$$\Phi_i(c_i) + \Phi_j(c_j) + \sum_{m \neq i,j} \Phi_m(0) \geq \Phi_j(c_j) + \sum_{m \neq j} \Phi_m(0),$$
$$\Phi_i(c_i) + \Phi_j(c_j) + \sum_{m \neq i,j} \Phi_m(0) \geq \Phi_i(c_i) + \sum_{m \neq i} \Phi_m(0),$$
$$\Phi_i(c_i) + \Phi_j(c_j) + \sum_{m \neq i,j} \Phi_m(0) \geq \Phi_j(c_j) + \sum_{m \neq j} \Phi_m(0) + \Phi_i(c_i) + \sum_{m \neq i} \Phi_m(0).$$

□

Proposition 2.13 implies that the constraints provided in (2.49) are tighter than the constraints provided by (2.51). Using the same argument, one can easily show that as the number of components of vector $\mathbf{z}$ such that $z_j = c_j$ is increased, the constraints enforced on the coefficients $\boldsymbol{\theta}$ by inequality (2.46) get weaker.

Theorem 2.14. Lifted generalized submodular path inequality (2.17) where $\theta_j = -\max\left\{0, \Phi_j(c_j) + \sum_{k \neq j} \Phi_k(0)\right\}$ is valid for $\mathcal{P}_G$.

Proof. Selecting θ_j as the lower bound given in (2.49) guarantees that inequality (2.46) is satisfied, due to the tightness argument in Proposition 2.13. Then, inequality (2.17) is valid for the feasibility set $\mathcal{P}_G$ from Proposition 2.9. □

Example 2 continued: We now calculate the lifting coefficients for arcs in $S^- = \{5, 6, 7\}$. From Theorem 2.14, we have $\theta_5 = -\max\{0, \Phi_a(5) + \Phi_b(0) + \Phi_c(0)\} = -\max\{0, 0-1-2\} = 0$, $\theta_6 = -\max\{0, \Phi_a(0)+\Phi_b(5)+\Phi_c(0)\} = -\max\{0, 0+4-2\} = -2$ and $\theta_7 = -\max\{0, \Phi_a(0) + \Phi_b(0) + \Phi_c(5)\} = -\max\{0, 0-1+2\} = -1$. Then, we lift inequality (2.16) using Theorem 2.14 and obtain

$$y_1 + y_2 + y_3 + y_4 + 7(1 - x_1) + (1 - x_2) + (1 - x_4) \leq 21 - 2(1 - x_6) - (1 - x_7) + 9x_8.$$

□

2.4 Computational study

We test the effectiveness of the submodular path inequalities when used as cuts in a branch-and-cut framework. The computational experiments were carried out on a `Linux` workstation with 2.93 GHz Intel®Core$^{\text{TM}}$ i7 CPU and 8 GB of RAM. The branch-and-cut algorithm was implemented in C++ using Concert technology of CPLEX (version 12.5) with one hour time limit and 1 GB memory limit. We set the number of threads to one in the computations.

We test the submodular path inequalities on networks where the nodes form a simple path, and each node has a fixed number of incoming and outgoing arcs and nonnegative demands, which is typical in production planning problems. Given fixed cost f_t and variable cost p_t associated with an arc $t \in E$, and variable and fixed inventory holding costs h_j and s_j for a path arc j, we solve the following mixed integer optimization problem:

$$\begin{aligned}
\min \quad & \sum_{t \in E} \big(f_t x_t + p_t y_t\big) + \sum_{j \in N} \big(h_j i_j + s_j z_j\big) \\
\text{s.t.} \quad & i_{j-1} + y(E_j^+) - y(E_j^-) - i_j = d_j, \quad j \in N, \\
& 0 \le y_t \le c_t x_t, \quad t \in E, \\
& 0 \le i_j \le u_j z_j, \quad j \in N, \\
& x_t \in \{0,1\}, \quad t \in E, \\
& z_j \in \{0,1\}, \quad j \in N,
\end{aligned}$$

where y_t and i_j are the flow values on a non-path arc $t \in E$ and a path arc $j \in N$ respectively, and x_t and z_j are the binary variables representing whether these arcs are being used or not.

Finding violated submodular path inequalities

Given a path and a feasible solution $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{i}^*)$ to the LP relaxation with fractional $\mathbf{x}^*$, the separation problem aims to find sets S^+, L^- and S^- that maximize the violation

$$\sum_{t \in S^+} \left[y_t^* + (c_t - \lambda_j)^+(1 - x_t^*)\right] - d_{k\ell} - c(S^-) - \sum_{t \in L^-} \lambda_j x_t^* - \sum_{t \in L^{--}} y_t^*.$$

We use the knapsack relaxation based heuristic separation strategy described in (Wolsey and Nemhauser, 1999, pg. 500) for flow cover inequalities to choose sets S^+ and S^- with a knapsack capacity $d_{k\ell}$. Then, we add arc $t \in E^- \setminus S^-$ to L^- if $\lambda_j x_t^* < y_t^*$ and $\lambda_j < c_t$.

Instance Generation

The data we used for computational experiments has the following properties: There are v nodes on the network and each node has a single non-path incoming arc and a single non-path outgoing arc. Demand is drawn from integer uniform between 1 and 19. The parameter $c \in \{2, 3, 4, 5\}$ controls the tightness of arc capacities compared to the average demand of all nodes and $f \in \{100, 200, 500, 1000\}$ controls how large the fixed costs are compared to inventory holding costs. The capacities of non-path incoming, non-path outgoing and path arcs are drawn from integer uniform with bounds $[0.75 \times c \times \bar{d}, 1.25 \times c \times \bar{d}]$, $2 \times [0.75 \times c \times \bar{d}, 1.25 \times c \times \bar{d}]$ and $[0.75 \times c \times \bar{d}, c \times \bar{d}]$ respectively where $\bar{d}$ is the average demand. Similarly, fixed costs of non-path incoming and non-path outgoing and path arcs are generated from integer uniform with bounds

$[0.9 \times f \times \bar{h}, 1.1 \times f \times \bar{h}]$, $[0.2 \times f \times \bar{h}, 0.5 \times f \times \bar{h}]$ respectively where $\bar{h}$ is the average inventory holding cost. Path arcs have fixed and variable costs of f and 10, respectively. Variable ordering costs of non-path incoming and outgoing arcs are generated from integer uniform at the interval $[81, 119]$ and $-1 \times [160, 240]$. Five random instances are generated for each (v, c, f) combination.

Preprocessing

In order to tighten the capacities of an instance we apply the following preprocessing steps for all arcs that have positive variable costs.

1) $\bar{u}_j = \min\{u_j, (d_{j+1})^+ + u_{j+1} + c(E^-_{j+1})\}$, for $j \in N$

2) $\bar{c}_t = \min\{c_t, (d_j)^+ + \bar{u}_j + c(E^+_j)\}$ for all $t \in E^+_j$, $j \in N$.

We report a number of performance measures in the following tables. Let z_{INIT} be the optimal objective function value of the LP relaxation with no valid inequalities added, z_{ROOT} be the optimal objective function value of the LP relaxation after adding the violated cuts at the root node. Moreover, let z_{UB} be the objective function value of the best feasible solution found within the time/memory limit. We report the percentage gap improvement (`gap imp` $= 100 \times \frac{z_{\text{ROOT}} - z_{\text{INIT}}}{z_{\text{BEST}} - z_{\text{INIT}}}$), the initial gap (`init gap` $= 100 \times \frac{z_{\text{UB}} - z_{\text{INIT}}}{z_{\text{UB}}}$), and the root gap (`root gap` $= 100 \times \frac{z_{\text{UB}} - z_{\text{ROOT}}}{z_{\text{UB}}}$). Moreover, we report the number of violated cuts added (`cuts`), the number of branch and bound nodes explored (`nodes`) and the elapsed time for solving an instance (`time`) in seconds. If the instance is not solved within the time/memory limit, then we report in parentheses the average percentage gap between the best lower bound and the best integer solution `endgap` and the number of instances with a positive end gap `unslv`. We round all the values to the nearest integer except for percent gaps. Each row reports the average values for `init gap`, `gap imp`, `cuts`, `nodes`, `time` and `endgap` over five instances. The user cuts are added only at the root node of the branch and bound tree.

In Table 2.1, we compare submodular path inequalities (2.12) (columns `spi`) with lifted generalized submodular path inequalities (2.17) (columns `lspi`) for instances with 50 nodes. To see first the effect of the proposed inequalities alone, in this experiment, all built-in CPLEX cuts are disabled. We enumerated all paths in the graph and for each path we used the separation heuristic explained above to find violated submodular path inequalities. Observe that both `spi` and `lspi` perform better for smaller f values. These results in the table suggest that submodular path inequalities alone reduce the integrality gap on average by 61% and that the generalized submodular path inequalities improve the gap reduction by another 15% on average. Furthermore, we observe that the number of nodes, the average solution times and the number of unsolved instances decrease substantially. Table 2.1 shows the positive impact of submodular path inequalities and their generalizations very clearly. In the remainder of computational study, we use `lspi` for the experiments.

Table 2.1: Comparison of submodular path inequalities (`spi`) and lifted generalized submodular path inequalities (`lspi`), $n = 50$.

			`gap imp %`		`cuts`		`nodes`		`time (endgap:unslv)`	
c	f	`init gap`	`spi`	`lspi`	`spi`	`lspi`	`spi`	`lspi`	`spi`	`lspi`
2	100	11.3	85.7	98.6	163	267	703	10	1	1
	200	10.2	69.6	74.8	500	618	641429	170959	230	88
	500	13.2	62.4	66.9	436	527	43400	12615	25	21
	1000	15.8	60.7	66.6	373	507	57140	21795	23	21
3	100	84.7	71.9	98.7	95	173	3469	4	1	1
	200	17.2	73.8	91	206	346	379879	349	78	3
	500	18.4	51.8	58.4	398	558	7646575	1799858	2088 (0.2:2)	617
	1000	20.3	49.1	55.6	346	449	2701397	744416	669	244
4	100	14.9	69.7	99.1	73	131	366	3	0	0
	200	87.7	69.1	96.2	108	201	12000	19	2	1
	500	20.1	48.2	54.9	417	494	8907851	3775955	2134 (1.3:3)	1093
	1000	23.6	42.7	49.3	403	450	5298416	2077797	1304 (0.1:1)	628
5	100	7.7	61.1	99.3	68	113	746	3	0	0
	200	47.9	67.1	97.9	88	148	19829	6	3	0
	500	25.6	51.7	62.8	312	383	10264838	2583423	2064 (1.5:3)	688
	1000	27	41.8	48.5	435	490	6378901	4765505	1495 (0.6:1)	1272 (0.3:1)
Average		**27.8**	**61.0**	**76.2**	**276**	**366**	**2647309**	**997045**	**632 (0.2:1)**	**292 (0.0:0)**

Although enumerating all the paths for a given network shows the full potential of the inequalities, it is too time consuming except for small cases. Therefore, in Table 2.2, we examine the effect of path size on the computational effectiveness of `lspi`. We use the same instances ($n = 50$) and separation heuristic as in Table 2.1. In columns $= 1$, we select all paths of size 1 whereas in columns $\leq k$, we enumerate all paths of sizes $1, 2, \ldots, k$. Note that column $\leq n$ corresponds to enumerating all simple paths of all sizes. The results in Table 2.2 underline the diminishing rate of returns with path size. Clearly, we find more violated inequalities with longer paths; however, the increase in the gap improvement diminishes. The tradeoff between searching for more paths and solution time and quality is most visible in the `time` column: the elapsed time first decreases and then increases with increasing efforts for finding paths. Therefore, in the remainder of the experiments, in order to find a balance between time spent for finding paths and the solution quality, for each given extreme point solution, we stop searching for paths if we fail to find violated inequalities for four consecutive path sizes.

Table 2.2: Effect of path length on the performance of `lspi`.

			`gap imp %`					`cuts`					`time`				
c	f	`init gap`	$=1$	≤ 2	≤ 3	≤ 4	$\leq n$	$=1$	≤ 2	≤ 3	≤ 4	$\leq n$	$=1$	≤ 2	≤ 3	≤ 4	$\leq n$
2	100	11.3	82.4	95.2	98.5	98.6	98.6	104	183	227	247	267	0	0	1	1	1
	200	10.2	58.1	70.3	72.9	74.3	74.8	140	269	367	433	618	1299	258	71	100	88
	500	13.2	49.9	61.4	65.0	66.2	66.9	124	224	298	361	527	142	14	7	7	21
	1000	15.8	53.2	62.3	65.0	66.3	66.6	121	212	274	326	507	29	11	8	8	21
3	100	84.7	86.7	97.6	98.7	98.7	98.7	86	142	163	171	173	0	0	0	0	1
	200	17.2	78.9	89.3	90.6	90.8	91.0	132	240	292	325	346	21	1	1	1	3
	500	18.4	48.3	55.9	57.9	58.3	58.4	152	262	358	410	558	1944	895	479	511	617
	1000	20.3	43.5	51.9	54.7	55.3	55.6	143	243	309	355	449	279	232	204	198	244
4	100	14.9	91.5	99.0	99.1	99.1	99.1	81	118	128	130	131	0	0	0	0	0
	200	87.7	85.5	95.1	95.7	95.9	96.2	97	159	183	192	201	0	0	0	1	1
	500	20.1	47.7	52.7	54.5	54.9	54.9	176	285	364	416	494	2273	1952	951	1166	1093
	1000	23.6	42.0	46.3	48.3	49.1	49.3	157	252	315	365	450	1703	659	408	570	628
5	100	7.7	92.6	99.7	99.3	99.3	99.3	77	104	111	112	113	0	0	0	0	0
	200	47.9	89.6	97.5	97.7	97.7	97.9	91	134	144	146	148	0	0	0	0	0
	500	25.6	57.0	60.4	62.2	62.6	62.8	168	267	327	354	383	1197	1360	1096	708	688
	1000	27.0	41.5	45.9	47.7	48.3	48.5	169	263	328	383	490	1291	1147	1114	1029	1272
Average		**27.8**	**65.5**	**73.8**	**75.5**	**76.0**	**76.2**	**126**	**210**	**262**	**295**	**366**	**636**	**408**	**271**	**269**	**292**

We next compare `lspi` with lifted single node flow cover (`fc`) and uncapacitated path inequalities (`uc`). The results with the lifted flow cover inequalities are reported in Table 3.3 under columns `fc` and are equivalent to inequality (2.17) where the path size is equal to one. Under columns `uc`, we report the uncapacitated path inequalities of van Roy and Wolsey (1985) that are of the form

$$y(S^+) \leq \sum_{j=k}^{\ell} d_{j\ell} x(S_j^+) + y(E^-), \tag{2.53}$$

where $N = [k, \ell]$ is the path and $S^+ \subseteq E^+$. We enumerate all paths in a given instance and given a fractional solution (x^*, y^*) we use a simple comparison to select set S^+ as $S_j^+ = \{t \in E_j^+ : \; y_t^* > d_{j\ell} x_t^*\}$ where $S_j^+ = S^+ \cap E_j^+$.

We observe in Table 3.3 that `lspi` helps to reduce the integrality gap by additional 11% and 45% on average compared to `fc` and `uc`, respectively. However for some instances, where both f and c are large, `uc` performs better than both `lspi` and `fc`. A positive effect of larger c on the performance of `uc` is expected since these cases are closer to being uncapacitated.

Although Table 3.3 clearly shows the positive impact of exploiting the path structure together with the arc capacities, in the final computational experiment, we test the marginal contribution of `lspi` over CPLEX fixed-charge network cuts, namely flow cover, flow path and multi-commodity flow cuts. In Table 2.4, we present the results of the experiments where we use both `lspi` and CPLEX flow cover, flow path and MCF cuts under columns `lspix`, and the results with only CPLEX flow cover, flow path and MCF cuts under columns `cpx`. The positive effect of `lspi` is most apparent for smaller f values. On average, gap improvement increases by 3% when `lspi` is used. The presence of `lspi` decreases the number of branch and bound nodes explored by 50% and increases the number of instances that is solved to optimality within time/memory limit. Under the `cuts` column we present total number of cuts flow, flow path and MCF cuts added. On average, 74% of all the cuts added in `lspix` are user cuts (i.e., `lspi`). Furthermore, we observe that total number of flow, flow path and MCF cuts added decreases by 38% on average under the presence of `lspi`. Although, we observe that `lspix` took a few more seconds to terminate the algorithm on average, in 56% of the instances `lspi` terminated the algorithm faster than `cpx`.

Table 2.3: Comparison of lifted submodular path inequalities (`lspi`) with lifted flow cover inequalities (`fc`) and uncapacitated path inequalities (`uc`).

			gap imp %			cuts			nodes			time (endgap:unslv)		
c	f	init gap	lspi	fc	uc	lspi	fc	uc	lspi	fc	uc	lspi	fc	uc
2	100	11.3	98.6	82.4	2.4	267	104	1	10	1955	16805127	1	0	1548 (0.7:1)
	200	10.2	74.8	58.1	35.7	618	140	241	170919	9295457	13166197	80	1299 (0.5:2)	1835 (2.5:5)
	500	13.2	66.9	49.9	49.1	522	124	1268	12622	973479	1404240	9	142	896
	1000	15.8	66.6	53.2	47.1	507	121	1164	21795	185758	594313	13	29	407
3	100	84.7	98.7	86.7	0	173	86	0	4	117	1228852	0	0	101
	200	17.2	91	78.9	11.6	345	132	37	350	131633	13026690	2	21	1264 (4.7:5)
	500	18.4	58.4	48.3	55.1	558	152	1516	1965086	13529271	5499897	661	1944 (1.0:3)	3312 (2.0:4)
	1000	20.3	55.6	43.5	50.7	444	143	1782	745041	2052371	2319528	226	279	2275 (0.8:2)
4	100	14.9	99.1	91.5	0	131	81	0	3	29	50625	0	0	5
	200	87.7	96.1	85.5	3.1	199	97	7	19	512	3460744	1	0	335
	500	20.1	54.9	47.7	63.8	494	176	1071	3775955	15171027	5572322	1080	2273 (1.4:4)	2704 (0.6:3)
	1000	23.6	49.3	42	60.4	450	157	1984	2077797	12365215	1867691	570	1703 (0.1:1)	2026 (0.5:2)
5	100	7.7	99.3	92.6	0	113	77	0	3	18	32383	0	0	4
	200	47.9	97.9	89.6	0.2	148	91	1	6	1097	1392639	0	0	138
	500	25.6	62.8	57	46.2	383	168	305	2678953	8274212	13353250	704	1197 (1.2:2)	2396 (3.0:5)
	1000	27	48.5	41.5	68	488	169	1977	4864049	9775043	1973802	1189 (0.2:1)	1291 (1.5:2)	2226 (0.5:3)
Average		**27.8**	**76.2**	**65.5**	**30.8**	**365**	**126**	**710**	**1019538**	**4484825**	**5109269**	**284 (0.0:0)**	**636 (0.4:1)**	**1342 (1.0:2)**

Table 2.4: Experiments with CPLEX flow cover, flow path and MCF cuts.

				gap imp %		cuts		nodes		time (endgap:unslv)	
n	c	f	init gap	cpx	lspix	cpx	lspix	cpx	lspix	cpx	lspix
50	2	100	11.3	94.2	99.6	170	330	67	3	0	1
		200	10.2	86.8	87.9	184	680	122456	8393	22	7
		500	13.2	85.4	82.4	146	666	9528	2261	2	6
		1000	15.8	80.5	81.5	130	605	12676	3277	2	6
	3	100	84.7	91.2	99.8	138	229	22	1	0	0
		200	17.2	92.9	97.4	197	374	707	43	0	1
		500	18.4	78.7	79.7	152	603	819590	146769	123	58
		1000	20.3	79.2	77.2	147	589	319068	87354	47	36
	4	100	14.9	90.4	99.7	119	191	18	1	0	0
		200	87.7	94.3	99.2	156	258	38	3	0	0
		500	20.1	82.6	83.2	159	559	186654	39244	28	17
		1000	23.6	79	76.8	142	566	297505	95019	42	37
	5	100	7.7	92.6	99.8	106	173	11	2	0	0
		200	47.9	92.8	99.7	141	208	22	2	0	0
		500	25.6	86.2	87.9	193	433	211945	22408	34	8
		1000	27	81.6	80.3	151	583	521651	153655	74	53
75	2	100	10.4	90	99.4	223	447	201	4	0	1
		200	10	87.2	90.6	266	909	1202846	13022	323	18
		500	13.9	79.1	76.3	187	824	1694669	812991	420	689
		1000	14	82.3	82.5	210	855	247773	38970	57	35
	3	100	502.8	90.3	99.6	209	361	289	1	0	1
		200	18.7	91.7	97.8	269	494	3814	75	1	2
		500	17.1	79.9	79.5	231	944	8210162	5065675	1560 (0.9:2)	2437 (0.2:2)
		1000	20.5	73.8	73.1	212	766	10467362	4429938	2021 (0.8:2)	1906 (0.0:1)
	4	100	14.3	92.4	99.9	185	300	27	1	0	1
		200	124.9	92.4	99.6	235	370	411	4	0	1
		500	21.4	78.1	79.4	244	816	11526938	5609289	2055 (2.2:4)	2134 (1.1:3)
		1000	24.6	75.9	76.8	222	889	8378586	5032511	1392 (1.8:3)	2312 (0.7:3)
	5	100	7.7	94.1	100	165	262	19	0	0	0
		200	61.2	90.2	99.8	201	330	136	3	0	1
		500	24.4	87.7	89.9	289	699	4926824	255153	1224 (0.1:1)	106
		1000	27.2	79.5	80.2	217	809	10433607	3940582	1690 (1.7:3)	1492 (0.6:1)
Average			**42.5**	**86.0**	**89.3**	**187**	**535**	**1862363**	**804896**	**347 (0.2:0)**	**355 (0.1:0)**

Chapter 3

Path Cover and Path Pack Inequalities for the Capacitated Fixed-Charge Network Flow Problem

Given a directed multigraph with demand or supply on the nodes, and capacity, fixed and variable cost of flow on the arcs, the capacitated fixed-charge network flow (CFNF) 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.

There are numerous polyhedral studies on the fixed-charge network flow problem. In a seminal paper Wolsey (1989) introduces the so-called submodular inequalities, which subsume almost all valid inequalities known for capacitated fixed-charge networks. Although the submodular inequalities are very general, their coefficients are defined implicitly through value functions. In this chapter, we give explicit valid inequalities that simultaneously make use of the path substructures of the network as well as the arc capacities.

For the *uncapacitated* fixed-charge network flow problem, van Roy and Wolsey (1985) give flow path inequalities that are based on path substructures. Rardin and Wolsey (1993) introduce a new family of dicut inequalities and show that they describe the projection of an extended multicommodity formulation onto the original variables of fixed-charge network flow problem. Ortega and Wolsey (2003) present a computational study on the performance of path and cut-set (dicut) inequalities.

For the *capacitated* fixed-charge network flow problem, almost all known valid inequalities are based on single-node relaxations. Padberg et al. (1985), van Roy and Wolsey (1986), Gu et al. (1999) give flow cover, generalized flow cover and lifted flow cover inequalities. Stallaert (1997) introduces a complementary class to generalized flow cover inequalities and Atamtürk (2001) describes lifted flow pack inequalities. Atamtürk et al. (2016) give generalizations of flow cover inequalities from three partitions. Both uncapacitated path inequalities and capacitated flow cover and flow pack inequalities are highly valuable in solving a host of practical problems and are part of the suite of cutting planes implemented in modern mixed-integer programming solvers.

The *path* structure arises naturally in network models of the lot-sizing problem. Atamtürk and Muñoz (2004) introduce valid inequalities for the capacitated lot-sizing problems with infinite inventory capacities. Atamtürk and Küçükyavuz (2005) give valid inequalities for the lot-sizing problems with finite inventory and infinite production capacities. Van Vyve (2013) introduces path-modular inequalities for the uncapacitated fixed charge transportation problems. These inequalities are derived from a value function that is neither globally submodular nor supermodular but that exhibits sub or supermodularity under certain set selections. Van Vyve and Ortega (2004) and Gade and Küçükyavuz (2011) give valid inequalities and extended formulations for uncapacitated lot-sizing with fixed charges on stocks. For uncapacitated lot-sizing with backlogging, Pochet and Wolsey (1988) and Pochet and Wolsey (1994) provide valid inequalities and Küçükyavuz and Pochet (2009) give an explicit description of the convex hull.

Contributions

In this chapter we consider a generic path relaxation, with supply and/or demand nodes and capacities on incoming and outgoing arcs. By exploiting the path substructure of the network and introducing notions of *path cover* and *path pack* we provide two explicitly-described subclasses of the submodular inequalities. The most important consequence of the explicit derivation is that the coefficients of the submodular inequalities on a path can be computed efficiently. In particular, we show that the coefficients of these inequalities can be computed by solving max-flow/min-cut problems parametrically over the path. Moreover, we show that *all* of the coefficients can be computed with a single linear-time algorithm. For a path with a single node, the inequalities reduce to the well-known flow cover and flow pack inequalities. In addition, the path cover and path pack inequalities dominate flow cover and flow pack inequalities for the corresponding single node relaxation of a path obtained by merging the path into a single node. We give necessary and sufficient facet-defining conditions. Finally, we report on computational experiments demonstrating the effectiveness of the proposed inequalities when used as cuts in a branch-and-cut algorithm.

Outline

The remainder of this chapter is organized as follows: In Section 3.1, we describe the capacitated fixed-charge flow problem on a path, its formulation and the assumptions we make. In Section 3.2, we review the submodular inequalities, discuss their computation on a path, and introduce two explicit subclasses: path cover inequalities and path pack inequalities. In Section 3.3, we analyze sufficient and necessary facet-defining conditions. In Section 3.4, we present computational experiments showing the effectiveness of the path cover and path pack inequalities compared to other network inequalities.

3.1 Capacitated fixed-charge network flow on a path

Let $G = (N', A)$ be a directed multigraph with nodes N' and arcs A. Let s_N and t_N be the source and the sink nodes of G. Let $N := N' \setminus \{s_N, t_N\}$. Without loss of generality, we label $N := \{1, \ldots, n\}$ such that a directed *forward path* arc exists from node i to node $i+1$ and a directed *backward path* arc exists from node $i+1$ to node i for each node $i = 1, \ldots, n-1$ (see Figure 3.1 for an illustration). In Remarks 3.1 and 3.2, we discuss how to obtain a "path" graph G from a more general directed multigraph.

Let $E^+ = \{(i,j) \in A : i = s_N,\ j \in N\}$ and $E^- = \{(i,j) \in A : i \in N,\ j = t_N\}$. Moreover, let us partition the sets E^+ and E^- such that $E_k^+ = \{(i,j) \in A : i \notin N, j = k\}$ and $E_k^- = \{(i,j) \in A : i = k, j \notin N\}$ for $k \in N$. We refer to the arcs in E^+ and E^- as *non-path arcs*. Finally, let $E := E^+ \cup E^-$ be the set of all non-path arcs. For convenience, we generalize this set notation scheme. Given an arbitrary subset of non-path arcs $Y \subseteq E$, let $Y_j^+ = Y \cap E_j^+$ and $Y_j^- = Y \cap E_j^-$.

Remark 3.1. Given a directed multigraph $\tilde{G} = (\tilde{N}, \tilde{A})$ with nodes $\tilde{N}$, arcs $\tilde{A}$ and a path that passes through nodes N, we can construct G as described above by letting $E^+ = \{(i,j) \in \tilde{A} : i \in \tilde{N} \setminus N, j \in N\}$ and $E^- = \{(i,j) \in \tilde{A} : i \in N, j \in \tilde{N} \setminus N\}$ and letting all the arcs in E^+ be the outgoing arcs from a dummy source s_N and all the arcs in E^- to be incoming to a dummy sink t_N.

Remark 3.2. If there is an arc $t = (i,j)$ from node $i \in N$ to $j \in N$, where $|i-j| > 1$, then we construct a relaxation by removing arc t, and replacing it with two arcs $t^- \in E_i^-$ and $t^+ \in E_j^+$. If there are multiple arcs from node i to node j, one can repeat the same procedure.

Throughout the chapter, we use the following notation: Let $[k,j] = \{k, k+1, \ldots, j\}$ if $k \le j$ and $\emptyset$ otherwise, $c(S) = \sum_{t \in S} c_t$, $y(S) = \sum_{t \in S} y_t$, $(a)^+ = \max\{0, a\}$ and $d_{kj} = \sum_{t=k}^{j} d_t$ if $j \ge k$ and 0 otherwise. Moreover, let $\dim(A)$ denote the dimension of a polyhedron A and $\text{conv}(S)$ be the convex hull of a set S.

The capacitated fixed-charge network flow problem on a path can be formulated as a mixed-integer optimization problem. Let d_j be the demand at node $j \in N$. We call a node $j \in N$ a demand node if $d_j \ge 0$ and a supply node if $d_j < 0$. Let the flow on forward path arc $(j, j+1)$ be represented by i_j with an upper bound u_j for $j \in N \setminus \{n\}$. Similarly, let the flow on backward path arc $(j+1, j)$ be represented by r_j with an upper bound b_j for $j \in N \setminus \{n\}$. Let y_t be the amount of flow on arc $t \in E$ with an

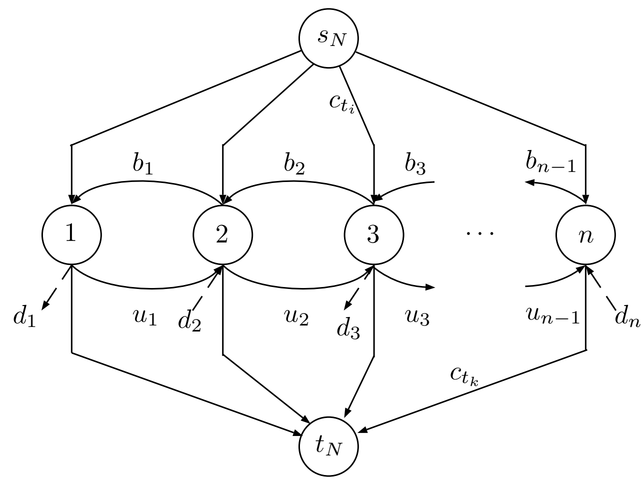


Figure 3.1: Fixed-charge network representation of a path.

upper bound c_t. Define binary variable x_t to be 1 if $y_t > 0$, and zero otherwise for all $t \in E$. An arc t is *closed* if $x_t = 0$ and *open* if $x_t = 1$. Moreover, let f_t be the fixed cost and p_t be the unit flow cost of arc t. Similarly, let h_j and g_j be the costs of unit flow, on forward and backward arcs $(j, j+1)$ and $(j+1, j)$ respectively for $j \in N \setminus \{n\}$. Then, the problem is formulated as

$$
\begin{aligned}
& \min && \sum_{t \in E} (f_t x_t + p_t y_t) + \sum_{j \in N} (h_j i_j + g_j r_j) && \text{(3.1a)} \\
& \text{s. t.} && i_{j-1} - r_{j-1} + y(E_j^+) - y(E_j^-) - i_j + r_j = d_j, \quad j \in N, && \text{(3.1b)} \\
& && 0 \le y_t \le c_t x_t, \quad t \in E, && \text{(3.1c)} \\
& && 0 \le i_j \le u_j, \quad j \in N, && \text{(3.1d)} \\
& \text{(F3.1)} && 0 \le r_j \le b_j, \quad j \in N, && \text{(3.1e)} \\
& && x_t \in \{0, 1\}, \quad t \in E, && \text{(3.1f)} \\
& && i_0 = i_n = r_0 = r_n = 0. && \text{(3.1g)}
\end{aligned}
$$

Let $\mathcal{P}$ be the set of feasible solutions of (F3.1). Figure 3.1 shows an example network representation of (F3.1).

Throughout we make the following assumptions on (F3.1):

(A.1) The set $\mathcal{P}_t = \{(x, y, i, r) \in \mathcal{P} : x_t = 0\} \neq \emptyset$ for all $t \in E$,

(A.2) $c_t > 0$, $u_j > 0$ and $b_j > 0$ for all $t \in E$ and $j \in N$,

(A.3) $c_t \le d_{1n} + c(E^-)$ for all $t \in E^+$,

(A.4) $c_t \le b_{j-1} + u_j + (d_j)^+ + c(E_j^-)$, for all $j \in N, t \in E_j^+$,

(A.5) $c_t \le b_j + u_{j-1} + (-d_j)^+ + c(E_j^+)$ for all $j \in N, t \in E_j^-$.

Assumptions (A.1)–(A.2) ensure that $\dim\big(\text{conv}(\mathcal{P})\big) = 2|E| + |N| - 2$. If (A.1) does not hold for some $t \in E$, then $x_t = 1$ for all points in $\mathcal{P}$. Similarly, if (A.2) does not hold, the flow on such an arc can be fixed to zero. Finally, assumptions (A.3)–(A.5) are without loss of generality. An upper bound on y_t can be obtained directly from the flow balance equalities (3.1b) by using the upper and lower bounds of the other flow variables that appear in the same constraint. As a result, the flow values on arcs $t \in E$ cannot exceed the capacities implied by (A.3)–(A.5).

Next, we review the submodular inequalities that are valid for any capacitated fixed-charge network flow problem. Furthermore, using the path structure, we provide an $O(|E| + |N|)$ time algorithm to compute their coefficients.

3.2 Submodular inequalities on paths

Let $S^+ \subseteq E^+$ and $L^- \subseteq E^-$. Wolsey (1989) shows that the value function of the following optimization problem is submodular:

$$
\begin{aligned}
v(S^+, L^-) = \max \quad & \sum_{t \in E} a_t y_t && \text{(3.2a)}\\
\text{s. t.} \quad & i_{j-1} - r_{j-1} + y(E_j^+) - y(E_j^-) - i_j + r_j \le d_j, \quad j \in N, && \text{(3.2b)}\\
& 0 \le i_j \le u_j, \quad j \in N, && \text{(3.2c)}\\
& 0 \le r_j \le b_j, \quad j \in N, && \text{(3.2d)}\\
\text{(F3.2)} \qquad & 0 \le y_t \le c_t, \quad t \in E, && \text{(3.2e)}\\
& i_0 = i_n = r_0 = r_n = 0, && \text{(3.2f)}\\
& y_t = 0, \quad t \in (E^+ \setminus S^+) \cup L^-, && \text{(3.2g)}
\end{aligned}
$$

where $a_t \in \{0, 1\}$ for $t \in E^+$ and $a_t \in \{0, -1\}$ for $t \in E^-$. Let $\mathcal{Q}$ denote the set of feasible solutions of (F3.2).

We call the sets S^+ and L^- that are used in the definition of $v(S^+, L^-)$ the *objective sets*. For ease of notation, we also represent the objective sets as $C := S^+ \cup L^-$. Following this notation, let $v(C) := v(S^+, L^-)$, $v(C \setminus \{t\}) = v(S^+ \setminus \{t\}, L^-)$ for $t \in S^+$ and $v(C \setminus \{t\}) = v(S^+, L^- \setminus \{t\})$ for $t \in L^-$. Similarly, let $v(C \cup \{t\}) = v(S^+ \cup \{t\}, L^-)$, for $t \in S^+$ and $v(C \cup \{t\}) = v(S^+, L^- \cup \{t\})$ for $t \in L^-$. Moreover, let

$$\rho_t(C) = v(C \cup \{t\}) - v(C)$$

be the marginal contribution of adding an arc t to C with respect to the value function v. Wolsey (1989) shows that the following inequalities are valid for $\mathcal{P}$:

$$\sum_{t \in E} a_t y_t + \sum_{t \in C} \rho_t(C \setminus \{t\})(1 - \bar{x}_t) \le v(C) + \sum_{t \in E \setminus C} \rho_t(\emptyset)\bar{x}_t, \tag{3.3}$$

$$\sum_{t \in E} a_t y_t + \sum_{t \in C} \rho_t(E \setminus \{t\})(1 - \bar{x}_t) \leq v(C) + \sum_{t \in E \setminus C} \rho_t(C)\bar{x}_t, \tag{3.4}$$

where the variable $\bar{x}_t$ is defined as

$$\bar{x}_t = \begin{cases} x_t, & t \in E^+ \\ 1 - x_t, & t \in E^-. \end{cases}$$

In fact, inequalities (3.3) and (3.4) are also valid for fixed-charge network flow formulations where the flow balance constraints (3.1b) are replaced with constraints (3.2b). However, in this chapter, we focus on formulations with flow balance equalities (3.1b).

We refer to submodular inequalities (3.3) and (3.4) derived for path structures as *path inequalities*. In this chapter, we consider sets S^+ and L^- such that (F3.2) is feasible for all objective sets C and $C \setminus \{t\}$ for all $t \in C$.

3.2.1 Equivalence to the maximum flow problem

Define sets K^+ and K^- such that the coefficients of the objective function (3.2a) are:

$$a_t = \begin{cases} 1, & t \in K^+ \\ -1, & t \in K^- \\ 0, & \text{otherwise}, \end{cases} \tag{3.5}$$

where $S^+ \subseteq K^+ \subseteq E^+$ and $K^- \subseteq E^- \setminus L^-$. We refer to the sets K^+ and K^- as *coefficient sets*. Let the set of arcs with zero coefficients in (3.2a) be represented by $\bar{K}^+ = E^+ \setminus K^+$ and $\bar{K}^- = E^- \setminus K^-$. Given a selection of coefficients as described in (3.5), we claim that (F3.2) can be transformed to a maximum flow problem. We first show this result assuming $d_j \geq 0$ for all $j \in N$. Then, in Appendix C, we show that the nonnegativity of demand is without loss of generality for the derivation of the inequalities.

Proposition 3.1. Let $S^+ \subseteq E^+$ and $L^- \subseteq E^-$ be the objective sets in (F3.2) and let $\mathcal{Y}$ be the nonempty set of optimal solutions of (F3.2). If $d_j \geq 0$ for all $j \in N$, then there exists at least one optimal solution $(\mathbf{y}^*, \mathbf{r}^*, \mathbf{i}^*) \in \mathcal{Y}$ such that $y^*_t = 0$ for $t \in \bar{K}^+ \cup K^- \cup L^-$.

Proof. Observe that $y^*_t = 0$ for all $t \in E^+ \setminus S^+$, due to constraints (3.2g). Since $\bar{K}^+ \subseteq E^+ \setminus S^+$, $y^*_t = 0$, for $t \in \bar{K}^+$ from feasibility of (F**??**). Similarly, $y^*_t = 0$ for all $t \in L^-$ by constraints (3.2g).

Now suppose that, $y^*_t = \epsilon > 0$ for some $t \in K^-_j$ (i.e., $a_t = -1$ for arc t in (F3.2)). Let the slack value at constraint (3.2b) for node j be

$$s_j = d_j - \left[i^*_{j-1} - r^*_{j-1} + y^*(E^+_j) - y^*(E^-_j \setminus \{t\}) - y^*_t - i^*_j + r^*_j \right].$$

If $s_j \geq \epsilon$, then decreasing y_t^* by ϵ both improves the objective function value and conserves the feasibility of flow balance inequality (3.2b) for node j, since $s_j - \epsilon \geq 0$.

If $s_j < \epsilon$, then decreasing y_t^* by ϵ violates flow balance inequality since $s_j - \epsilon < 0$. In this case, there must exist a simple directed path P from either the source node s_N or a node $k \in N \setminus \{j\}$ to node j where all arcs have at least a flow of $(\epsilon - s_j)$. This is guaranteed because, $s_j < \epsilon$ implies that, without the outgoing arc t, there is more incoming flow to node j than outgoing. Then, notice that decreasing the flow on arc t and all arcs in path P by $\epsilon - s_j$ conserves feasibility. Moreover, the objective function value either remains the same or increases, because decreasing y_t by $\epsilon - s_j$ increases the objective function value by $\epsilon - s_j$ and the decreasing the flow on arcs in P decreases it by at most $\epsilon - s_j$. At the end of this transformation, the slack value s_j does not change, however; the flow at arc t is now $y_t^* = s_j$ which is equivalent to the first case that is discussed above. As a result, we obtain a new solution to (F3.2) where $y_t^* = 0$ and the objective value is at least as large.

□

Proposition 3.2. If $d_j \geq 0$ for all $j \in N$, then (F3.2) is equivalent to a maximum flow problem from source s_N to sink t_N on graph G.

Proof. At the optimal solution of problem (F3.2) with objective set (S^+, L^-), the decision variables y_t, for $t \in (E^+ \setminus S^+) \cup K^- \cup L^-$ can be assumed to be zero due to Proposition 3.1 and constraints (3.2g). Then, these variables can be dropped from (F3.2) since the value $v(S^+, L^-)$ does not depend on them and formulation (F3.2) reduces to

$$v(S^+, L^-) = \max \big\{ y(S^+) : \; i_{j-1} - r_{j-1} + y(S_j^+) - y(\bar{K}_j^-) - i_j + r_j \leq d_j, \; j \in N, \\ (3.2c) - (3.2f) \big\}. \tag{3.6}$$

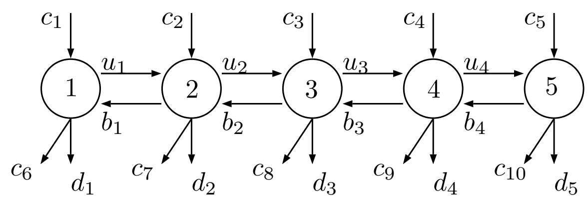
Now, we reformulate (3.6) by representing the left hand side of the flow balance constraint by a new nonnegative decision variable z_j that has an upper bound of d_j for each $j \in N$:

$$\max \big\{ y(S^+) : \; i_{j-1} - r_{j-1} + y(S_j^+) - y(\bar{K}_j^-) - i_j + r_j = z_j, \; j \in N, \\ 0 \leq z_j \leq d_j, \; j \in N, \quad (3.2c) - (3.2f) \big\}.$$

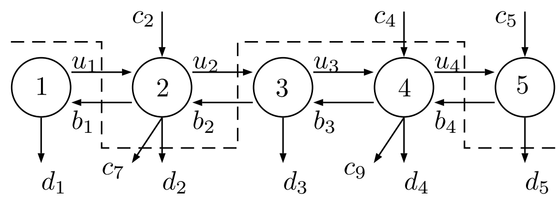
The formulation above is equivalent the maximum flow formulation from the source node s_N to the sink node t_N for the path structures we are considering in this chapter.

□

Under the assumption that $d_j \geq 0$ for all $j \in N$, Proposition 3.1 and Proposition 3.2 together show that the optimal objective function value $v(S^+, L^-)$ can be computed by solving a maximum flow problem from source s_N to sink t_N. We generalize this



(a) A path graph with $E^+ = [1, 5]$ and $E^- = [6, 10]$.



(b) An $s_N - t_N$ cut for set $S^+ = \{2, 4, 5\}$, $L^- = \{10\}$ and $\bar{K}^- = \{7, 9, 10\}$.

Figure 3.2: An example of an $s_N - t_N$ cut.

result in Appendix C for node sets N such that $d_j < 0$ for some $j \in N$. As a result, obtaining the explicit coefficients of submodular inequalities (3.3) and (3.4) reduces to solving $|E| + 1$ maximum flow problems. For a general underlying graph, solving $|E| + 1$ maximum flow problems would take $O(|E|^2|N|)$ time (e.g., see King et al. (1994)), where $|E|$ and $|N|$ are the number of arcs and nodes, respectively. In the following subsection, by utilizing the equivalence of maximum flow and minimum cuts and the path structure, we show that all coefficients of (3.3) and (3.4) can be obtained in $O(|E| + |N|)$ time using dynamic programming.

3.2.2 Computing the coefficients of the submodular inequalities

Throughout the chapter, we use minimum cut arguments to find the explicit coefficients of inequalities (3.3) and (3.4). Figure 3.2a illustrates an example where $N = [1, 5]$, $E^+ = [1, 5]$, $E^- = [6, 10]$ and in Figure 3.2b, we give an example of an $s_N - t_N$ cut for $S^+ = \{2, 4, 5\}$, $L^- = \{10\}$ and $\bar{K}^- = \{7, 9, 10\}$. The dashed line in Figure 3.2b represents a cut that corresponds to the partition $\{s_N, 2, 5\}$ and $\{t_N, 1, 3, 4\}$ with a value of $b_1 + d_2 + c_7 + u_2 + c_4 + b_4 + d_5$. Moreover, we say that a cut *passes below node* j if j is in the source partition and *passes above node* j if j is in the sink partition.

Let α_j^u and α_j^d be the minimum value of a cut on nodes $[1, j]$ that passes above and below node j, respectively. Similarly, let β_j^u and β_j^d be the minimum values of cuts on nodes $[j, n]$ that passes above and below node j respectively. Finally, let

$$S^- = E^- \setminus (K^- \cup L^-),$$

where K^- is defined in (3.5). Recall that S^+ and L^- are the given objective sets. Given the notation introduced above, all of the arcs in sets S^- and L^- have a coefficient zero in (F3.2). Therefore, dropping an arc from L^- is equivalent to adding that arc to S^-. We compute $\alpha_j^{\{u,d\}}$ by a forward recursion and $\beta_j^{\{u,d\}}$ by a backward recursion:

$$\alpha_j^u = \min\{\alpha_{j-1}^d + u_{j-1}, \alpha_{j-1}^u\} + c(S_j^+) \tag{3.7}$$
$$\alpha_j^d = \min\{\alpha_{j-1}^d, \alpha_{j-1}^u + b_{j-1}\} + d_j + c(S_j^-), \tag{3.8}$$

where $\alpha_0^u = \alpha_0^d = 0$ and

$$\beta_j^u = \min\{\beta_{j+1}^u, \beta_{j+1}^d + b_j\} + c(S_j^+) \tag{3.9}$$
$$\beta_j^d = \min\{\beta_{j+1}^u + u_j, \beta_{j+1}^d\} + d_j + c(S_j^-), \tag{3.10}$$

where $\beta_{n+1}^u = \beta_{n+1}^d = 0$.

Let m_j^u and m_j^d be the values of minimum cuts for nodes $[1, n]$ that pass above and below node j, respectively. Notice that

$$m_j^u = \alpha_j^u + \beta_j^u - c(S_j^+) \tag{3.11}$$

and

$$m_j^d = \alpha_j^d + \beta_j^d - d_j - c(S_j^-). \tag{3.12}$$

For convenience, let

$$m_j := \min\{m_j^u, m_j^d\}.$$

Notice that m_j is the minimum of the minimum cut values that passes above and below node j. Since the minimum cut corresponding to $v(C)$ has to pass either above or below node j, m_j is equal to $v(C)$ for all $j \in N$. As a result, the minimum cut (or maximum flow) value for the objective set $C = S^+ \cup L^-$ is

$$v(C) = m_1 = \cdots = m_n. \tag{3.13}$$

Proposition 3.3. All values m_j, for $j \in N$, can be computed in $O(|E| + |N|)$ time.

Obtaining the explicit coefficients of inequalities (3.3) and (3.4) also requires finding $v(C \setminus \{t\})$ for $t \in C$ and $v(C \cup \{t\})$ for $t \notin C$ in addition to $v(C)$. It is important to note that we do not need to solve the recursions above repeatedly. Once the values m_j^u and m_j^d are obtained for the set C, the marginals $\rho_t(C \setminus \{t\})$ and $\rho_t(C)$ can be found in $O(1)$ time for each $t \in E$.

We use the following observation while computing the marginal values $\rho_t(C)$ and $\rho_t(C \setminus \{t\})$ as a function of m_j^u and m_j^d for $t \in E_j^+ \cup E_j^-$ and $j \in N$.

Observation 2. Let $c \geq 0$ and $d := (b - a)^+$, then,

1. $\min\{a+c, b\} - \min\{a, b\} = \min\{c, d\}$,

2. $\min\{a, b\} - \min\{a, b-c\} = (c-d)^+$.

In the remainder of this section, we give a linear-time algorithm to compute the coefficients ρ_t for inequalities (3.3) and (3.4) explicitly for paths.

Coefficients of inequality (3.3): Path cover inequalities

Let S^+ and L^- be the objective sets in (F3.2) and $S^- \subseteq E^- \setminus L^-$. We select the coefficient sets in (3.5) as $K^+ = S^+$ and $K^- = E^- \setminus (L^- \cup S^-)$ to obtain the explicit form of inequality (3.3). As a result, the set definition of $S^- = E^- \setminus (K^- \cup L^-)$ is conserved.

Definition Let the coefficient sets in (3.5) be selected as above and (S^+, L^-) be the objective set. The set (S^+, S^-) is called a *path cover* for the node set N if

$$v(S^+, L^-) = d_{1n} + c(S^-).$$

If we assume that the set (S^+, S^-) is a path cover for N, then by definition,

$$v(C) = m_1 = \cdots = m_n = d_{1n} + c(S^-)$$

in inequality (3.3). After obtaining the values m_j^u and m_j^d for a node $j \in N$ using recursions in (4.3)–(4.6), it is trivial to find the minimum cut value after dropping an arc t from S_j^+:

$$v(C \setminus \{t\}) = \min\{m_j^u - c_t, m_j^d\}, \quad t \in S_j^+, \quad j \in N.$$

Similarly, dropping an arc $t \in L_j^-$ results in the minimum cut value:

$$v(C \setminus \{t\}) = \min\{m_j^u, m_j^d + c_t\}, \quad t \in L_j^-, \quad j \in N.$$

Using Observation 2, we obtain the marginal values

$$\rho_t(C \setminus \{t\}) = (c_t - \lambda_j)^+, \quad t \in S_j^+, \quad j \in N$$

and

$$\rho_t(C \setminus \{t\}) = \min\{\lambda_j, c_t\}, \quad t \in L_j^-, \quad j \in N$$

where

$$\lambda_j = (m_j^u - m_j^d)^+, \quad j \in N.$$

On the other hand, all the coefficients $\rho_t(\emptyset) = 0$ for arcs $t \in E \setminus C$. First, notice that, for $t \in E^+ \setminus S^+$, $v(\{t\}) = 0$, because the coefficient $a_t = 0$ for $t \in E^+ \setminus S^+$.

Furthermore, $v(\{t\}) = 0$ for $t \in E^- \setminus L^-$, since all incoming arcs would be closed for an objective set $(\emptyset, \{t\})$. As a result, inequality (3.3) for the objective set (S^+, L^-) can be written as

$$y(S^+) + \sum_{j \in N} \sum_{t \in S_j^+} (c_t - \lambda_j)^+ (1 - x_t) \leq d_{1n} + c(S^-) + \sum_{j \in N} \sum_{t \in L_j^-} \min\{c_t, \lambda_j\} x_t + y(E^- \setminus (L^- \cup S^-)). \quad (3.14)$$

We refer to inequalities (3.14) as *path cover inequalities.*

Remark 3.3. Observe that for a path consisting of a single node $N = \{j\}$ with demand $d := d_j > 0$, the path cover inequalities (3.14) reduce to the flow cover inequalities (Padberg et al., 1985, van Roy and Wolsey, 1986). Suppose that the path consists of a single node $N = \{j\}$ with demand $d := d_j > 0$. Let $S^+ \subseteq E^+$ and $S^- \subseteq E^-$. The set (S^+, S^-) is a flow cover if $\lambda := c(S^+) - d - c(S^-) > 0$ and the resulting path cover inequality

$$y(S^+) + \sum_{t \in S^+} (c_t - \lambda)^+ (1 - x_t) \leq d + c(S^-) + \lambda x(L^-) + y(E^- \setminus L^-) \quad (3.15)$$

is a flow cover inequality.

Proposition 3.4. Let (S^+, S^-) be a path cover for the node set N. The path cover inequality for node set N is at least as strong as the flow cover inequality for the single node relaxation obtained by merging the nodes in N.

Proof. Flow cover and path cover inequalities differ in the coefficients of variables x_t for $t \in S^+$ and $t \in L^-$. Therefore, we compare the values λ_j, $j \in N$ of path cover inequalities (3.14) to the value λ of flow cover inequalities (3.15) and show that $\lambda_j \leq \lambda$, for all $j \in N$. The merging of node set N in graph G is equivalent to relaxing the values u_j and b_j to be infinite for $j \in [1, n-1]$. As a result, the value of the minimum cut that goes above the merged node is $\bar{m}^u = c(S^+)$ and the value of the minimum cut that goes below the merged node is $\bar{m}^d = d_{1n} + c(S^-)$. Now, observe that the recursions in (4.3)–(4.6) imply that the minimum cut values for the original graph G are smaller:

$$m_j^u = \alpha_j^u + \beta_j^u - c(S_j^+) \leq c(S^+) = \bar{m}^u$$

and

$$m_j^d = \alpha_j^d + \beta_j^d - d_j - c(S_j^-) \leq d_{1n} + c(S^-) = \bar{m}^d$$

for all $j \in N$. Recall that the coefficient for the flow cover inequality is $\lambda = (\bar{m}^u - \bar{m}^d)^+$ and the coefficients for path cover inequality are $\lambda_j = (m_j^u - m_j^d)^+$ for $j \in N$. The fact that (S^+, S^-) is a path cover implies that $m_j^d = d_{1n} + c(S^-)$ for all $j \in N$.

Since $\bar{m}^d = m_j^d$ and $m_j^u \leq \bar{m}^u$ for all $j \in N$, we observe that $\lambda_j \leq \lambda$ for all $j \in N$. Consequently, the path cover inequality (3.14) is at least as strong as the flow cover inequality (3.15). □

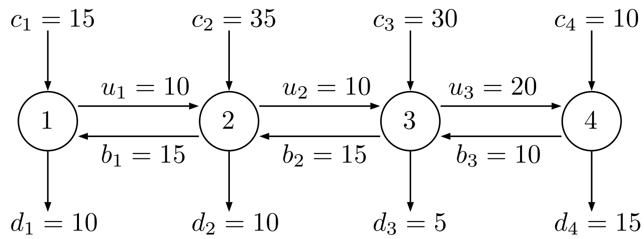


Figure 3.3: A lot-sizing instance with backlogging.

Example 3.1. Consider the lot-sizing instance in Figure 3.3 where $N = [1, 4]$, $S^+ = \{2, 3\}$, $L^- = \emptyset$. Observe that $m_1^u = 45$, $m_1^d = 40$, $m_2^u = 65$, $m_2^d = 40$, $m_3^u = 60$, $m_3^d = 40$, and $m_4^u = 45$, $m_4^d = 40$. Then, $\lambda_1 = 5$, $\lambda_2 = 25$, $\lambda_3 = 20$, and $\lambda_4 = 5$ leading to coefficients 10 and 10 for $(1 - x_2)$ and $(1 - x_3)$, respectively. Furthermore, the maximum flow values are $v(C) = 40$, $v(C \setminus \{2\}) = 30$, and $v(C \setminus \{3\}) = 30$. Then, the resulting path cover inequality (3.14) is

$$y_2 + y_3 + 10(1 - x_2) + 10(1 - x_3) \leq 40, \tag{3.16}$$

and it is facet-defining for conv($\mathcal{P}$) as will be shown in Section 3.3. Now, consider the relaxation obtained by merging the nodes in $[1, 4]$ into a single node with incoming arcs $\{1, 2, 3, 4\}$ and demand $d = 40$. As a result, the flow cover inequalities can be applied to the merged node set. The excess value for the set $S^+ = \{2, 3\}$ is $\lambda = c(S^+) - d = 25$. Then, the resulting flow cover inequality (3.15) is

$$y_2 + y_3 + 10(1 - x_2) + 5(1 - x_3) \leq 40,$$

and it is weaker than the path cover inequality (3.16). □

Coefficients of inequality (3.4): Path pack inequalities

Let S^+ and L^- be the objective sets in (F3.2) and let $S^- \subseteq E^- \setminus L^-$. We select the coefficient sets in (3.5) as $K^+ = E^+$ and $K^- = E^- \setminus (S^- \cup L^-)$ to obtain the explicit form of inequality (3.4). As a result, the set definition of $S^- = E^- \setminus (K^- \cup L^-)$ is conserved.

Definition Let the coefficients in (3.5) be selected as above and (S^+, L^-) be the objective set. The set (S^+, S^-) is called a *path pack* for node set N if

$$v(S^+, L^-) = c(S^+).$$

For inequality (3.4), we assume that the set (S^+, S^-) is a path pack for N and $L^- = \emptyset$ for simplicity. Now, we need to compute the values of $v(C)$, $v(E)$, $v(E \setminus \{t\})$ for $t \in C$ and $v(C \cup \{t\})$ for $t \in E \setminus C$. The value of $v(C \cup \{t\})$ can be obtained using the values m^u_j and m^d_j that are given by recursions (4.3)–(4.6). Then,

$$v(C \cup \{t\}) = \min\{m^u_j + c_t, m^d_j\}, \quad t \in E^+_j \setminus S^+_j, \quad j \in N$$

and

$$v(C \cup \{t\}) = \min\{m^u_j, m^d_j + c_t\}, \quad t \in S^-_j, \quad j \in N.$$

Then, using Observation 2, we compute the marginal values

$$\rho_t(C) = \min\{c_t, \mu_j\}, \quad t \in E^+_j \setminus S^+_j, \quad j \in N$$

and

$$\rho_t(C) = (c_t - \mu_j)^+, \quad t \in S^-_j, \quad j \in N$$

where

$$\mu_j = (m^d_j - m^u_j)^+, \quad j \in N.$$

Next, we compute the values $v(E)$ and $v(E \setminus \{t\})$ for $t \in C$. The feasibility of (F3.1) implies that $(E^+, \emptyset)$ is a path cover for N.

By Assumption (A.1), $(E^+ \setminus \{t\}, \emptyset)$ is also a path cover for N for each $t \in S^+$. Then $v(E) = v(E \setminus \{t\}) = d_{1n}$ and

$$\rho_t(E \setminus \{t\}) = 0, \quad t \in S^+ \cup L^-.$$

Then, inequality (3.4) can be explicitly written as

$$y(S^+) + \sum_{j \in N} \sum_{t \in E^+_j \setminus S^+_j} (y_t - \min\{c_t, \mu_j\} x_t) \le c(S^+) + y(E^- \setminus S^-) - \sum_{j \in N} \sum_{t \in S^-_j} (c_t - \mu_j)^+ (1 - x_t). \quad (3.17)$$

We refer to inequalities (3.17) as *path pack inequalities.*

Remark 3.4. Observe that for a path consisting of a single node $N = \{j\}$ with demand $d := d_j > 0$, the path pack inequalities (3.17), reduce to the flow pack inequalities (Atamtürk, 2001). Let (S^+, S^-) be a flow pack and $\mu := d - c(S^+) + c(S^-) > 0$. Moreover, the maximum flow that can be sent through S^+ for demand d and arcs in S^- is $c(S^+)$. Then, the value function $v(S^+) = c(S^+)$ and the resulting path pack inequality

$$y(S^+) + \sum_{t \in E^+ \setminus S^+} (y_t - \min\{c_t, \mu\}x_t) \leq c(S^+)$$
$$+ y(E^- \setminus S^-) - \sum_{t \in S^-} (c_t - \mu)^+(1 - x_t) \quad (3.18)$$

is equivalent to the flow pack inequality.

Proposition 3.5. Let (S^+, S^-) be a path pack for the node set N. The path pack inequality for N is at least as strong as the flow pack inequality for the single node relaxation obtained by merging the nodes in N.

Proof. The proof is similar to that of Proposition 3.4. Flow pack and path pack inequalities only differ in the coefficients of variables x_t for $t \in E^+ \setminus S^+$ and $t \in S^-$. Therefore, we compare the values μ_j, $j \in N$ of path pack inequalities (3.17) to the value μ of flow pack inequalities (3.18) and show that $\mu_j \leq \mu$ for all $j \in N$. For the single node relaxation, the values of the minimum cuts that pass above and below the merged node are $\bar{m}^u = c(S^+)$ and $\bar{m}^d = d_{1n} + c(S^-)$, respectively. The recursions in (4.3)–(4.6) imply that

$$m_j^u = \alpha_j^u + \beta_j^u - c(S_j^+) \leq c(S^+) = \bar{m}^u$$

and

$$m_j^d = \alpha_j^d + \beta_j^d - d_j - c(S_j^-) \leq d_{1n} + c(S^-) = \bar{m}^d.$$

The coefficient for flow pack inequality is $\mu = (\bar{m}^d - \bar{m}^u)^+$ and for path pack inequality $\mu_j = (m_j^d - m_j^u)^+$. Since (S^+, S^-) is a path pack, the minimum cut passes above all nodes in N and $m_j^u = c(S^+)$ for all $j \in N$. As a result, $m_j^u = \bar{m}^u$ for all $j \in N$ and $m_j^d \leq \bar{m}^d$. Then, observe that the values

$$\mu_j \leq \mu, \quad j \in N.$$

□

Example 1 (continued). Recall the lot-sizing instance with backlogging given in Figure 3.3. Let the node set $N = [1, 4]$ with $E^- = \emptyset$ and $S^+ = \{3\}$. Then, $m_1^u = 30$, $m_1^d = 40$, $m_2^u = 30$, $m_2^d = 40$, $m_3^u = 30$, $m_3^d = 30$, $m_4^u = 30$, $m_4^d = 30$, leading to $\mu_1 = 10$, $\mu_2 = 10$, $\mu_3 = 0$ and $\mu_4 = 0$. Moreover, the maximum flow values are $v(C) = 30$, $v(C \cup \{1\}) = 40$, $v(C \cup \{2\}) = 40$, $v(C \cup \{4\}) = 30$, $v(E) = 40$, and $v(E \setminus \{3\}) = 40$. Then the resulting path pack inequality (3.17) is

$$y_1 + y_2 + y_3 + y_4 \leq 30 + 10x_1 + 10x_2 \quad (3.19)$$

and it is facet-defining for conv($\mathcal{P}$) as will be shown in Section 3.3. Now, suppose that the nodes in $[1, 4]$ are merged into a single node with incoming arcs $\{1, 2, 3, 4\}$

and demand $d = 40$. For the same set S^+, we get $\mu = 40 - 30 = 10$. Then, the corresponding flow pack inequality (3.18) is

$$y_1 + y_2 + y_3 + y_4 \leq 30 + 10x_1 + 10x_2 + 10x_4,$$

which is weaker than the path pack inequality (3.19). □

Proposition 3.6. If $|E^+ \setminus S^+| \leq 1$ and $S^- = \emptyset$, then inequalities (3.14) and (3.17) are equivalent.

Proof. If $E^+ \setminus S^+ = \emptyset$ and $S^- = \emptyset$, then it is easy to see that the coefficients of inequality (3.17) are the same as (3.14). Moreover, if $|E^+ \setminus S^+| = 1$ (and wlog $E^+ \setminus S^+ = \{j\}$), then the resulting inequality (3.17) is

$$\begin{aligned} y(E^+) - y(E^-) &\leq v(C) + \rho_j(C)x_j \\ &= v(C) + \big(v(C \cup \{j\}) - v(C)\big)x_j \\ &= v(C \cup \{j\}) - \rho_j(C)(1 - x_j), \end{aligned}$$

which is equivalent to path cover inequality (3.14) with the objective set $(E^+, \emptyset)$. □

3.3 The strength of the path cover and pack inequalities

The capacities of the forward and the backward path arcs play an important role in finding the coefficients of the path cover and pack inequalities (3.14) and (3.17). Recall that K^+ and K^- are the coefficient sets in (3.5), (S^+, L^-) is the objective set for (F3.2) and $S^- = E^- \setminus (K^- \cup L^-)$.

Definition A node $j \in N$ is called *backward independent* for set (S^+, S^-) if

$$\alpha_j^u = \alpha_{j-1}^d + u_{j-1} + c(S_j^+),$$

or

$$\alpha_j^d = \alpha_{j-1}^u + b_{j-1} + d_j + c(S_j^-).$$

Definition A node $j \in N$ is called *forward independent* for set (S^+, S^-) if

$$\beta_j^u = \beta_{j+1}^d + b_j + c(S_j^+),$$

or

$$\beta_j^d = \beta_{j+1}^u + u_j + d_j + c(S_j^-).$$

Intuitively, backward independence of node $j \in N$ implies that the minimum cut either passes through the forward path arc $(j-1, j)$ or through the backward path arc $(j, j-1)$. Similarly, forward independence of node $j \in N$ implies that the minimum cut either passes through the forward path arc $(j, j+1)$ or through the backward path arc $(j+1, j)$. In Lemmas 3.1 and 3.2 below, we further explain how forward and backward independence affect the coefficients of path cover and pack inequalities. First, let $S^+_{jk} = \cup_{i=j}^k S^+_i$, $S^-_{jk} = \cup_{i=j}^k S^-_i$ and $L^-_{jk} = \cup_{i=j}^k L^-_i$ if $j \leq k$, and $\emptyset$ otherwise.

Lemma 3.1. If a node $j \in N$ is backward independent for set (S^+, S^-), then the values λ_j and μ_j do not depend on the sets S^+_{1j-1}, S^-_{1j-1} and the value d_{1j-1}.

Proof. If a node j is backward independent, then either $\alpha^u_j = \alpha^d_{j-1} + u_{j-1} + c(S^+_j)$ or $\alpha^d_j = \alpha^u_{j-1} + b_{j-1} + d_j + c(S^-_j)$. If $\alpha^u_j = \alpha^d_{j-1} + u_{j-1} + c(S^+_j)$, then the equality in (4.3) implies $\alpha^d_{j-1} + u_{j-1} \leq \alpha^u_{j-1}$. As a result, the equality in (4.4) gives $\alpha^d_j = \alpha^d_{j-1} + d_j + c(S^-_j)$. Following the definitions in (4.7)–(4.8), the difference

$$w_j := m^u_j - m^d_j$$

is $\beta^u_j - \beta^d_j + u_{j-1}$ which only depends on sets S^+_k and S^-_k for $k \in [j, n]$, the value d_{jn} and the capacity of the forward path arc $(j-1, j)$.

If $\alpha^d_j = \alpha^u_{j-1} + b_{j-1} + d_j + c(S^-_j)$, then the equality in (4.4) implies $\alpha^u_{j-1} + b_{j-1} \leq \alpha^d_{j-1}$. As a result, the equality in (4.3) gives $\alpha^u_j = \alpha^u_{j-1} + c(S^+_j)$. Then, the difference $w_j = \beta^u_j - \beta^d_j - b_{j-1}$ which only depends on sets S^+_k and S^-_k for $k \in [j, n]$, the value d_{jn} and the capacity of the backward path arc $(j, j-1)$.

Since the values λ_j and μ_j are defined as $(w_j)^+$ and $(-w_j)^+$ respectively, the result follows. □

Remark 3.5. Let $w_j := m^u_j - m^d_j$. If a node $j \in N$ is backward independent for a set (S^+, S^-), then we observe the following: (1) If $\alpha^u_j = \alpha^d_{j-1} + u_{j-1} + c(S^+_j)$, then

$$w_j = \beta^u_j - \beta^d_j + u_{j-1},$$

and (2) if $\alpha^d_j = \alpha^u_{j-1} + b_{j-1} + d_j + c(S^-_j)$, then

$$w_j = \beta^u_j - \beta^d_j - b_{j-1}.$$

Lemma 3.2. If a node $j \in N$ is forward independent for set (S^+, S^-), then the values λ_j and μ_j do not depend on the sets S^+_{j+1n}, S^-_{j+1n} and the value d_{j+1n}.

Proof. The forward independence implies either $\beta^u_j = \beta^d_{j+1} + b_j + c(S^+_j)$ and $\beta^d_j = \beta^d_{j+1} + d_j + c(S^-_j)$ or $\beta^u_j = \beta^u_{j+1} + c(S^+_j)$ and $\beta^d_j = \beta^u_{j+1} + u_j + d_j + c(S^-_j)$. Then, the difference $w_j = m^u_j - m^d_j$ is either $\alpha^u_j + \alpha^d_j + b_j$ or $\alpha^u_j + \alpha^d_j - u_j$ and in both cases, it is independent of the sets S^+_k, S^-_k for $k \in [1, j-1]$ and the value d_{1j-1}. □

Remark 3.6. Let $w_j := m_j^u - m_j^d$. If a node $j \in N$ is forward independent for a set (S^+, S^-), then we observe the following: (1) If $\beta_j^u = \beta_{j+1}^d + b_j + c(S_j^+)$, then

$$w_j = \alpha_j^u - \alpha_j^d + b_j,$$

and (2) if $\beta_j^d = \beta_{j+1}^u + u_j + d_j + c(S_j^-)$, then

$$w_j = \alpha_j^u - \alpha_j^d - u_j.$$

Corollary 3.7. If a node $j \in N$ is backward independent for set (S^+, S^-), then the values λ_k and μ_k for $k \in [j, n]$ are also independent of the sets S_{1j-1}^+, S_{1j-1}^- and the value d_{1j-1}. Similarly, if a node $j \in N$ is forward independent for set (S^+, S^-), then the values λ_k and μ_k for $k \in [1, j]$ are also independent of the sets S_{j+1n}^+, S_{j+1n}^- and the value d_{j+1n}.

Proof. The proof follows from recursions in (4.3)–(4.6). If a node j is backward independent, we write α_{j+1}^u and α_{j+1}^d in terms of α_{j-1}^u and α_{j-1}^d and observe that the difference $w_{j+1} = m_{j+1}^u - m_{j+1}^d$ does not depend on α_{j-1}^u nor α_{j-1}^d which implies independence of sets S_{1j-1}^+, S_{1j-1}^- and the value d_{1n}. We can repeat the same argument for w_j, $j \in [j+2, n]$ to show independence.

We show the same result for forward independence by writing β_{j-1}^u and β_{j-1}^d in terms of β_{j+1}^u and β_{j+1}^d, we observe that w_j does not depend on β_{j+1}^u nor β_{j+1}^d. Then, it is clear that w_{j-1} is also independent of the sets S_{j+1n}^+, S_{j+1n}^- and the value d_{j+1n}. We can repeat the same argument for w_j, $j \in [1, j-1]$ to show independence. □

Proving the necessary facet conditions frequently requires a partition of the node set N into two disjoint sets. Suppose, N is partitioned into $N_1 = [1, j-1]$ and $N_2 = [j, n]$ for some $j \in N$. Let E_{N1} and E_{N2} be the set of non-path arcs associated with node sets N_1 and N_2. We consider the forward and backward path arcs $(j-1, j)$ and $(j, j-1)$ to be in the set of non-path arcs E_{N1} and E_{N2} since the node $j-1 \in N_i$ and $j \notin N_i$ for $i = 1, 2$. In particular, $E_{N1}^+ := (j, j-1) \cup E_{1j-1}^+$, $E_{N1}^- := (j-1, j) \cup E_{1j-1}^-$ and $E_{N2}^+ := (j-1, j) \cup E_{jn}^+$, $E_{N2}^- := (j, j-1) \cup E_{jn}^-$, where $E_{k\ell}^+$ and $E_{k\ell}^-$ are defined as $\cup_{i=k}^{\ell} E_i^+$ and $\cup_{i=k}^{\ell} E_i^-$ if $k \leq \ell$ respectively, and as the empty set otherwise. Since the path arcs for N do not have associated fixed-charge variables, one can assume that there exists auxiliary binary variables $\tilde{x}_k = 1$ for $k \in \{(j-1, j), (j, j-1)\}$. Moreover, we partition the sets S^+, S^- and L^- into $S_{N1}^+ \supseteq S_{1j-1}^+$, $S_{N1}^- \supseteq S_{1j-1}^-$, $L_{N1}^- := L_{1j-1}^-$ and $S_{N2}^+ \supseteq S_{jn}^+$, $S_{N2}^- \supseteq S_{jn}^-$, $L_{N2}^- := L_{jn}^-$. Then, let v_1 and v_2 be the value functions defined in (F3.2) for the node sets N_1 and N_2 and the objective sets (S_{N1}^+, L_{N1}^-) and (S_{N2}^+, L_{N2}^-). Moreover, let α_j^u, α_j^d, β_j^u and β_j^d be defined for $j \in N$ in recursions (4.3)–(4.6) for the set (S^+, S^-) and recall that $S^- = E^- \setminus (K^- \cup L^-)$.

Lemma 3.3. Let (S^+, L^-) be the objective set for the node set for $N = [1, n]$. If

$\alpha_j^u = \alpha_{j-1}^d + u_{j-1} + c(S_j^+)$ or $\beta_{j-1}^u = \beta_j^d + b_{j-1} + c(S_{j-1}^+)$, then

$$v(S^+, L^-) = v_1(S_{N1}^+, L_{N1}^-) + v_2(S_{N2}^+, L_{N2}^-),$$

where $N_1 = [1, j-1]$, $N_2 = [j, n]$ and the arc sets are $S_{N1}^+ = (j, j-1) \cup S_{1j-1}^+$, $S_{N2}^+ = (j-1, j) \cup S_{jn}^+$, $S_{N1}^- = S_{1j-1}^-$, $S_{N2}^- = S_{jn}^-$.

Proof. See Appendix D.1. □

Lemma 3.4. Let (S^+, L^-) be the objective set for the node set for $N = [1, n]$. If $\alpha_j^d = \alpha_{j-1}^u + b_{j-1} + d_{j-1} + c(S_j^-)$ or $\beta_{j-1}^d = \beta_j^u + u_{j-1} + d_{j-1} + c(S_{j-1}^-)$, then

$$v(S^+, L^-) = v_1(S_{N1}^+, L_{N1}^-) + v_2(S_{N2}^+, L_{N2}^-),$$

where $N_1 = [1, j-1]$, $N_2 = [j, n]$ and the arc sets are $S_{N1}^+ = S_{1j-1}^+$, $S_{N2}^+ = S_{jn}^+$, $S_{N1}^- = (j-1, j) \cup S_{1j-1}^-$, $S_{N2}^- = (j, j-1) \cup S_{jn}^-$.

Proof. See Appendix D.2. □

Lemma 3.5. Let (S^+, L^-) be the objective set for the node set for $N = [1, n]$. If $\alpha_j^u = \alpha_{j-1}^d + u_{j-1} + c(S_j^+)$ and $\beta_{j-1}^d = \beta_j^u + u_{j-1} + d_{j-1} + c(S_{j-1}^-)$, then

$$v(S^+, L^-) = v_1(S_{N1}^+, L_{N1}^-) + v_2(S_{N2}^+, L_{N2}^-),$$

where $N_1 = [1, j-1]$, $N_2 = [j, n]$ and the arc sets are $S_{N1}^+ = S_{1j-1}^+$, $S_{N2}^+ = (j-1, j) \cup S_{jn}^+$, $S_{N1}^- = S_{1j-1}^-$, $S_{N2}^- = (j, j-1) \cup S_{jn}^-$.

Proof. See Appendix D.3. □

Lemma 3.6. Let (S^+, L^-) be the objective set for the node set for $N = [1, n]$. If $\alpha_j^d = \alpha_{j-1}^u + b_{j-1} + d_j + c(S_j^-)$ and $\beta_{j-1}^u = \beta_j^d + b_{j-1} + c(S_{j-1}^+)$, then

$$v(S^+, L^-) = v_1(S_{N1}^+, L_{N1}^-) + v_2(S_{N2}^+, L_{N2}^-),$$

where $N_1 = [1, j-1]$, $N_2 = [j, n]$ and the arc sets are $S_{N1}^+ = (j, j-1) \cup S_{1j-1}^+$, $S_{N2}^+ = S_{jn}^+$, $S_{N1}^- = (j-1, j) \cup S_{1j-1}^-$, $S_{N2}^- = S_{jn}^-$.

Proof. See Appendix D.4. □

In the remainder of this section, we give necessary and sufficient conditions for path cover and pack inequalities (3.14) and (3.17) to be facet-defining for the convex hull of $\mathcal{P}$.

Theorem 3.8. Let $N = [1, n]$, and $d_j \geq 0$ for all $j \in N$. If $L^- = \emptyset$ and the set (S^+, S^-) is a path cover for N, then the following conditions are necessary for path cover inequality (3.14) to be facet-defining for conv($\mathcal{P}$):

(i) $\rho_t(C \setminus \{t\}) < c_t$, for all $t \in C$,

(ii) $\max_{t\in S^+} \rho_t(C \setminus \{t\}) > 0$,

(iii) if a node $j \in [2, n]$ is forward independent for set (S^+, S^-), then node $j-1$ is not backward independent for set (S^+, S^-),

(iv) if a node $j \in [1, n-1]$ is backward independent for set (S^+, S^-), then node $j+1$ is not forward independent for set (S^+, S^-),

(v) if $\max_{t\in S_i^+}(c_t - \lambda_i)^+ = 0$ for $i = p, \dots, n$ for some $p \in [2, n]$, then the node $p-1$ is not forward independent for (S^+, S^-),

(vi) if $\max_{t\in S_i^+}(c_t - \lambda_i)^+ = 0$ for $i = 1, \dots, q$ for some $q \in [1, n-1]$, then the node $q+1$ is not backward independent for (S^+, S^-).

Proof. (i) If for some $t' \in S^+$, $\rho_{t'}(C \setminus \{t'\}) \geq c_{t'}$, then the path cover inequality with the objective set $S^+ \setminus \{t'\}$ summed with $y_{t'} \leq c_{t'}x_{t'}$ results in an inequality at least as strong. Rewriting the path cover inequality using the objective set S^+, we obtain

$$\sum_{t\in S^+\setminus\{t'\}} (y_t + \rho_t(S^+ \setminus \{t\})(1 - x_t)) + y_{t'} \leq v(S^+) - \rho_{t'}(S^+ \setminus \{t'\})(1 - x_{t'}) + y(E^- \setminus S^-)$$
$$= v(S^+ \setminus \{t'\}) + \rho_{t'}(S^+ \setminus \{t'\})x_{t'} + y(E^- \setminus S^-).$$

Now, consider summing the path cover inequality for the objective set $S^+ \setminus \{t'\}$

$$\sum_{t\in S^+\setminus\{t'\}} (y_t + \rho_t(S^+ \setminus \{t, t'\})(1 - x_t)) \leq v(S \setminus \{t'\}) + y(E^- \setminus S^-),$$

and $y_{t'} \leq c_{t'}x_{t'}$. The resulting inequality dominates inequality (3.3) because $\rho_t(S^+ \setminus \{t\}) \leq \rho_t(S^+ \setminus \{t, t'\})$, from the submodularity of the set function v. If the assumption of $L^- = \emptyset$ is dropped, this condition extends for arcs $t \in L^-$ as $\rho_t(C \setminus \{t\}) > -c_t$ with a similar proof.

(ii) If $L^- = \emptyset$ and $\max_{t\in S^+} \rho_t(C \setminus \{t\}) = 0$, then summing flow balance equalities (3.1b) for all nodes $j \in N$ gives an inequality at least as strong.

(iii) Suppose a node j is forward independent for (S^+, S^-) and the node $j-1$ is backward independent for (S^+, S^-) for some $j \in [2, n]$. Lemmas 3.3–3.6 show that the nodes N and the arcs $C = S^+ \cup L^-$ can be partitioned into $N_1 = [1, j-1]$, $N_2 = [j, n]$ and C_1, C_2 such that the sum of the minimum cut values for N_1, N_2 is equal to the minimum cut for N. From Remarks 3.5 and 3.6 and Corollary 3.7, it is easy to see that λ_i for $i \in N$ will not change by the partition procedures described in Lemmas 3.3–3.6. We examine the four cases for node $j-1$ to be forward independent and node j to be backward independent for the set (S^+, S^-).

(a) Suppose $\alpha^u_j = \alpha^d_{j-1} + u_{j-1} + c(S^+_j)$ and $\beta^u_{j-1} = \beta^d_j + b_{j-1} + c(S^+_{j-1})$. Consider the partition procedure described in Lemma 3.3, where $S^+_{N1} = (j, j-1) \cup S^+_{1j-1}$, $S^+_{N2} = (j-1, j) \cup S^+_{jn}$, $S^-_{N1} = S^-_{1j-1}$, $S^-_{N2} = S^-_{jn}$. Then, the path cover inequalities for nodes N_1 and N_2

$$r_{j-1} + \sum_{i=1}^{j-1} \sum_{t \in S^+_i} \big(y_t + (c_t - \lambda_i)^+(1 - x_t)\big) \le v_1(S^+_{N1}) + \sum_{i=1}^{j-1} y(E^-_i \setminus S^-_i) + i_{j-1}$$

and

$$i_{j-1} + \sum_{i=j}^{n} \sum_{t \in S^+_i} \big(y_t + (c_t - \lambda_i)^+(1 - x_t)\big) \le v_2(S^+_{N2}) + \sum_{i=j}^{n} y(E^-_i \setminus S^-_i) + r_{j-1}$$

summed gives

$$\sum_{i=1}^{n} \sum_{t \in S^+_i} \big(y_t + (c_t - \lambda_i)^+(1 - x_t)\big) \le v(S^+) + y(E^- \setminus S^-),$$

which is the path cover inequality for N with the objective set S^+.

(b) Suppose $\alpha^d_j = \alpha^u_{j-1} + b_{j-1} + d_{j-1} + c(S^-_j)$ and $\beta^d_{j-1} = \beta^u_j + u_{j-1} + d_{j-1} + c(S^-_{j-1})$. Consider the partition described in Lemma 3.4, where $S^+_{N1} = S^+_{1j-1}$, $S^+_{N2} = S^+_{jn}$, $S^-_{N1} = (j-1, j) \cup S^-_{1j-1}$, $S^-_{N2} = (j, j-1) \cup S^-_{jn}$. The path cover inequalities for nodes N_1 and N_2

$$\sum_{i=1}^{j-1} \sum_{t \in S^+_i} \big(y_t + (c_t - \lambda_i)^+(1 - x_t)\big) \le v_1(S^+_{N1}) + \sum_{i=1}^{j-1} y(E^-_i \setminus S^-_i)$$

and

$$\sum_{i=j}^{n} \sum_{t \in S^+_i} \big(y_t + (c_t - \lambda_i)^+(1 - x_t)\big) \le v_2(S^+_{N2}) + \sum_{i=j}^{n} y(E^-_i \setminus S^-_i).$$

summed gives the path cover inequality for nodes N and arcs C.

(c) Suppose $\alpha^u_j = \alpha^d_{j-1} + u_{j-1} + c(S^+_j)$ and $\beta^d_{j-1} = \beta^u_j + u_{j-1} + d_{j-1} + c(S^-_{j-1})$. Consider the partition described in Lemma 3.5, where $S^+_{N1} = S^+_{1j-1}$, $S^+_{N2} = (j-1, j) \cup S^+_{jn}$, $S^-_{N1} = S^-_{1j-1}$, $S^-_{N2} = (j, j-1) \cup S^-_{jn}$. The path cover inequalities for nodes N_1 and N_2

$$\sum_{i=1}^{j-1}\sum_{t\in S_i^+}\big(y_t+(c_t-\lambda_i)^+(1-x_t)\big)\le v_1(S_{N1}^+)+\sum_{i=1}^{j-1}y(E_i^-\setminus S_i^-)+i_{j-1}$$

and

$$i_{j-1}+\sum_{i=j}^{n}\sum_{t\in S_i^+}\big(y_t+(c_t-\lambda_i)^+(1-x_t)\big)\le v_2(S_{N2}^+)+\sum_{i=j}^{n}y(E_i^-\setminus S_i^-).$$

summed gives the path cover inequality for nodes N and arcs C.

(d) Suppose $\alpha_j^d = \alpha_{j-1}^u + b_{j-1} + d_j + c(S_j^-)$ and $\beta_{j-1}^u = \beta_j^d + b_{j-1} + c(S_{j-1}^+)$. Consider the partition described in Lemma 3.6, where $S_{N1}^+ = (j, j-1) \cup S_{1j-1}^+$, $S_{N2}^+ = S_{jn}^+$, $S_{N1}^- = (j-1, j) \cup S_{1j-1}^-$, $S_{N2}^- = S_{jn}^-$. The path cover inequalities for nodes N_1 and N_2

$$r_{j-1}+\sum_{i=1}^{j-1}\sum_{t\in S_i^+}\big(y_t+(c_t-\lambda_i)^+(1-x_t)\big)\le v_1(S_{N1}^+)+\sum_{i=1}^{j-1}y(E_i^-\setminus S_i^-)$$

and

$$\sum_{i=j}^{n}\sum_{t\in S_i^+}\big(y_t+(c_t-\lambda_i)^+(1-x_t)\big)\le v_2(S_{N2}^+)+\sum_{i=j}^{n}y(E_i^-\setminus S_i^-)+r_{j-1}.$$

summed gives the path cover inequality for nodes N and arcs C.

(iv) The same argument for condition (iii) above also proves the desired result here.

(v) Suppose $(c_t - \lambda_i)^+ = 0$ for all $t \in S_i^+$ and $i \in [p, n]$ and the node $p-1$ is forward independent for some $p \in [2, n]$. Then, we partition the node set $N = [1, n]$ into $N_1 = [1, p-1]$ and $N_2 = [p, n]$. We follow Lemma 3.3 if $\beta_{p-1}^u = \beta_p^d + b_{p-1} + c(S_{p-1}^+)$ and follow Lemma 3.4 if $\beta_{p-1}^d = \beta_p^u + u_{p-1} + d_{p-1} + c(S_{p-1}^-)$ to define S_{N1}^+, S_{N1}^-, S_{N2}^+ and S_{N2}^-. Remark 3.6 along with the partition procedure described in Lemma 3.3 or 3.4 implies that λ_i will remain unchanged for $i \in N_1$. The path cover inequality for nodes N and arcs C is

$$y(S^+)+\sum_{i=1}^{p-1}\sum_{t\in S_i^+}(c_t-\lambda_i)^+(1-x_t)\le v(S^+)+y(E^-\setminus S^-).$$

If $\beta_{p-1}^u = \beta_j^d + b_{p-1} + c(S_{p-1}^+)$, then the path cover inequality for nodes N_1 and arcs S^+_{N1}, S^-_{N1} described in Lemma 3.3 is

$$r_{p-1} + \sum_{i=1}^{p-1} \sum_{t \in S_i^+} \big(y_t + (c_t - \lambda_i)^+(1 - x_t)\big) \leq v(S^+_{N1}) + \sum_{i=1}^{p-1} y(E_i^- \setminus S_i^-) + i_{p-1}.$$

Moreover, let $\bar{m}^u_p$ and $\bar{m}^d_p$ be the values of minimum cut that goes above and below node p for the node set N_2 and arcs S^+_{N2}, S^-_{N2} and observe that

$$\bar{m}^u_p = \beta^u_p + u_{p-1} \text{ and } \bar{m}^d_p = \beta^d_p.$$

Then, comparing the difference $\bar{\lambda}_p := (\bar{m}^u_p - \bar{m}^d_p)^+ = (\beta^u_p - \beta^d_p + u_{p-1})^+$ to $\lambda_p = (m^u_p - m^d_p)^+ = (\beta^u_p - \beta^d_p + \alpha^u_p - \alpha^d_p + c(S^+_p) - d_p - c(S^-_p))^+$, we observe that $\bar{\lambda}_p \geq \lambda_p$ since $\alpha^u_p - \alpha^d_p + c(S^+_p) - d_p - c(S^-_p) \leq u_{p-1}$ from (4.3)–(4.4). Since $(c_t - \lambda_p)^+ = 0$, then $(c_t - \bar{\lambda}_p)^+ = 0$ as well. Using the same technique, it is easy to observe that $\bar{\lambda}_i \geq \lambda_i$ for $i \in [p+1, n]$ as well. As a result, the path cover inequality for N_2 with sets S^+_{N2}, S^-_{N2} is

$$i_{p-1} + \sum_{i=p}^{n} y(S_i^+) \leq v(S^+_{N2}) + \sum_{i=p}^{n} y(E_i^- \setminus S_i^-) + r_{p-1}.$$

The path cover inequalities for N_1, S^+_{N1}, S^-_{N1} and for N_2, S^+_{N2}, S^-_{N2} summed gives the path cover inequality for N, S^+, S^-.

Similarly, if $\beta^d_{p-1} = \beta^u_j + u_{p-1} + d_{p-1} + c(S^-_{p-1})$, the proof follows very similarly to the previous argument using Lemma 3.4. Letting $\bar{m}^u_p$ and $\bar{m}^d_p$ be the values of minimum cut that goes above and below node p for the node set N_2 and arcs S^+_{N2}, S^-_{N2}, we get

$$\bar{m}^u_p = \beta^u_p \text{ and } \bar{m}^d_p + b_{p-1} = \beta^d_p$$

under this case. Now, notice that $\alpha^u_p - \alpha^d_p + c(S^+_p) - d_p - c(S^-_p) \geq -b_{p-1}$ from (4.3)–(4.4), which leads to $\bar{\lambda}_p \geq \lambda_p$. Then the proof follows same as above.

(vi) The proof is similar to that of the necessary condition (v). We use Lemmas 3.5 and 3.6 and Remark 3.6 to partition the node set N and arcs S^+, S^- into node sets $N_1 = [1, q]$ and $N_2 = [q+1, n]$ for $q \in [2, n]$ and arcs S^+_{N1}, S^-_{N1} and S^+_{N2}, S^-_{N2}. Next, we check the values of minimum cut that goes above and below node q for the node set N_1 and arcs S^+_{N1}, S^-_{N1}. Then, observing $-u_q \leq \beta^u_q - \alpha^d_q + c(S^+_q) - d_q - c(S^-_q) \leq b_q$ from (4.5)–(4.6), it is easy to show that the coefficients x_t for $t \in S^+_{N1}$ are equal to zero in the path cover inequality for node set N_1. As a result, the path cover inequalities for N_1, S^+_{N1}, S^-_{N1} and for N_2, S^+_{N2}, S^-_{N2} summed gives the path cover inequality for N, S^+, S^-.

□

Remark 3.7. If the node set N consists of a single node, then the conditions (i) and (ii) of Theorem 3.8 reduce to the sufficient facet conditions of flow cover inequalities (Padberg et al., 1985, Theorem 4), (van Roy and Wolsey, 1986, Theorem 6). In this setting, conditions (iii)–(vi) are no longer relevant.

Theorem 3.9. Let $N = [1, n]$, $E^- = \emptyset$, $d_j > 0$ and $|E_j^+| = 1$, for all $j \in N$ and let the set S^+ be a path cover. The necessary conditions in Theorem 3.8 along with

(i) $(c_t - \lambda_j)^+ > 0$ for all $t \in S_j^+$, $j \in N$,

(ii) $(c_t - \lambda_j)^+ < c(E^+ \setminus S^+)$ for all $t \in S_j^+$, $j \in N$

are sufficient for path cover inequality (3.14) to be facet-defining for $\text{conv}(\mathcal{P})$.

Proof. Recall that $\dim\big(\text{conv}(\mathcal{P})\big) = 2|E| + n - 2$. In this proof, we provide $2|E| + n - 2$ affinely independent points that lie on the face F

$$F = \left\{ (\mathbf{x}, \mathbf{y}, \mathbf{i}, \mathbf{r}) \in \mathcal{P} \ : \ y(S^+) + \sum_{t \in S^+} (c_t - \lambda_j)^+ (1 - x_t) = d_{1n} \right\}.$$

First, we provide Algorithm 1 which outputs an initial feasible solution $(\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\mathbf{i}}, \bar{\mathbf{r}})$, where all the arcs in S^+ have non-zero flow. Let $\bar{d}_j$ be the effective demand on node j, that is, the sum of d_j and the minimal amount of flow that needs to be sent from the arcs in S_j^+ to ensure $v(S^+) = d_{1n}$. In Algorithm 1, we perform a backward pass and a forward pass on the nodes in N. This procedure is carried out to obtain the minimal amounts of flow on the forward and backward path arcs to satisfy the demands. For each node $j \in N$, these minimal outgoing flow values added to the demand d_j give the effective demand $\bar{d}_j$.

Algorithm 1 ensures that at most one of the path arcs $(j-1, j)$ and $(j, j-1)$ have non-zero flow for all $j \in [2, n]$. Moreover, note that sufficient condition (i) ensures that all the arcs in S^+ have nonzero flow. Moreover, for at least one node $i \in N$, it is guaranteed that $c(S_i^+) > \bar{d}_i$. Otherwise, $\rho_t(C) = c_t$ for all $t \in S^+$ which contradicts the necessary condition (i). Necessary conditions (iii) and (iv) ensure that $\bar{i}_j < u_j$ and $\bar{r}_j < b_j$ for all $j = 1, \dots, n-1$. Let

$$e := \arg\max_{i \in N} \{c(S_i^+) - \bar{d}_i\}$$

be the node with the largest excess capacity. Also let $\mathbf{1}_j$ be the unit vector with 1 at position j.

Next, we give $2|S^+|$ affinely independent points represented by $\bar{w}^t = (\bar{\mathbf{x}}^t, \bar{\mathbf{y}}^t, \bar{\mathbf{i}}^t, \bar{\mathbf{r}}^t)$ and $\tilde{w}^t = (\tilde{\mathbf{x}}^t, \tilde{\mathbf{y}}^t, \tilde{\mathbf{i}}^t, \tilde{\mathbf{r}}^t)$ for $t \in S^+$:

(i) Select $\bar{w}^e = (\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\mathbf{i}}, \bar{\mathbf{r}})$ given by Algorithm 1. Let $\varepsilon > 0$ be a sufficiently small value. We define $\bar{w}^t$ for $e \neq t \in S^+$ as $\bar{\mathbf{y}}^t = \bar{\mathbf{y}}^e + \varepsilon \mathbf{1}_e - \varepsilon \mathbf{1}_t$, $\bar{\mathbf{x}}^t = \bar{\mathbf{x}}^e$. If $t < e$, then $\bar{\mathbf{i}}^t = \bar{\mathbf{i}}^e$ and $\bar{r}_j^t = \bar{r}_j^e$ for $j < t$ and for $t \geq e$, $\bar{r}_j^t = \bar{r}_j^e + \varepsilon$ for $t \leq j < e$.

Algorithm 1

Initialization: Let $\bar{d}_j = d_j$ for $j \in N$
for $j = (n-1)$ **to** 1 **do**
  Let $\Delta = \min\left\{u_j, \left(\bar{d}_{j+1} - c(S^+_{j+1})\right)^+\right\}$,
  $\bar{d}_j = \bar{d}_j + \Delta$, $\bar{d}_{j+1} = \bar{d}_{j+1} - \Delta$,
  $\bar{i}_j = \Delta$.
end for
for $j = 2$ **to** n **do**
  Let $\Delta = \left(\bar{d}_{j-1} - c(S^+_{j-1})\right)^+$,
  $\bar{d}_j = \bar{d}_j + \Delta$, $\bar{d}_{j-1} - \Delta$
  $\bar{r}_{j-1} = \Delta - \min\{\Delta, \bar{i}_{j-1}\}$
  $\bar{i}_{j-1} = \bar{i}_{j-1} - \min\{\Delta, \bar{i}_{j-1}\}$
end for
$\bar{y}_j = \bar{d}_j$, for all $j \in S^+$.
$\bar{x}_j = 1$ if $j \in S^+$, 0 otherwise.
$\bar{y}_j = \bar{x}_j = 0$, for all $j \in E^-$.

(ii) In this class of affinely independent solutions, we close the arcs in S^+ one at a time and open all the arcs in $E^+ \setminus S^+$: $\tilde{\mathbf{x}}^t = \bar{\mathbf{x}} - \mathbf{1}_t + \sum_{j \in E^+ \setminus S^+} \mathbf{1}_j$. Next, we send an additional $\bar{y}_t - (c_t - \lambda_j)^+$ amount of flow from the arcs in $S^+ \setminus \{t\}$. This is a feasible operation because $v(C \setminus \{t\}) = d_{1n} - (c_t - \lambda_j)^+$. Let $(\mathbf{y}^*, \mathbf{i}^*, \mathbf{r}^*)$ be the optimal solution of (F3.2) corresponding to $v(S^+ \setminus \{t\})$. Then let, $\tilde{y}^t_j = y^*_j$ for $j \in S^+ \setminus \{t\}$. Since $v(C \setminus \{t\}) < d_{1n}$, additional flow must be sent through nodes in $E^+ \setminus S^+$ to satisfy flow balance equations (3.1b). This is also a feasible operation, because of assumption (A.1). Then, the forward and backward path flows $\tilde{\mathbf{i}}^t$ and $\tilde{\mathbf{r}}^t$ are calculated using the flow balance equations.

In the next set of solutions, we give $2|E^+ \setminus S^+| - 1$ affinely independent points represented by $\hat{w}^t = (\hat{\mathbf{x}}^t, \hat{\mathbf{y}}^t, \hat{\mathbf{i}}^t, \hat{\mathbf{r}}^t)$ and $\check{w}^t = (\check{\mathbf{x}}^t, \check{\mathbf{y}}^t, \check{\mathbf{i}}^t, \check{\mathbf{r}}^t)$ for $t \in E^+ \setminus S^+$.

(iii) Starting with solution $\bar{w}^e$, we open arcs in $E^+ \setminus S^+$, one by one. $\hat{\mathbf{y}}^t = \bar{\mathbf{y}}^e$, $\hat{\mathbf{x}}^t = \bar{\mathbf{x}}^e + \mathbf{1}_t$, $\hat{\mathbf{i}}^t = \bar{\mathbf{i}}^e$, $\hat{\mathbf{r}}^t = \bar{\mathbf{r}}^e$.

(iv) If $|E^+ \setminus S^+| \geq 2$, then we can send a sufficiently small $\varepsilon > 0$ amount of flow from arc $t \in E^+ \setminus S^+$ to $t \neq k \in E^+ \setminus S^+$. Let this set of affinely independent points be represented by $\check{w}^t$ for $t \in E^+ \setminus S^+$. While generating $\check{w}^t$, we start with the solution $\tilde{w}^e$, where the non-path arc in S^+_e is closed. The feasibility of this operation is guaranteed by the sufficiency conditions (ii) and necessary conditions (iii) and (iv).

(a) If $\tilde{y}^e_t = c_t$, then there exists at least one arc $t \neq m \in E^+ \setminus S^+$ such that $0 \leq \tilde{y}^e_m < c_m$ due to sufficiency assumption (ii), then for each $t \in E^+ \setminus S^+$

such that $\tilde{y}_t^e = c_t$, let $\check{\mathbf{y}}^t = \tilde{\mathbf{y}}^e - \varepsilon \mathbf{1}_t + \varepsilon \mathbf{1}_m$, $\check{\mathbf{x}}^t = \tilde{\mathbf{x}}^e$. If $t < m$, then $\check{\mathbf{i}}^t = \tilde{\mathbf{i}}^e$ and $\check{\mathbf{r}}^t = \tilde{\mathbf{r}}^e + \varepsilon \sum_{i=t}^{m-1} \mathbf{1}_i$. If $t > m$, then $\check{\mathbf{i}}^t = \tilde{\mathbf{i}}^e + \varepsilon \sum_{i=m}^{t-1} \mathbf{1}_i$ and $\check{\mathbf{r}}^t = \tilde{\mathbf{r}}^e$.

(b) If $\tilde{y}_t^e < c_t$ and there exists at least one arc $t \neq m \in E^+ \setminus S^+$ such that $\tilde{y}_m^e = 0$, then the same point described in (a) is feasible.

(c) If $\tilde{y}_t^e < c_t$ and there exists at least one arc $t \neq m \in E^+ \setminus S^+$ such that $\tilde{y}_m^e = c_m$, then, we send ε amount of flow from t to m, $\check{\mathbf{y}}^t = \tilde{\mathbf{y}}^e + \varepsilon \mathbf{1}_t - \varepsilon \mathbf{1}_m$, $\check{\mathbf{x}}^t = \tilde{\mathbf{x}}^e$. If $t < m$, then $\check{\mathbf{i}}^t = \tilde{\mathbf{i}}^e + \varepsilon \sum_{i=t}^{m-1} \mathbf{1}_i$ and $\check{\mathbf{r}}^t = \tilde{\mathbf{r}}^e$. If $t > m$, then $\check{\mathbf{i}}^t = \tilde{\mathbf{i}}^e$ and $\check{\mathbf{r}}^t = \tilde{\mathbf{r}}^e + \varepsilon \sum_{i=m}^{t-1} \mathbf{1}_i$.

Finally, we give $n-1$ points that perturb the flow on the forward path arcs $(j, j+1)$ for $j = 1, \ldots, n-1$ represented by $\breve{w}^j = (\check{\mathbf{x}}^j, \check{\mathbf{y}}^j, \check{\mathbf{i}}^j, \check{\mathbf{r}}^j)$. Let $k = \min\{i \in N : S_i^+ \neq \emptyset\}$ and $\ell = \max\{i \in N : \; S_i^+ \neq \emptyset\}$. The solution given by Algorithm 1 guarantees $\bar{i}_j < u_j$ and $\bar{r}_j < b_j$ for $j = 1, \ldots, n-1$ due to necessary conditions (iii) and (iv).

(v) For $j = 1, \ldots, n-1$, we send an additional ε amount of flow from the forward path arc $(j, j+1)$ and the backward path arc $(j+1, j)$. Formally, the solution $\breve{w}^j$ can be obtained by: $\check{\mathbf{y}}^j = \bar{\mathbf{y}}^e$, $\check{\mathbf{x}}^j = \bar{\mathbf{x}}^e$, $\check{\mathbf{i}}^j = \bar{\mathbf{i}}^e + \varepsilon \mathbf{1}_j$ and $\check{\mathbf{r}}^j = \bar{\mathbf{r}}^e + \varepsilon \mathbf{1}_j$.

□

Next, we identify conditions under which path pack inequality (3.17) is facet-defining for conv($\mathcal{P}$).

Theorem 3.10. Let $N = [1, n]$, $d_j \geq 0$ for all $j \in N$, let the set (S^+, S^-) be a path pack and $L^- = \emptyset$. The following conditions are necessary for path pack inequality (3.17) to be facet-defining for conv($\mathcal{P}$):

(i) $\rho_j(S^+) < c_j$, for all $j \in E^+ \setminus S^+$,

(ii) $\max_{t \in S^-} \rho_t(C) > 0$,

(iii) if a node $j \in [2, n]$ is forward independent for set (S^+, S^-), then node $j-1$ is not backward independent for set (S^+, S^-),

(iv) if a node $j \in [1, n-1]$ is backward independent for set (S^+, S^-), then node $j+1$ is not forward independent for set (S^+, S^-),

(v) if $\max_{t \in E_i^+ \setminus S_i^+} \rho_t(C) = 0$ and $\max_{t \in S_i^-} \rho_t(C) = 0$ for $i = p, \ldots, n$ for some $p \in [2, n]$, then the node $p-1$ is not forward independent for (S^+, S^-),

(vi) if $\max_{t \in E_i^+ \setminus S_i^+} \rho_t(C) = 0$ and $\max_{t \in S_i^-} \rho_t(C) = 0$ for $i = 1, \ldots, q$ for some $q \in [1, n-1]$, then the node $q+1$ is not backward independent for (S^+, S^-).

Proof. (i) Suppose that for some $k \in E^+ \setminus S^+$, $\rho_k(S^+) = c_k$. Then, recall the implicit form of path pack inequality (3.17) is

$$y(E^+ \setminus \{k\}) + y_k + \sum_{t \in S^-} \rho_t(S^+)(1 - x_t) \leq v(S^+) + \sum_{k \neq t \in E^+ \setminus S^+} \rho_t(S^+)x_t + c_k x_k + y(E^- \setminus S^-).$$

Now, if we select $a_k = 0$ in (F3.2), then the coefficients of x_k and y_k become zero and summing the path cover inequality

$$y(E^+ \setminus \{k\}) + \sum_{t \in S^-} \rho_t(S^+)(1 - x_t) \leq v(S^+) + \sum_{k \neq t \in E^+ \setminus S^+} \rho_t(S^+)x_t + y(E^- \setminus S^-).$$

with $y_k \leq c_k x_k$ gives the first path cover inequality.

(ii) Suppose that $\rho_j(S^+) = 0$ for all $j \in S^-$. Then the path pack inequality is

$$y(E^+) \leq v(S^+) + \sum_{t \in E^+ \setminus S^+} \rho_t(S^+)x_t + y\left(E^- \setminus (L^- \cup S^-)\right),$$

where $L^- = \emptyset$. If an arc j is dropped from S^- and added to L^-, then $v(S^+) = v(S^+ \cup \{j\})$ since $\rho_j(S^+) = 0$ for $j \in S^-$. Consequently, the path pack inequality with $S^- = S^- \setminus \{j\}$ and $L^- = \{j\}$

$$y(E^+) + \sum_{t \in S^-} \rho_t(S^+ \cup \{j\})(1 - x_t) \leq v(S^+) + \sum_{t \in E^+ \setminus S^+} \rho_t(S^+ \cup \{j\})x_t + y\left(E^- \setminus (L^- \cup S^-)\right).$$

But since $0 \leq \rho_t(S^+ \cup \{j\}) \leq \rho_t(S^+)$ from submodularity of v and $\rho_t(S^+) = 0$ for all $t \in S^-$, we observe that the path pack inequality above reduces to

$$y(E^+) \leq v(S^+) + \sum_{t \in E^+ \setminus S^+} \rho_t(S^+ \cup \{j\})x_t + y\left(E^- \setminus (L^- \cup S^-)\right)$$

and it is at least as strong as the first pack inequality for S^+, S^- and $L^- = \emptyset$.

(iii)–(iv) We repeat the same argument of the proof of condition (iii) of Theorem 3.8. Suppose a node j is forward independent for (S^+, S^-) and the node $j-1$ is backward independent for (S^+, S^-) for some $j \in [2, n]$. Lemmas 3.3–3.6 show that the nodes N and the arcs $C = S^+ \cup L^-$ can be partitioned into $N_1 = [1, j-1]$, $N_2 = [j, n]$ and C_1, C_2 such that the sum of the minimum cut values for N_1, N_2 is equal to the minimum cut for N. From Remarks 3.5 and 3.6 and Corollary 3.7, it is easy to see that μ_i for $i \in N$ will not change by the partition procedures described in Lemmas 3.3–3.6. We examine the four cases for node $j-1$ to be forward independent and node j to be backward independent for the set (S^+, S^-).

For ease of notation, let

$$Q^+_{jk} := \sum_{i=j}^{k} \sum_{t \in E^+_i \setminus S^+_i} (y_t - \min\{\mu_i, c_t\} x_t)$$

and

$$Q^-_{jk} := \sum_{i=j}^{k} \sum_{t \in S^-_i} (c_t - \mu_i)^+ (1 - x_t)$$

for $j \leq k$ and $j \in N$, $k \in N$ (and zero if $j > k$), where the values μ_i are the coefficients that appear in the path pack inequality (3.17). As a result, the path pack inequality can be written as

$$y(S^+) + Q^+_{1n} \leq v(C) + Q^-_{1n} + y(E^- \setminus S^-). \tag{3.20}$$

(a) Suppose $\alpha^u_j = \alpha^d_{j-1} + u_{j-1} + c(S^+_j)$ and $\beta^u_{j-1} = \beta^d_j + b_{j-1} + c(S^+_{j-1})$. Consider the partition procedure described in Lemma 3.3, where $S^+_{N1} = (j, j-1) \cup S^+_{1j-1}$, $S^+_{N2} = (j-1, j) \cup S^+_{jn}$, $S^-_{N1} = S^-_{1j-1}$, $S^-_{N2} = S^-_{jn}$. Then, the path pack inequalities for nodes N_1 is

$$r_{j-1} + y(S^+_{1j-1}) + Q^+_{1j-1} \leq v_1(C_1) + Q^-_{1j-1} + y(E^-_{1j-1} \setminus S^-_{1j-1}) f + i_{j-1}. \tag{3.21}$$

Similarly, the path pack inequality for N_2 is

$$i_{j-1} + y(S^+_{jn}) + Q^+_{jn} \leq v_2(C_2) + Q^-_{jn} + y(E^-_{jn} \setminus S^-_{jn}) + r_{j-1}. \tag{3.22}$$

Inequalities (3.21)–(3.22) summed gives the path pack inequality (3.20).

(b) Suppose $\alpha^d_j = \alpha^u_{j-1} + b_{j-1} + d_j + c(S^-_j)$ and $\beta^d_{j-1} = \beta^u_j + u_{j-1} + d_{j-1} + c(S^-_{j-1})$. Consider the partition described in Lemma 3.4, where $S^+_{N1} = S^+_{1j-1}$, $S^+_{N2} = S^+_{jn}$, $S^-_{N1} = (j-1, j) \cup S^-_{1j-1}$, $S^-_{N2} = (j, j-1) \cup S^-_{jn}$. The submodular inequality (3.4) for nodes N_1 where the objective coefficients of (F3.2) are selected as $a_t = 1$ for $t \in E^+_{1j-1}$, $a_t = 0$ for $t = (j, j-1)$, $a_t = -1$ for $t \in E^-_{N1} \setminus S^-_{N1}$ and $a_t = 0$ for $t \in S^-_{N1}$ is

$$y(S^+_{1j-1}) + \sum_{t \in S^+_{N1}} k_t (1 - x_t) + Q^+_{1j-1} \leq v_1(C_1) - Q^-_{1j-1} + y(E^-_{1j-1} \setminus S^-_{1j-1}), \tag{3.23}$$

where k_t for $t \in S^+_{N1}$ are some nonnegative coefficients. Similarly, the submodular inequality (3.4) for nodes N_2, where the objective coefficients of (F3.2) are selected as $a_t = 1$ for $t \in E^+_{jn}$, $a_t = 0$ for $t = (j-1, j)$, $a_t = -1$

for $t \in E^-_{N2} \setminus S^-_{N2}$ and $a_t = 0$ for $t \in S^-_{N2}$ is

$$y(S^+_{jn}) + \sum_{t \in S^+_{N2}} k_t(1 - x_t) + Q^+_{jn} \leq v_2(C_2) - Q^-_{jn} + y(E^-_{jn} \setminus S^-_{jn}), \quad (3.24)$$

where k_t for $t \in S^+_{N2}$ are some nonnegative coefficients. The sum of inequalities (3.23)–(3.24) is at least as strong as the path pack inequality (3.20).

(c) Suppose $\alpha^u_j = \alpha^d_{j-1} + u_{j-1} + c(S^+_j)$ and $\beta^d_{j-1} = \beta^u_j + u_{j-1} + d_{j-1} + c(S^-_{j-1})$. Consider the partition described in Lemma 3.5, where $S^+_{N1} = S^+_{1j-1}$, $S^+_{N2} = (j-1, j) \cup S^+_{jn}$, $S^-_{N1} = S^-_{1j-1}$, $S^-_{N2} = (j, j-1) \cup S^-_{jn}$. The submodular inequality (3.4) for nodes N_1 where the objective coefficients of (F3.2) are selected as $a_t = 1$ for $t \in E^+_{1j-1}$, $a_t = 0$ for $t = (j, j-1)$, $a_t = -1$ for $t \in E^-_{N1} \setminus S^-_{N1}$ and $a_t = 0$ for $t \in S^-_{N1}$ is

$$y(S^+_{1j-1}) + \sum_{t \in S^+_{N1}} k_t(1 - x_t) + Q^+_{1j-1} \leq v_1(C_1) - Q^-_{1j-1} + y(E^-_{1j-1} \setminus S^-_{1j-1}) + i_{j-1}, \quad (3.25)$$

where k_t for $t \in S^+_{N1}$ are some nonnegative coefficients. The path pack inequality for N_2 is

$$i_{j-1} + y(S^+_{jn}) + Q^+_{jn} \leq v_2(C_2) + Q^-_{jn} + y(E^-_{jn} \setminus S^-_{jn}). \quad (3.26)$$

The sum of inequalities (3.25)–(3.26) is at least as strong as inequality (3.20).

(d) Suppose $\alpha^d_j = \alpha^u_{j-1} + b_{j-1} + d_j + c(S^-_j)$ and $\beta^u_{j-1} = \beta^d_j + b_{j-1} + c(S^+_{j-1})$. Consider the partition described in Lemma 3.6, where $S^+_{N1} = (j, j-1) \cup S^+_{1j-1}$, $S^+_{N2} = S^+_{jn}$, $S^-_{N1} = (j-1, j) \cup S^-_{1j-1}$, $S^-_{N2} = S^-_{jn}$. The path pack inequalities for nodes N_1 is

$$r_{j-1} + y(S^+_{1j-1}) + Q^+_{1j-1} \leq v_1(C_1) + Q^-_{1j-1} + y(E^-_{1j-1} \setminus S^-_{1j-1}). \quad (3.27)$$

The submodular inequality (3.4) for nodes N_2 where the objective coefficients of (F3.2) are selected as $a_t = 1$ for $t \in E^+_{jn}$, $a_t = 0$ for $t = (j-1, j)$, $a_t = -1$ for $t \in E^-_{N2} \setminus S^-_{N2}$ and $a_t = 0$ for $t \in S^-_{N2}$ is

$$y(S^+_{jn}) + \sum_{t \in S^+_{N2}} k_t(1 - x_t) + Q^+_{jn} \leq v_2(C_2) - Q^-_{jn} + y(E^-_{jn} \setminus S^-_{jn}) + r_{j-1}, \quad (3.28)$$

where k_t for $t \in S^+_{N2}$ are some nonnegative coefficients. The sum of inequalities (3.27)–(3.28) is at least as strong as the path pack inequality (3.20).

(v) Suppose $(c_t - \mu_i)^+ = 0$ for all $t \in S_i^-$ and $i \in [p, n]$ and the node $p-1$ is forward independent. Then, we partition the node set $N = [1, n]$ into $N_1 = [1, p-1]$ and $N_2 = [p, n]$. We follow Lemma 3.3 if $\beta_{p-1}^u = \beta_p^d + b_{p-1} + c(S_{p-1}^+)$ and follow Lemma 3.4 if $\beta_{p-1}^d = \beta_p^u + u_{p-1} + d_{p-1} + c(S_{p-1}^-)$ to define S_{N1}^+, S_{N1}^-, S_{N2}^+ and S_{N2}^-. Remark 3.6 along with the partition procedure described in Lemma 3.3 or 3.4 implies that μ_i will remain unchanged for $i \in N_1$.

If $\beta_{p-1}^u = \beta_j^d + b_{p-1} + c(S_{p-1}^+)$, then the coefficients μ_i of the path pack inequality for nodes N_1 and arcs S_{N1}^+, S_{N1}^- described in Lemma 3.3 is the same as the coefficients of the path pack inequality for nodes N and arcs S^+, S^-. Moreover, let $\bar{m}_p^u$ and $\bar{m}_p^d$ be the values of minimum cut that goes above and below node p for the node set N_2 and arcs S_{N2}^+, S_{N2}^- and observe that

$$\bar{m}_p^u = \beta_p^u + u_{p-1} \text{ and } \bar{m}_p^d = \beta_p^d.$$

Then, comparing the difference $\bar{\mu}_p := (\bar{m}_p^d - \bar{m}_p^u)^+ = (\beta_p^d - \beta_p^u - u_{p-1})^+$ to $\mu_p = (m_p^d - m_p^u)^+ = (\beta_p^d - \beta_p^u + \alpha_p^d - \alpha_p^u - c(S_p^+) + d_p + c(S_p^-))^+$, we observe that $\bar{\mu}_p \geq \mu_p$ since $\alpha_p^d - \alpha_p^u - c(S_p^+) + d_p + c(S_p^-) \geq -u_{p-1}$ from (4.3)–(4.4). Since $(c_t - \mu_p)^+ = 0$, then $(c_t - \bar{\mu}_p)^+ = 0$ as well. Using the same technique, it is easy to observe that $\bar{\mu}_i \geq \mu_i$ for $i \in [p+1, n]$ as well. As a result, the path pack inequality for N_2 with sets S_{N2}^+, S_{N2}^-, summed with the path pack inequality for nodes N_1 and arcs S_{N1}^+, S_{N1}^- give the path pack inequality for nodes N and arc S^+, S^-.

Similarly, if $\beta_{p-1}^d = \beta_j^u + u_{p-1} + d_{p-1} + c(S_{p-1}^-)$, the proof follows very similarly to the previous argument using Lemma 3.4. Letting $\bar{m}_p^u$ and $\bar{m}_p^d$ be the values of minimum cut that goes above and below node p for the node set N_2 and arcs S_{N2}^+, S_{N2}^-, we get

$$\bar{m}_p^u = \beta_p^u \text{ and } \bar{m}_p^d + b_{p-1} = \beta_p^d$$

under this case. Now, notice that $\alpha_p^d - \alpha_p^u - c(S_p^+) + d_p + c(S_p^-) \leq b_{p-1}$ from (4.3)–(4.4), which leads to $\bar{\mu}_p \geq \mu_p$. Then the proof follows same as above.

(vi) The proof is similar to that of the necessary condition (v) above. We use Lemmas 3.5–3.6 and Remark 3.6 to partition the node set N and arcs S^+, S^- into node sets $N_1 = [1, q]$ and $N_2 = [q+1, n]$ and arcs S_{N1}^+, S_{N1}^- and S_{N2}^+, S_{N2}^-. Next, we check the values of minimum cut that goes above and below node q for the node set N_1 and arcs S_{N1}^+, S_{N1}^-. Then, observing $-b_q \leq \beta_q^d - \alpha_q^u - c(S_q^+) + d_q + c(S_q^-) \leq u_q$ from (4.5)–(4.6), it is easy to see that the coefficients of x_t for $t \in S_{N1}^-$ and $t \in E_{N1}^+ \setminus S_{N1}^+$ are equal to zero in the path pack inequality for node set N_1. As a result, the path pack inequalities for N_1, S_{N1}^+, S_{N1}^- and for N_2, S_{N2}^+, S_{N2}^- summed gives the path pack inequality for N, S^+, S^-.

□

Remark 3.8. If the node set N consists of a single node, then the conditions (i) and (ii) of Theorem 3.10 reduce to the necessary and sufficient facet conditions of flow pack inequalities (Atamtürk, 2001, Proposition 1). In this setting, conditions (iii)–(vi) are no longer relevant.

Theorem 3.11. Let $N = [1, n]$, $E^- = \emptyset$, $d_j > 0$ and $|E_j^+| = 1$, for all $j \in N$ and let the objective set S^+ be a path pack for N. The necessary conditions in Theorem 3.10 along with

(i) for each $j \in E^+ \setminus S^+$, either $S^+ \cup \{j\}$ is a path cover for N or $\rho_j(S^+) = 0$,

(ii) for each $t \in S^+$, there exists $j_t \in E^+ \setminus S^+$ such that $S^+ \setminus \{t\} \cup \{j_t\}$ is a path cover for N,

(iii) for each $j \in [1, n-1]$, there exists $k_j \in E^+ \setminus S^+$ such that the set $S^+ \cup \{k_j\}$ is a path cover and neither node j is backward independent nor node $j+1$ is forward independent for the set $S^+ \cup \{k_j\}$

are sufficient for path pack inequality (3.17) to be facet-defining for conv($\mathcal{P}$).

Proof. We provide $2|E| + n - 2$ affinely independent points that lie on the face:

$$F = \left\{ (\mathbf{x}, \mathbf{y}, \mathbf{i}, \mathbf{r}) \in \mathcal{P} \ : \ y(S^+) + \sum_{t \in E^+ \setminus S^+} (y_t - \min\{\mu_j, c_t\} x_t) = c(S^+) \right\}.$$

Let $(\mathbf{y}^*, \mathbf{i}^*, \mathbf{r}^*) \in \mathcal{Q}$ be an optimal solution to (F3.2). Since S^+ is a path pack and $E^- = \emptyset$, $v(S^+) = c(S^+)$. Then, notice that $y_t^* = c_t$ for all $t \in S^+$. Moreover, let e be the arc with largest capacity in S^+, $\varepsilon > 0$ be a sufficiently small value and $\mathbf{1}_j$ be the unit vector with 1 at position j. First, we give $2|E^+ \setminus S^+|$ affinely independent points represented by $\bar{z}^t = (\bar{\mathbf{x}}^t, \bar{\mathbf{y}}^t, \bar{\mathbf{i}}^t, \bar{\mathbf{r}}^t)$ and $\tilde{z}^t = (\tilde{\mathbf{x}}^t, \tilde{\mathbf{y}}^t, \tilde{\mathbf{i}}^t, \tilde{\mathbf{r}}^t)$ for $t \in E^+ \setminus S^+$.

(i) Let $t \in E^+ \setminus S^+$, where $S^+ \cup \{t\}$ is a path cover for N. The solution $\bar{z}^t$ has arcs in $S^+ \cup \{t\}$ open, $\bar{x}_j^t = 1$ for $j \in S^+ \cup \{t\}$, 0 otherwise, $\bar{y}_j^t = y_j^*$ for $j \in S^+$ and $\bar{y}_t^t = \rho_t(S^+)$, 0 otherwise. The forward and backward path arc flow values $\bar{i}_j^t$ and $\bar{r}_j^t$ can then be calculated using flow balance equalities (3.1b) where at most one of them can be nonzero for each $j \in N$. Sufficiency condition (i) guarantees the feasibility of $\bar{z}^t$.

(ii) Let $t \in E^+ \setminus S^+$, where $\rho_t(S^+) = 0$ and let $t \neq \ell \in E^+ \setminus S^+$, where $S^+ \cup \{\ell\}$ is a path cover for N. The solution $\bar{z}^t$ has arcs in $S^+ \cup \{t, \ell\}$ open, $\bar{x}_j^t = 1$ for $j \in S^+ \cup \{t, \ell\}$, and 0 otherwise, $\bar{y}_j^t = y_j^*$ for $j \in S^+$, $\bar{y}_t^t = 0$, $\bar{y}_\ell^t = \rho_\ell(S^+)$, and 0 otherwise. The forward and backward path arc flow values $\bar{i}_j^t$ and $\bar{r}_j^t$ can then be calculated using flow balance equalities (3.1b) where at most one of them can be nonzero for each $j \in N$. Sufficiency condition (i) guarantees the feasibility of $\bar{z}^t$.

(iii) The necessary condition (i) ensures that $\rho_t(S^+) < c_t$, therefore $\bar{y}_t^t < c_t$. In solution $\tilde{z}^t$, starting with $\bar{z}^t$, we send a flow of ε from arc $t \in E^+ \setminus S^+$ to $e \in S^+$. Let $\tilde{\mathbf{y}}^t = \bar{\mathbf{y}}^t + \varepsilon \mathbf{1}_t - \varepsilon \mathbf{1}_e$ and $\tilde{\mathbf{x}}^t = \bar{\mathbf{x}}^t$. If $e < t$, then $\tilde{\mathbf{r}}^t = \bar{\mathbf{r}}^t + \varepsilon \sum_{i=e}^{t-1} \mathbf{1}_i$, $\tilde{\mathbf{i}}^t = \bar{\mathbf{i}}^t$ and if $e > t$, then $\tilde{\mathbf{i}}^t = \bar{\mathbf{i}}^t + \varepsilon \sum_{i=t}^{e-1} \mathbf{1}_i$, $\tilde{\mathbf{r}}^t = \bar{\mathbf{r}}^t$.

Next, we give $2|S^+| - 1$ affinely independent feasible points $\hat{z}^t$ and $\check{z}^t$ corresponding to $t \in S^+$ that are on the face F. Let k be the arc in $E^+ \setminus S^+$ with largest capacity.

(iv) In the feasible solutions $\hat{z}^t$ for $e \neq t \in S^+$, we open arcs in $S^+ \cup \{k\}$ and send an ε flow from arc k to arc t. Let $\hat{\mathbf{y}}^t = \bar{\mathbf{y}}^k + \varepsilon \mathbf{1}_k - \varepsilon \mathbf{1}_t$ and $\hat{\mathbf{x}}^t = \bar{\mathbf{x}}^k$. If $t < k$, then $\hat{\mathbf{r}}^t = \bar{\mathbf{r}}^k + \varepsilon \sum_{i=t}^{k-1} \mathbf{1}_i$, $\hat{\mathbf{i}}^t = \bar{\mathbf{i}}^k$ and if $t > k$, then $\hat{\mathbf{i}}^t = \bar{\mathbf{i}}^k + \varepsilon \sum_{i=k}^{t-1} \mathbf{1}_i$, $\hat{\mathbf{r}}^t = \bar{\mathbf{r}}^k$.

(v) In the solutions $\check{z}^t$ for $t \in S^+$, we close arc t and open arc $j_t \in E^+ \setminus S^+$ that is introduced in the sufficient condition (ii). Then, $\check{x}_j^t = 1$ if $j \in S^+ \setminus \{t\}$ and if $j = j_t$ and $\check{x}_j^t = 0$ otherwise. From sufficient condition (ii), there exists $\check{y}_j^t$ values that satisfy the flow balance equalities (3.1b). Moreover, these $\check{y}_j^t$ values satisfy inequality (3.17) at equality since both $S^+ \cup \{j_t\}$ and $S^+ \setminus \{t\} \cup \{j_t\}$ are path covers for N. Then, the forward and backward path arc flows are found using flow balance equalities where at most one of $\check{i}_j^t$ and $\check{r}_j^t$ are nonzero for each $j \in N$.

Finally, we give $n - 1$ points $\breve{z}^j$ corresponding to forward and backward path arcs connecting nodes j and $j + 1$.

(vi) In the solution set $\breve{z}^j$ for $j = 1, \ldots, n-1$, starting with solution $\bar{z}^{k_j}$, where k_j is introduced in the sufficient condition (iii), we send a flow of ε from both forward path arc $(j-1, j)$ and backward path arc $(j, j-1)$. Since the sufficiency condition (iii) ensures that $\bar{r}_j^{k_j} < b_j$ and $\bar{i}_j^{k_j} < u_j$, the operation is feasible. Let $\breve{\mathbf{y}}^j = \bar{\mathbf{y}}^{k_j}$, $\breve{\mathbf{x}}^j = \bar{\mathbf{x}}^{k_j}$, $\breve{\mathbf{i}}^j = \bar{\mathbf{i}}^{k_j} + \varepsilon \mathbf{1}_j$ and $\breve{\mathbf{r}}^j = \bar{\mathbf{r}}^{k_j} + \varepsilon \mathbf{1}_j$.

□

3.4 Computational study

We test the effectiveness of path cover and path pack inequalities (3.14) and (3.17) by embedding them in a branch-and-cut framework. The experiments are ran on a Linux workstation with 2.93 GHz Intel®Core™ i7 CPU and 8 GB of RAM with 1 hour time limit and 1 GB memory limit. The branch-and-cut algorithm is implemented in C++ using IBM's Concert Technology of CPLEX (version 12.5). The experiments are ran with one hour limit on elapsed time and 1 GB limit on memory usage. The number of threads is set to one and the dynamic search is disabled. We also turn off heuristics and preprocessing as the purpose is to see the impact of the inequalities by themselves.

Instance generation

We use a capacitated lot-sizing model with backlogging, where constraints (3.1b) reduce to:

$$i_{j-1} - r_{j-1} + y_j - i_j + r_j = d_j, \quad j \in N.$$

Let n be the total number of time periods and f be the ratio of the fixed cost to the variable cost associated with a non-path arc. The parameter c controls how large the non-path arc capacities are with respect to average demand. All parameters are generated from a discrete uniform distribution. The demand for each node is drawn from the range $[0, 30]$ and non-path arc capacities are drawn from the range $[0.75 \times c \times \bar{d}, 1.25 \times c \times \bar{d}]$, where $\bar{d}$ is the average demand over all time periods. Forward and backward path arc capacities are drawn from $[1.0 \times \bar{d}, 2.0 \times \bar{d}]$ and $[0.3 \times \bar{d}, 0.8 \times \bar{d}]$, respectively. The variable costs p_t, h_t and g_t are drawn from the ranges $[1, 10]$, $[1, 10]$ and $[1, 20]$ respectively. Finally, fixed costs f_t are set equal to $f \times p_t$. Using these parameters, we generate five random instances for each combination of $n \in \{50, 100, 150\}$, $f \in \{100, 200, 500, 1000\}$ and $c \in \{2, 5, 10\}$.

Finding violated inequalities

Given a feasible solution $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{i}^*, \mathbf{r}^*)$ to a linear programming (LP) relaxation of (F3.1), the separation problem aims to find sets S^+ and L^- that maximize the difference

$$y^*(S^+) - y^*(E^- \setminus L^-) + \sum_{t \in S^+} (c_t - \lambda_j)^+ (1 - x_t^*) - \sum_{t \in L^-} (\min\{\lambda_j, c_t\}) x_t^* - d_{1n} - c(S^-)$$

for path cover inequality (3.14) and sets S^+ and S^- that maximize

$$y^*(S^+) - y^*(E^- \setminus S^-) - \sum_{t \in E^+ \setminus S^+} \min\{c_t, \mu_j\} x_t^* + \sum_{t \in S^-} (c_t - \mu_j)^+ (1 - x_t) - c(S^+)$$

for path pack inequality (3.17). We use the knapsack relaxation based heuristic separation strategy described in Wolsey and Nemhauser (1999, pg. 500) for flow cover inequalities to choose the objective set S^+ with a knapsack capacity d_{1n}. Using S^+, we obtain the values λ_j and μ_j for each $j \in N$ and let $S^- = \emptyset$ for path cover and path pack inequalities (3.14) and (3.17). For path cover inequalities (3.14), we add an arc $t \in E^-$ to L^- if $\lambda_j x_t^* < y_t^*$ and $\lambda_j < c_t$. We repeat the separation process for all subsets $[k, \ell] \subseteq [1, n]$.

Results

We report multiple performance measures. Let z_{INIT} be the objective function value of the initial LP relaxation and z_{ROOT} be the objective function value of the LP relaxation

Table 3.1: Effect of path size on the performance.

				$p = 1$		$p \leq 5$				$p \leq 0.5 \times n$				$p \leq n$			
n	f	c	init	gap imp	cuts	gap imp		cuts		gap imp		cuts		gap imp		cuts	
			gap	(m)spi	(m)spi	spi	mspi	spi	mspi	spi	mspi	spi	mspi	spi	mspi	spi	mspi
50	100	2	14.8	34%	21	87%	52%	212	106	97%	52%	1164	158	97%	52%	1233	158
		5	44.3	56%	52	99%	69%	303	148	99%	69%	664	151	99%	69%	664	151
		10	58.3	60%	54	96%	70%	277	147	99%	70%	574	167	99%	70%	574	167
	200	2	14.5	32%	22	81%	57%	257	133	96%	61%	1965	241	96%	61%	2387	241
		5	49.8	43%	44	99%	57%	378	162	100%	57%	1264	171	100%	57%	1420	171
		10	66.3	38%	47	98%	50%	392	169	99%	51%	1235	197	99%	51%	1283	197
	500	2	19.1	23%	19	77%	48%	266	128	90%	49%	2286	222	90%	49%	3249	222
		5	54.4	35%	36	99%	49%	522	185	100%	49%	1981	205	100%	49%	2074	205
		10	73.0	34%	43	99%	40%	498	196	99%	40%	1336	236	99%	40%	1445	236
	1000	2	14.6	18%	15	73%	39%	264	99	83%	40%	2821	211	83%	40%	3918	212
		5	59.7	31%	36	98%	45%	487	201	100%	45%	2077	239	100%	45%	2329	239
		10	76.9	30%	41	100%	36%	529	215	100%	37%	1935	268	100%	37%	2149	268
Average:			**45.5**	**36%**	**36**	**92%**	**51%**	**365**	**157**	**97%**	**52%**	**1609**	**206**	**97%**	**52%**	**1894**	**206**

after all the valid inequalities added. Moreover, let z_{UB} be the objective function value of the best feasible solution found within time/memory limit among all experiments for an instance. Let `init gap`$= 100 \times \frac{z_{\text{UB}} - z_{\text{INIT}}}{z_{\text{UB}}}$, `root gap`$= 100 \times \frac{z_{\text{UB}} - z_{\text{ROOT}}}{z_{\text{UB}}}$. We compute the improvement of the relaxation due to adding valid inequalities as `gap imp`$= 100 \times \frac{\texttt{init gap} - \texttt{root gap}}{\texttt{init gap}}$. We also measure the optimality gap at termination as `end gap` $= \frac{z_{\text{UB}} - z_{\text{LB}}}{z_{\text{UB}}}$, where z_{LB} is the value of the best lower bound given by CPLEX. We report the average number of valid inequalities added at the root node under column `cuts`, average elapsed time in seconds under `time`, average number of branch-and-bound nodes explored under `nodes`. If there are instances that are not solved to optimality within the time/memory limit, we report the average end gap and the number of unsolved instances under `unslvd` next to `time` results. All numbers except initial gap, end gap and time are rounded to the nearest integers.

In Tables 3.1, 3.2 and 3.3, we present the performance with the path cover (3.14) and path pack (3.17) inequalities under columns `spi`. To understand how the forward and backward path arc capacities affect the computational performance, we also apply them to the single node relaxations obtained by merging a path into a single node, where the capacities of forward and backward path arcs within a path are considered to be infinite. In this case, the path inequalities reduce to the flow cover and flow pack inequalities. These results are presented under columns `mspi`.

In Table 3.1, we focus on the impact of path size on the gap improvement of the path cover and path pack inequalities for instances with $n = 50$. In the columns under $p = 1$, we obtain the same results for both `mspi` and `spi` since the paths are singleton nodes.. We present these results under `(m)spi`. In columns $p \leq q$, we add valid inequalities for paths of size $1, \ldots, q$ and observe that as the path size increases, the gap improvement of the path inequalities increase rapidly. On average 97% of the initial gap is closed as longer paths are used. On the other hand, flow cover and pack inequalities from merged paths reduce about half of the initial gap. These results underline the importance of exploiting path arc capacities for strengthening the formulations. We also observe that the increase in gap improvement diminishes as path size grows. We choose a conservative maximum path size limit of $0.75 \times n$ for the experiments reported in Tables 3.2, 3.3 and 3.4.

In Table 3.2, we investigate the computational performance of path cover and path pack inequalities independently. We present the results for path cover inequalities under columns titled `cov`, for path pack inequalities under `pac` and for both of them under the columns titled `spi`. On average, path cover and path pack inequalities independently close the gap by 63% and 53%, respectively. However, when used together, the gap improvement is 96%, which shows that the two classes of inequalities complement each other very well.

In Table 3.3, we present other performance measures as well for instances with 50, 100, and 150 nodes. We observe that the forward and backward path arc capacities have a large impact on the performance level of the path cover and pack inequalities. Compared to flow cover and pack inequalities added from merged paths, path cover and path pack inequalities reduce the number of nodes and solution times by orders of magnitude. This is mainly due to better integrality gap improvement (50% vs 95% on average).

In Table 3.4, we examine the incremental effect of path cover and path pack inequalities over the fixed-charge network cuts of CPLEX, namely flow cover, flow path and multi-commodity flow cuts. Under `cpx`, we present the performance of flow cover, flow path and multi-commodity flow cuts added by CPLEX and under `cpx_spi`, we add path cover and path pack inequalities addition to these cuts. We observe that with the addition of path cover and pack inequalities, the gap improvement increases from 86% to 95%. The number of branch and bound nodes explored is reduced about 900 times. Moreover, with path cover and path pack inequalities the average elapsed time is reduced to almost half and the total number of unsolved instances reduces from 13 to 6 out of 180 instances.

Tables 3.1, 3.2, 3.3 and 3.4 show that submodular path inequalities are quite effective in tackling lot-sizing problems with finite arc capacities. When added to the LP relaxation, they improve the optimality gap by 95% and the number of branch and bound nodes explored decreases by a factor of 1000. In conclusion, our computational experiments indicate that the use of path cover and path pack inequalities is beneficial in improving the performance of the branch-and-cut algorithms.

Table 3.2: Effect of path cover (`cov`) and path pack (`pac`) inequalities when used separately and together (`spi`).

n	f	c	init gap	gap imp			nodes			cuts			time		
				cov	pac	spi	cov	pac	spi	cov	pac	spi	cov	pac	spi
50	100	2	14.8	62%	18%	96%	273	6258	7	759	13	1151	0.3	0.6	0.2
		5	44.3	75%	37%	97%	319	12366	9	357	25	435	0.1	1.0	0.1
		10	58.3	77%	39%	93%	213	29400	63	290	11	386	0.1	2.1	0.1
	200	2	14.5	73%	34%	92%	148	3268	18	1593	41	2469	0.7	0.3	0.6
		5	49.8	67%	38%	100%	576	11525	3	736	36	1022	0.4	0.9	0.1
		10	66.3	61%	48%	97%	226	8799	14	619	32	739	0.2	0.7	0.1
	500	2	19.1	57%	57%	92%	635	1825	19	1587	316	2577	1.6	0.3	1.0
		5	54.4	56%	75%	99%	348	363	1	902	148	1164	0.3	0.1	0.1
		10	73.0	59%	65%	97%	8410	5284	11	727	67	698	3.0	0.5	0.1
	1000	2	14.6	61%	65%	90%	278	258	60	1362	427	2094	0.8	0.3	0.9
		5	59.7	59%	81%	100%	1063	208	2	1673	364	1792	1.3	0.1	0.1
		10	76.9	51%	77%	99%	3791	1452	5	1202	155	1032	2.1	0.2	0.1
Average:			**45.5**	**63%**	**53%**	**96%**	**1357**	**6751**	**18**	**984**	**136**	**1297**	**0.9**	**0.6**	**0.3**

Table 3.3: Comparison of path inequalities applied to paths (`spi`) versus applied to merged paths (`mspi`).

n	f	c	init gap	gapimp		nodes		cuts		time (endgap:unslvd)	
				spi	mspi	spi	mspi	spi	mspi	spi	mspi
50	100	2	14.8	96%	52%	7	430	1151	195	0.2	0.2
		5	44.3	97%	69%	9	553	435	146	0.1	0.1
		10	58.3	93%	70%	63	468	386	160	0.1	0.1
	200	2	14.5	92%	59%	18	330	2469	226	0.6	0.2
		5	49.8	100%	57%	3	1112	1022	176	0.1	0.3
		10	66.3	97%	53%	14	615	739	173	0.1	0.2
	500	2	19.1	92%	43%	19	2041	2577	238	1.0	0.7
		5	54.4	99%	48%	1	705	1164	214	0.1	0.3
		10	73.0	97%	48%	11	5659	698	248	0.1	1.4
	1000	2	14.6	90%	45%	60	612	2094	301	0.9	0.4
		5	59.7	100%	50%	2	2265	1792	241	0.1	0.7
		10	76.9	99%	40%	5	9199	1032	314	0.1	2.3
100	100	2	13.9	95%	65%	39	7073	3114	410	1.3	3.2
		5	42.2	98%	70%	19	20897	1337	297	0.2	4.8
		10	57.8	94%	59%	230	395277	1298	346	0.4	88.2
	200	2	16.1	89%	56%	290	151860	6919	478	11.0	58.4
		5	47.6	99%	55%	7	455192	2355	331	0.3	126.1
		10	65.7	95%	54%	104	4130780	1872	399	0.5	962.3 (1.1:1)
	500	2	17.5	84%	36%	1047	956745	11743	475	47.7	390.9
		5	53.9	99%	41%	4	332041	3874	444	0.4	115.5
		10	72.9	96%	42%	34	1175647	1495	474	0.3	352.5
	1000	2	17.9	91%	41%	173	57147	10919	570	21.3	23.0
		5	58.5	100%	45%	1	284979	3261	501	0.3	92.8
		10	75.7	97%	36%	88	3158262	2358	657	0.5	1047.0 (0.7:1)
150	100	2	13.2	94%	64%	336	163242	5159	704	11.3	107.6
		5	44.8	99%	65%	17	3024118	2087	431	0.5	929.6
		10	56.9	95%	65%	404	7254052	1492	476	0.9	2087.3 (0.7:1)
	200	2	14.7	92%	53%	519	2772494	12636	744	27.2	1390.6 (0.1:1)
		5	48.1	99%	55%	15	3802938	2462	508	0.6	1483.0 (1.2:2)
		10	65.2	95%	50%	330	9377122	2047	567	0.9	3585.9 (8.2:5)
	500	2	19.3	86%	33%	7927	7619674	22275	792	1087.3	3165.6 (4.0:4)
		5	54.4	100%	45%	7	7873043	4927	641	0.8	2813.6 (4.3:3)
		10	72.3	97%	41%	250	10219548	2678	713	1.2	3422.8 (11.0:5)
	1000	2	19.6	88%	34%	2824	7316675	33729	724	804.8	3260.3 (2.5:3)
		5	57.5	100%	39%	2	9661586	6710	709	0.8	3578.9 (9.7:5)
		10	75.8	96%	37%	99	9910056	3981	829	1.2	3412.3 (15.2:5)
Average:			**45.2**	**95%**	**50%**	**416**	**2504012**	**4619**	**440**	**56.3**	**903.0 (1.6:36)**

Table 3.4: Effectiveness of the path inequalities when used together with CPLEX's network cuts.

				gapimp		nodes		time (endgap:unslvd)			
n	f	c	init gap	cpx_spi	cpx	cpx_spi	cpx	cpx_spi		cpx	
100		2	13.9	96%	85%	35	1715	1.0		0.5	
	100	5	42.2	99%	97%	5	75	0.2		0.1	
		10	57.8	99%	93%	10	2970	0.3		0.6	
		2	16.1	90%	79%	288	9039	6.6		2.1	
	200	5	47.6	99%	95%	7	52	0.3		0.1	
		10	65.7	97%	89%	61	3186	0.4		0.7	
		2	17.5	85%	63%	1232	455068	57.3		95.2	
	500	5	53.9	99%	94%	6	92	0.4		0.1	
		10	72.9	98%	89%	11	4621	0.4		0.9	
		2	17.9	91%	76%	173	18109	22.2		3.6	
	1000	5	58.5	100%	93%	1	156	0.3		0.1	
		10	75.7	97%	85%	117	5297	0.7		1.0	
150		2	13.2	94%	86%	365	60956	9.7		19.0	
	100	5	44.8	100%	97%	5	119	0.4		0.1	
		10	56.9	99%	92%	16	15929	0.5		3.9	
		2	14.7	92%	80%	954	216436	44.9		66.7	
	200	5	48.1	99%	96%	11	284	0.5		0.2	
		10	65.2	97%	91%	181	3992	0.9		1.2	
		2	19.3	86%	69%	7647	4943603	1049.9		1215.1	(0.2:1)
	500	5	54.4	100%	94%	5	5434	0.8		1.6	
		10	72.3	97%	88%	141	141211	1.4		35.9	
		2	19.6	88%	71%	3051	2788993	917.4	(0.2:1)	619.4	(0.4:1)
	1000	5	57.5	100%	90%	3	4322	0.8		1.2	
		10	75.8	96%	89%	196	10588	2.5		2.8	
200		2	14.1	94%	82%	1623	864841	32.2		384.0	
	100	5	42.7	100%	97%	8	213	0.5		0.1	
		10	57.5	99%	93%	26	45263	0.7		13.8	
		2	16.3	89%	78%	4279	5634851	259.9		1940.4	(0.1:1)
	200	5	48.0	99%	95%	13	1310	0.9		0.5	
		10	65.0	98%	90%	128	163145	1.2		52.3	
		2	16.3	88%	72%	8083	6805861	1226.3	(0.3:1)	2137.6	(0.7:3)
	500	5	54.5	99%	93%	7	6606	1.6		2.2	
		10	72.0	96%	90%	376	900152	3.2		302.4	
		2	18.0	82%	63%	13906	9894589	3000.5	(1.2:4)	2835.9	(3.0:5)
	1000	5	57.9	100%	94%	4	1977	3.4		0.8	
		10	75.6	96%	84%	704	6127929	15.0		1785.0	(1.8:2)
Average:			**45.0**	**95%**	**86%**	**1213**	**1087194**	**185.1**	**(0.0:6)**	**320.2**	**(0.2:13)**

Chapter 4

Path Pack Inequalities for Lot-sizing Problems with Backlogging and Inventory Bounds

Given the upper-bounds of inventory holding and backlogging values for each time period, the lot-sizing with backlogging and inventory bounds (LSBIB) problem aims to find a production schedule that minimizes the total cost of fixed and unit production, inventory holding and backlogging. In this chapter, we give valid inequalities for single item LSBIB that utilizes inventory and backlogging capacities as well as the underlying path structure of the lot-sizing problems.

There is a vast literature on lot-sizing problems. We refer the reader to Pochet and Wolsey (2006) for a detailed review of different lot-sizing problems. Moreover, Brahimi et al. (2006) survey single item and Karimi et al. (2003) overviews capacitated lot-sizing problems.

The uncapacitated version of lot-sizing with backlogging (ULSB) is tackled extensively in the literature. Pochet and Wolsey (1988) and Pochet and Wolsey (1994) carry out a polyhedral study and provide valid inequalities for ULSB. Agra and Constantino (1999) consider start-up costs and give an extended linear formulation for instances with Wagner-Whitin costs. Federgruen and Tzur (1993), Zangwill (1966), Zangwill (1969) and Ganas and Papachristos (2005) propose polynomial time algorithms. Küçükyavuz and Pochet (2009) provide an explicit description of the convex hull of ULSB.

For the capacitated lot-sizing with backlogging, Van Vyve (2006) gives a linear extended formulation for the constant production capacity case with infinite inventory and backlog capacities. Zhong et al. (2016) and Chu et al. (2013) give polynomial time algorithms for lot-sizing problems with infinite production and finite inventory/backlogging capacities. Constantino (2000) considers an extended multi-item capacitated lot-sizing framework with start-up costs and infinite inventory and backlogging capacities. The author provides a cutting plane algorithm and a class of valid inequalities. Wu et al. (2011) give two formulations for multi-item capacitated lot-sizing with backlogging and analyze the tightness of the lower bounds they provide.

The path structure arises naturally in network models of the lot-sizing problem. Atamtürk and Muñoz (2004) introduce valid inequalities for the capacitated lot-sizing problems with infinite inventory capacities. Atamtürk and Küçükyavuz (2005) give valid inequalities for the lot-sizing problems with finite inventory and infinite production capacities. Van Vyve (2013) introduces valid inequalities for uncapacitated fixed charge transportation problems. Van Vyve and Ortega (2004) and Gade and Küçükyavuz (2011) give valid inequalities and extended formulations for uncapacitated lot-sizing with fixed charges on stocks.

Lot-sizing problems are also special cases of fixed-charge network flow problems. For a single node relaxation of capacitated fixed-charge network problems, Padberg et al. (1985), van Roy and Wolsey (1986) and Gu et al. (1999) give flow cover, generalized flow cover and lifted flow cover inequalities. For a general fixed-charge network, Wolsey (1989) introduces submodular inequalities, however these inequalities have implicit coefficients. Atamturk et al. (2017) derive submodular path inequalities which are explicit description of submodular inequalities for a simple path sub-structure. In this work, we extend submodular path inequalities by considering backlogging arcs between consecutive nodes in a path. We show that all coefficients of these inequalities can be computed in linear time using maximum flow minimum cut equivalency. We give necessary and sufficient facet-defining condition. Finally, we show the effectiveness of these inequalities when used in a branch-and-cut algorithm.

Outline: The remainder of the chapter is organized as follows: In Section 4.1, we give the mathematical programming formulation and the underlying fixed-charge network flow problem of LSBIB. In Section 4.2, we introduce eleven classes of path pack inequalities that are valid for LSBIB. In Section 4.3, we lift these inequalities by incorporating the fixed charge variables of inventory and backlog arcs. Finally, in Section 4.4, we present some computational results that shows the effectiveness of both path pack inequalities introduced in Section 4.2 and lifted path pack inequalities introduced in Section 4.3.

4.1 Lot-sizing Problems with Inventory and Backlogging Bounds

We provide a mixed-integer programming formulation for single-item lot-sizing problem with inventory and backlogging capacities. Let $N = \{1, \ldots, n\}$ be the set of time periods and d_t be the demand at time $t \in N$. Without loss of generality, we assume time period i comes before time period j if and only if $i < j$. Let y_t, i_t and r_t be the production, inventory and backlogging units at time period t with upper bounds c_t, u_t and b_t respectively. In this chapter, we assume that c_t is a very large value for all $t \in N$. Let the binary variable x_t be 1 if there is production at time t, 0 otherwise and let f_t^p represent the fixed cost of production at time t. Let z_t and q_t be the binary variables that are equal to 1 if there is a non-zero amount of inventory held and a non-zero amount backlogged at time t, respectively. The fixed costs associated with z_t and q_t are represented by f_t^i and f_t^b, respectively. Let the unit cost of production, inventory holding and backlogging be represented by parameters v_t^p, v_t^i and v_t^b, respectively. The single-item lot-sizing problem with inventory and backlogging upper-bounds can be represented as the mathematical optimization problem:

$$
\begin{aligned}
& \min \quad \sum_{t \in V} f_t^p x_t + f_t^i z_t + f_t^b q_t + v_t^p y_t + v_t^i i_t + v_t^b r_t && \text{(4.1a)}\\
& \text{s. t.} \quad i_{t-1} - r_{t-1} + y_t - i_t + r_t = d_t, \quad t \in N, && \text{(4.1b)}\\
& \qquad 0 \leq y_t \leq c_t x_t, \quad t \in N, && \text{(4.1c)}\\
& \qquad 0 \leq i_t \leq u_t z_t, \quad t \in N, && \text{(4.1d)}\\
& \qquad 0 \leq r_t \leq b_t q_t, \quad t \in N, && \text{(4.1e)}\\
& \text{(LSBIB)} \quad x_t \in \{0,1\}, \quad t \in N, && \text{(4.1f)}\\
& \qquad z_t \in \{0,1\}, \quad t \in N, && \text{(4.1g)}\\
& \qquad q_t \in \{0,1\}, \quad t \in N, && \text{(4.1h)}\\
& \qquad i_0 = i_n = r_0 = r_n = 0. && \text{(4.1i)}
\end{aligned}
$$

where $u_0 = b_0 = u_n = b_n = 0$. Moreover, let $\mathcal{P}$ be the set of feasible solutions of formulation 4.1.

Throughout the chapter, we use the following notation: Let $[k, j] := \{k, k+1, \dots, j\}$ if $j \geq k$ and $\emptyset$ if $j < k$, $(a)^+ := \max\{0, a\}$ and $d_{kj} = \sum_{t=k}^{j} d_t$ if $j \geq k$ and 0 otherwise. Moreover, let $\dim(A)$ denote the dimension of a polyhedron A and $\text{conv}(S)$ be the convex hull of a set S.

We make some assumptions on the parameter set of the problem:

(1) $b_t + u_{t-1} \geq d_t, \quad t \in N$. In other words, the feasible set $F_t = \{x \in \mathcal{P} : \ y_t = 0\}$ is non-empty for all $t \in N$,

(2) $u_j > 0$ and $b_j > 0$ for all $j \in N \setminus \{n\}$,

(3) $d_j \geq 0$, for $j \in N$,

(4) $u_{j-1} \leq d_j + u_j$, for $j \in N$,

(5) $b_j \leq b_{j-1} + d_{j-1}$, for $j \in N$.

Assumptions (1) and (2) and enforced to ensure that $\dim\big(\text{conv}(\mathcal{P})\big) = 5n - 2$. Notice that, if assumption (1) does not hold for some $t \in N$, then $x_t = 1$ in all feasible solutions of LSBIB. Similarly, if $b_j = 0$ (or $u_j = 0$), then $r_j = 0$ ($i_j = 0$) at all feasible solutions of LSBIB which would reduce the dimension of conv($\mathcal{P}$). Assumptions (3), (4) and (5) significantly reduce the number of potential minimum cuts in a given a set of production arcs that are open which directly reduces the complexity of computing submodular inequality coefficients. The utilization of these assumptions will be further explained in Section 4.2.

There are $6n - 4$ decision variables in LSBIB. However, using constraints (4.1i), any feasible point in LSBIB can be described by $(\mathbf{y}, \mathbf{i}, \mathbf{x}, \mathbf{z}, \mathbf{q}, r_1)$, where $\mathbf{y}$ and $\mathbf{x}$ are n-dimensional and $\mathbf{i}$, $\mathbf{z}$ and $\mathbf{q}$ are $n-1$-dimensional vectors. Consequently, we observe that $\dim\big(\text{conv}(\mathcal{P})\big) = 5n - 2$.

One can represent the optimization problem LSBIB using a fixed-charge network where time periods are vertices and demand, production, inventory and backlog units are arcs as in Figure 4.1. Throughout the chapter, we refer to production arcs (s_N, j) as "arc j" for convenience. Let A be the union of all production, inventory and backlog arcs and let $N' := N \cup \{s_N, t_N\}$.

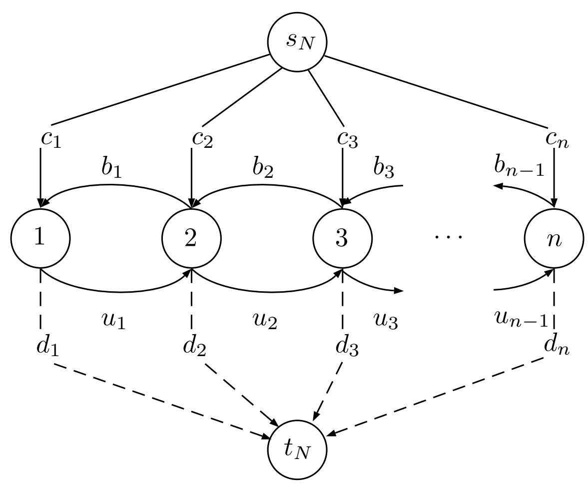


Figure 4.1: Fixed-charge network representation of a path.

Next, we review the path pack inequalities introduced in Atamturk et al. (2017) that are valid for any capacitated-path graph with fixed-charge variables.

4.2 Path pack inequalities

Let $[k, \ell]$ be a subset of consecutive time periods, where $k \geq 1$ and $\ell \leq n$. Let $E^+_{k\ell} := \{(i, j) \in A : i \notin [k, \ell], j \in [k, \ell]\}$. In other words, $E^+_{k\ell}$ is the set of production arcs $[k, \ell]$, the inventory arc $(k-1, k)$ and the backlog arc $(\ell+1, \ell)$. Similarly, let $E^-_{k\ell} := \{(i, j) \in A : i \in [k, \ell], j \notin [k, \ell]\}$. For the graph structure of lot-sizing problems the set $E^-_{k\ell}$ is equivalent to $\{(k, k-1), (\ell, \ell+1)\}$.

Let $S^+ \subseteq E^+_{k\ell}$, $L^+ \subseteq E^+_{k\ell} \setminus S^+$ and $S^- \subseteq E^-_{k\ell}$. We refer to the subsets (S^+, L^+, S^-) as the *objective sets*. For all $j \in [k, \ell]$, define $S^+_j = \{(j-1, j), (s_N, j)\} \cap E^+_{k\ell}$, $L^+_j = \{(j-1, j), (s_N, j)\} \cap E^+_{k\ell}$ and $S^-_j = \{(j, j-1), (j, j+1)\} \cap E^-_{k\ell}$.

In Atamturk et al. (2017), the authors introduce path pack inequalities which are derived from submodular inequalities introduced by Wolsey (1989). Each coefficient that appears in a submodular inequality is computed by solving a maximum flow problem. Letting the value function v be computed using the optimization problem

$$
\begin{aligned}
v(S^+, S^-) = \max \quad & \sum_{t \in S^+ \cup L^+} y_t - \sum_{E^-_{k\ell} \setminus S^-} y_t \\
\text{s. t.} \quad & i_{j-1} - r_{j-1} + y_j - i_j + r_j \le d_j, \quad j \in [k, \ell], \\
& 0 \le i_j \le u_j, \quad j \in [k, \ell], \\
& 0 \le r_j \le b_j, \quad j \in [k, \ell], \\
\text{(F2)} \quad & 0 \le y_t \le c_t, \quad t \in [k, \ell], \\
& i_0 = i_n = r_0 = r_n = 0, \\
& y_t = 0, \quad t \in E^+_{k\ell} \setminus S^+.
\end{aligned}
$$

Then, Wolsey (1989) shows that the the inequality

$$\sum_{t \in S^+ \cup L^+} y_t \le v(S^+, S^-) + \sum_{t \in L^+} \rho_t(S^+, S^-) x_t + \sum_{t \in E^-_{k\ell} \setminus S^-} y_t, \tag{4.2}$$

is valid for capacitated lot-sizing problems with fixed-charge variables, where

$$\rho_t(S^+, S^-) = v(S^+ \cup \{t\}, S^-) - v(S^+, S^-), \quad t \in L^+.$$

In the remainder of the chapter, for convenience, if $S^- = \emptyset$, we refer to $v(S^+, \emptyset)$ and $\rho_t(S^+, \emptyset)$ as $v(S^+)$ and $\rho_t(S^+)$, respectively.

For problems with an underlying path graph, Atamturk et al. (2017) show that all of the coefficients of inequality (4.2) can be computed in $O(n)$ time. In the remainder of this section, we introduce the steps necessary to describe path pack inequalities for LSBIB explicitly.

Let α^u_j and α^d_j be the minimum value of a cut on nodes $[k, j]$ that passes above and below node j, respectively. Similarly, let β^u_j and β^d_j be the minimum values of cuts on nodes $[j, \ell]$ that passes above and below node j respectively. We compute $\alpha^{\{u,d\}}_j$ by a forward recursion and $\beta^{\{u,d\}}_j$ by a backward recursion:

$$\alpha^u_j = \min\{\alpha^d_{j-1} + u_{j-1}, \alpha^u_{j-1}\} + c(S^+_j), \quad j \in [k, \ell], \tag{4.3}$$

$$\alpha^d_j = \min\{\alpha^d_{j-1}, \alpha^u_{j-1} + b_{j-1}\} + d_j + c(S^-_j), \quad j \in [k, \ell], \tag{4.4}$$

where $\alpha^u_{k-1} = \alpha^d_{k-1} = 0$ and

$$\beta^u_j = \min\{\beta^u_{j+1}, \beta^d_{j+1} + b_j\} + c(S^+_j), \quad j \in [k, \ell], \tag{4.5}$$
$$\beta^d_j = \min\{\beta^u_{j+1} + u_j, \beta^d_{j+1}\} + d_j + c(S^-_j), \quad j \in [k, \ell], \tag{4.6}$$

where $\beta^u_{\ell+1} = \beta^d_{\ell+1} = 0$ and for some arc set S the function c is defined as

$$c(S) := \sum_{t \in S \cap [k,\ell]} c_t + \sum_{t \in S \cap \{(k-1,k),(\ell,\ell+1)\}} u_t + \sum_{t \in S \cap \{(k,k-1),(\ell+1,\ell)\}} b_t.$$

Let m^u_j and m^d_j be the values of minimum cuts for nodes $[k, \ell]$ that pass above and below node j, respectively. Notice that

$$m^u_j = \alpha^u_j + \beta^u_j - c(S^+_j), \quad j \in [k, \ell] \tag{4.7}$$

and

$$m^d_j = \alpha^d_j + \beta^d_j - d_j - c(S^-_j), \quad j \in [k, \ell]. \tag{4.8}$$

For convenience, let

$$m_j := \min\{m^u_j, m^d_j\}, \quad j \in [k, \ell].$$

Then, the path pack inequalities which are valid for LSBIB can be expressed as

$$y(S^+ \cup L^+) \leq m_k + \sum_{j=k}^{\ell} \sum_{t \in L^+_j} \mu_j x_t + y(E^-_{k\ell} \setminus S^-), \tag{4.9}$$

where

$$\mu_j := (m^d_j - m^u_j)^+, \quad j \in [k, \ell]$$

and for some arc set C, the function y is defined as

$$y(C) := \sum_{t \in C \cap [k,\ell]} y_t + \sum_{t \in C \cap \{(k-1,k),(\ell,\ell+1)\}} i_t + \sum_{t \in C \cap \{(k,k-1),(\ell+1,\ell)\}} r_t.$$

4.2.1 Explicit inequalities for LSBIB

In this section, we give path pack inequalities where the coefficients are computed explicitly for LSBIB problems.

Proposition 4.1. The inequality

$$i_{k-1} + \sum_{t \in L^+} y_t \leq u_{k-1} + \sum_{j \in L^+} \big(\min\{d_{kj} + u_j - u_{k-1}, d_{k\ell} - u_{k-1}\}\big)^+ x_j + i_\ell + r_{k-1} \tag{4.10}$$

is valid for LSBIB.

Proof. Inequality (4.10) is equivalent to inequality (4.9), where $S^+ = \{(k-1,k)\}$, $S^- = \emptyset$ and $L^+ \subseteq [k,\ell]$. We compute α_j^u, α_j^d, β_j^u and β_j^d using the recursive equations (4.3)–(4.6). Due to assumption (4), we observe that

$$\alpha_j^u = u_{k-1}, \quad \forall j \in [k,\ell]. \tag{4.11}$$

Similarly, due to assumption (1), we observe that

$$\alpha_j^d = d_{kj}, \quad \forall j \in [k,\ell]. \tag{4.12}$$

Furthermore, using assumption (4) , it is easy to see that

$$\beta_j^u = \begin{cases} u_{k-1} & \text{if} \quad j = k, \\ 0 \quad \text{if} & j \in [k+1,\ell]. \end{cases} \tag{4.13}$$

and

$$\beta_j^d = \min\{u_j, d_{j+1,\ell}\} + d_j, \quad \forall j \in [k,\ell]. \tag{4.14}$$

As a result, we obtain the minimum cut values that pass above and below node j as

$$m_j^u = u_{k-1}$$

and

$$m_j^d = d_{kj} + \min\{u_j, d_{j+1\ell}\}$$

respectively, for each $j \in [k,\ell]$. Recall that the value of the overall minimum cut can be computed as

$$v(S^+) = \min\{m_j^u, m_j^d\}$$

using any $j \in [k,\ell]$. Now, suppose we would like to compute $\rho_j(S^+) = v(S^+ \cup \{j\}) - v(S^+)$. Since production arcs are assumed to be uncapacitated (i.e., very large), the minimum cut has to pass below node j when the production arc j is open. Consequently,

$$v(S^+ \cup \{j\}) = m_j^d$$

and $\rho_j(S^+) = \max\{m_j^d - m_j^u, 0\}$. Using the values computed above, we observe that for each $j \in L^+$,

$$\rho_j(S^+) = \big(\min\{d_{kj} + u_j - u_{k-1}, d_{k\ell} - u_{k-1}\}\big)^+.$$

□

Proposition 4.2. The inequality

$$r_\ell + \sum_{t \in L^+} y_t \le b_\ell + \sum_{j \in L^+} \big(\min\{d_{j\ell} + b_{j-1} - b_\ell, d_{k\ell} - b_\ell\}\big)^+ x_j + i_\ell + r_{k-1}, \tag{4.15}$$

is valid for LSBIB.

Proof. Inequality (4.15) is equivalent to inequality (4.9), where $S^+ = \{(\ell+1, \ell)\}$, $S^- = \emptyset$ and $L^+ \subseteq [k, \ell]$. The proof follows very similar to that of proposition 4.1. □

Proposition 4.3. The inequality

$$i_{k-1} + \sum_{t \in L^+} y_t \le u_{k-1} + \sum_{j \in L^+} (d_{kj} + u_j - u_{k-1}) x_j + r_{k-1} \tag{4.16}$$

is valid for LSBIB.

Proof. Inequality (4.16) is equivalent to inequality (4.9), where $S^+ = \{(k-1, k)\}$, $S^- = \{(\ell, \ell+1)\}$ and $L^+ \subseteq [k, \ell]$. □

Proposition 4.4. The inequality

$$i_{k-1} + \sum_{t \in L^+} y_t \le u_{k-1} + \sum_{j \in L^+} \big(\min\{b_{k-1} + d_{kj-1}, u_{k-1} + b_{j-1}\} + u_j + d_j - u_{k-1}\big)^+ x_j \tag{4.17}$$

is valid for LSBIB.

Proof. Inequality (4.17) is equivalent to inequality (4.9), where $S^+ = \{(k-1, k)\}$, $S^- = \{(k, k-1), (\ell, \ell+1)\}$ and $L^+ \subseteq [k, \ell]$. □

Remark 4.1. If the backlogging upper bounds are set to zero, then this problem reduces to the lot-sizing problem studied in Atamtürk and Küçükyavuz (2005). If the backlog arc capacities are zero, then inequality (4.17) reduces to inequality (5) of Atamtürk and Küçükyavuz (2005).

Proposition 4.5. The inequality

$$i_{k-1} + \sum_{t \in L^+} y_t \le u_{k-1} + \sum_{t \in L^+} \rho_t x_j + i_\ell, \tag{4.18}$$

where

$$\rho_t = \big(\min\{u_j + d_{kj} + b_{k-1} - u_{k-1}, u_j + d_j + b_{j-1}, d_{k\ell} + b_{k-1} - u_{k-1}, d_{j\ell} + b_{j-1}\}\big)^+$$

for all $t \in L^+$, is valid for LSBIB.

Proof. Inequality (4.18) is equivalent to inequality (4.9), where $S^+ = \{(k-1,k)\}$, $S^- = \{(k,k-1)\}$ and $L^+ \subseteq [k,\ell]$. □

Remark 4.2. If the backlog arc capacities are zero, then inequality (4.18) reduces to inequality (3) of Atamtürk and Küçükyavuz (2005).

Proposition 4.6. The inequality

$$r_\ell + \sum_{t \in L^+} y_t \le b_\ell + \sum_{j \in L^+} \big(d_{j\ell} + b_{j-1} - b_\ell\big) x_j + i_\ell, \tag{4.19}$$

is valid for LSBIB.

Proof. Inequality (4.19) is equivalent to inequality (4.9), where $S^+ = \{(\ell+1,\ell)\}$, $S^- = \{(k,k-1)\}$ and $L^+ \subseteq [k,\ell]$. □

Proposition 4.7. The inequality

$$r_\ell + \sum_{t \in L^+} y_t \le b_\ell + \sum_{j \in L^+} \big(\min\{b_\ell + u_j, d_{j+1,\ell} + u_\ell\} + d_j + b_{j-1} - b_\ell\big)^+ x_j, \tag{4.20}$$

is valid for LSBIB.

Proof. Inequality (4.20) is equivalent to inequality (4.9), where $S^+ = \{(\ell+1,\ell)\}$, $S^- = \{(k,k-1),(\ell,\ell+1)\}$ and $L^+ \subseteq [k,\ell]$. □

Proposition 4.8. The inequality

$$r_\ell + \sum_{j \in L^+} y_j \le b_\ell + \sum_{j \in L^+} \rho_j x_j + r_{k-1}, \tag{4.21}$$

where

$$\rho_j = \big(\min\{u_j + b_{j-1} + d_j, u_j + d_{kj}, d_{j\ell} + u_\ell + b_{j-1} - b_\ell, d_{k\ell} + u_\ell - b_\ell\}\big)^+$$

for all $j \in L^+$, is valid for LSBIB.

Proof. Inequality (4.21) is equivalent to inequality (4.9), where $S^+ = \{(\ell+1,\ell)\}$, $S^- = \{(\ell,\ell+1)\}$ and $L^+ \subseteq [k,\ell]$. □

Proposition 4.9. The inequality

$$i_{k-1} + r_\ell + \sum_{j \in L^+} y_j \le \min\{u_{k-1} + b_\ell, b_{k-1} + d_{k\ell} + u_\ell\} + \sum_{j \in L^+} \rho_j x_j, \tag{4.22}$$

where

$$\rho_j = \big(\min\{d_{k,j-1} + b_{k-1}, u_{k-1} + b_{j-1}\} + \min\{b_\ell + u_j, d_{j+1,\ell} + u_\ell\} + d_j - u_{k-1} - b_\ell\big)^+$$

for all $j \in L^+$, is valid for LSBIB.

Proof. Inequality (4.22) is equivalent to inequality (4.9), where $S^+ = \{(k-1,k),(\ell+1,\ell)\}$, $S^- = \{(k,k-1),(\ell,\ell+1)\}$ and $L^+ \subseteq [k,\ell]$. □

Proposition 4.10. The inequality

$$i_{k-1} + r_\ell + \sum_{j \in L^+} y_j \leq \min\{u_{k-1} + b_\ell, d_{k\ell} + u_\ell\} + \sum_{j \in L^+} \rho_j x_j + r_{k-1}, \tag{4.23}$$

where

$$\rho_j = \big(\min\{b_\ell + u_j, d_{j+1,\ell} + u_\ell\} + d_{kj} - u_{k-1} - b_\ell\big)^+$$

for all $j \in L^+$, is valid for LSBIB.

Proof. Inequality (4.23) is equivalent to inequality (4.9), where $S^+ = \{(k-1,k),(\ell+1,\ell)\}$, $S^- = \{(\ell,\ell+1)\}$ and $L^+ \subseteq [k,\ell]$. □

Proposition 4.11. The inequality

$$i_{k-1} + r_\ell + \sum_{j \in L^+} y_j \leq \min\{u_{k-1} + b_\ell, d_{k\ell} + b_{k-1}\} + \sum_{j \in L^+} \rho_j x_j + i_\ell, \tag{4.24}$$

where

$$\rho_j = \big(\min\{d_{k,j-1} + b_{k-1}, u_{k-1} + b_{j-1}\} + d_{j\ell} - u_{k-1} - b_\ell\big)^+$$

for all $j \in L^+$, is valid for LSBIB.

Proof. Inequality (4.24) is equivalent to inequality (4.9), where $S^+ = \{(k-1,k),(\ell+1,\ell)\}$, $S^- = \{(k,k-1)\}$ and $L^+ \subseteq [k,\ell]$. □

Remark 4.3. Selecting $S^+ = \{(k-1,k),(\ell+1,\ell)\}$ and $S^- = \emptyset$ gives a weak submodular inequality. Due to assumptions (1), (4) and (5) we know that $v(\{(k-1,k),(\ell+1,\ell)\}) = d_{k\ell}$. Notice that, for this set selection, the sum of constraints in (F2) implies that $d_{k\ell}$ is the largest value that the function v can take. As a result, $\rho_j(S^+) = 0$ for all $j \in L^+$.

Remark 4.4. In inequalities (4.10), (4.16), (4.17), (4.18), (4.15), (4.19), (4.20), (4.21), (4.22), (4.23) and (4.24), if $\rho_j(S^+) \geq u_j + b_{j-1} + d_j$ for any $j \in L^+$, then the inequality is weak.

Proof. Suppose for an arc $j \in L^+$, $\rho_j(S^+) \geq u_j + b_{j-1} + d_j$. Then, the path pack inequality where arc j is dropped from L^+ summed with $y_j \leq (u_j + b_{j-1} + d_j)x_j$ gives an inequality at least as strong as the initial path pack inequality. □

4.2.2 Finding violated inequalities

In this section, we discuss how to find inequalities (4.10), (4.16), (4.17), (4.18), (4.15), (4.19), (4.20), (4.21), (4.22), (4.23) and (4.24) violated by a given point $(\bar{x}, \bar{y}, \bar{s}, \bar{r}, \bar{q}, \bar{z}) \in \mathbb{R}^{6n}$. For a given $[k, \ell] \subseteq [1, n]$, and the sets S^+ and S^-, the set L^+ that maximizes the violation of these inequalities can be found in linear time. A production arc $t \in [k, \ell]$ is added to L^+ if and only if

$$\bar{y}_t > \rho_t \bar{x}_t,$$

where ρ_t is the coefficient of x_t if t is added to L^+.

In inequalities (4.10), (4.15), (4.16), (4.17), (4.18) , (4.19), (4.20), (4.21), (4.22), (4.23) and (4.24), we assume that all of the inventory and backlog arcs are open. In the next section, we extend these inequalities by incorporating the fixed charge variables of inventory and backlog arcs.

4.3 Inventory and backlog fixed charge variables

Let $H = \{h_1, \ldots, h_{|H|}\}$ be a subset of inventory arc indices in increasing order. In this convention, $h_i = j$ refers to the inventory arc $(j, j+1)$. For an inventory arc $(j, j+1)$ we define $m(j) = \min\{t \in L^+ \cup \{(\ell+1)\} : t > j\}$. Moreover, let $M \supseteq \cup_{i=1}^{p} m(h_i)$. Alternatively, we represent this set as $M = \{m_1, \ldots, m_{|M|}\}$. For convenience, let $m_0 := k$, $m_{|M|+1} := \ell$ and $h_0 = k$, $h_{|H|+1} = \ell$. Finally, let $L(t) = L^+ \cap [t+1, \ell]$.

Proposition 4.12. The inequality

$$i_{k-1} + \sum_{j \in L^+} y_j \leq u_{k-1} + \sum_{j \in L^+} \rho_j x_j + i_\ell + r_{k-1} + \sum_{j \in M} \delta_j q_{j-1}, \tag{4.25}$$

where

$$\rho_j = \begin{cases} \min\{d_{kj} + u_j - u_{k-1}, d_{k\ell} - u_{k-1}, u_j + d_j, d_{j\ell}\} \text{ if } j \in M, \\ \min\{d_{kj} + u_j - u_{k-1}, d_{k\ell} - u_{k-1}, u_j + d_j, d_{j\ell}\} \\ \qquad + \min\{(d_{k,j-1} - u_{k-1})^+, d_{m_i,j-1}\} \quad \text{if } m_i \leq j \leq m_{i+1}, i = 0, \ldots, |M| \end{cases} \tag{4.26}$$

and

$$\delta_{m_i} = \min\{b_{m_i - 1}, d_{m_{i-1}, m_i - 1}\}, \quad \forall\, m_i \in M \tag{4.27}$$

is valid for LSBIB.

Proof. Let $(x^*, y^*, i^*, r^*, q^*, z^*)$ be a feasible solution to LSBIB. We derive the coefficients ρ_j and δ_j are derived from the submodular function v where

$$\delta_{m_j} = v(\{m_j\}|q_{m_i-1} = 0, \forall i \in [1, j-1]) - v(\{m_j\}|q_{m_i-1} = 0, \forall i \in [1, j]) \qquad (4.28)$$

and

$$\rho_j = \rho_j(S^+|q_{i-1} = 0, \forall i \in M). \qquad (4.29)$$

For feasible solutions where $q^*_{t-1} = 0$ for all $t \in M$, inequality (4.25) is equivalent to submodular inequality (4.9) with $S^+ = \{(k-1, k)\}$ for the lot-sizing graph where the backlog arcs $(t, t-1)$ for $t \in M$ have capacities of zero. Consequently, inequality (4.25) is valid when $q^*_{t-1} = 0$ for all $t \in M$.

Now, let $\hat{L}^+ = \{j \in L^+ : x^*_j = 1\}$, $\hat{M} := \{t \in M : q^*_{t-1} = 1\}$ and suppose $\hat{M} \neq \emptyset$. From the definition of v, with the objective $i_{k-1} + \sum_{t \in L^+} y_t - i_\ell - r_{k-1}$, it is clear that

$$i^*_{k-1} + \sum_{j \in \hat{L}^+} y^*_j - i^*_\ell - r^*_{k-1} \leq v(\{s\} \cup \hat{L}^+|q_{t-1} = 0, \forall t \in M \setminus \hat{M}).$$

Without loss of generality, let $\hat{M} = \{\hat{m}_1, \ldots, \hat{m}_{|\hat{M}|}\}$. First, we observe that

$$v(\{s\}) + \sum_{t \in \hat{L}^+} \rho_t(\{s\}|q_{j-1} = 0, \forall j \in M) \geq v(\{s\} \cup \hat{L}|q_{j-1} = 0, \forall j \in M).$$

To obtain the inequality above, we use the submodularity of the function v and the fact that $v(\{s\}) = v(\{s\}|q_{j-1} = 0, \forall j \in M)$. This is because maximum flow sent by the inventory arc $\{k-1, k\}$ is u_{k-1} and only depends on the inventory arcs of the path and not on backlog arcs.

In the remainder of the proof, we use induction to show that the right hand side of (4.25) at this solution has the following relationship

$$u_{k-1} + \sum_{j \in \hat{L}^+} \rho_j + \sum_{t \in \hat{M}} \delta_t \geq v(\{s\} \cup \hat{L}^+|q_{t-1} = 0, \forall t \in M \setminus \hat{M}).$$

First, we make some observations on the values δ_j. The flow sent on a production arc j while computing $v(\{j\})$ flows on three directions: (i) to satisfy the demand d_j, (ii) through the backlog arc $(j, j-1)$ and (iii) through the inventory arc $(j, j+1)$. If the backlog arc $(j, j-1)$ is closed, then the flow corresponding to $v(\{j\}|q_{j-1} = 0)$, consists of the latter two components. As a result,

$$v(\{j\}) = v(\{j\}|q_{j-1} = 0) + v(\{j\}|z_j = 0) - d_j.$$

This logic extends to the δ_{m_j} values as well. The definition in (4.28) is equivalent to the maximum change in the maximum flow value sent on arc m_j when the backlog arc

(m_j, m_j-1) is opened, given that the backlog arcs $\{(m_1, m_1-1), \ldots, (m_{j-1}, m_{j-1}-1)\}$ are closed.

In the base case, let $\hat{m}_{|\hat{M}|} = m_t$ for some $m_t \in M$ and suppose that all backlog arcs $(j, j-1)$ for $j \in M$ are closed. When the backlog arc (m_t, m_t-1) is opened, the maximum flow that can be sent on it through the production arc m_t is equal to δ_{m_t}. Due to assumption (1), among the production arcs $[m_t, \ell]$, the production arc m_t can send the largest flow on the backlog arc (m_t, m_t-1). Consequently,

$$\begin{aligned} v(\{s\} \cup \hat{L} | q_{j-1} = 0, \forall j \in M \setminus \{m_t\}) - v(\{s\} \cup \hat{L} | q_{j-1} = 0, \forall j \in M) \\ \leq v(\{m_t\} | q_{m_i-1} = 0, \forall i \in [1, t-1]) - v(\{m_t\} | q_{m_i-1} = 0, \forall i \in [1, t])) \quad (4.30) \end{aligned}$$

which proves the base case of the induction.

At level j of the induction, assuming that

$$\begin{aligned} v(\{s\} \cup \hat{L} | q_{i-1} = 0, \forall j \in M \setminus \{\hat{m}_{j+2}, \ldots, \hat{m}_{|\hat{M}|}\}) + \delta_{\hat{m}_{j+1}} \\ \geq v(\{s\} \cup \hat{L} | q_{i-1} = 0, \forall i \in M \setminus \{\hat{m}_{j+1}, \ldots, \hat{m}_{|\hat{M}|}\}) \end{aligned}$$

holds we show

$$\begin{aligned} v(\{s\} \cup \hat{L} | q_{i-1} = 0, \forall i \in M \setminus \{\hat{m}_{j+1}, \ldots, \hat{m}_{|\hat{M}|}\}) + \delta_{\hat{m}_j} \\ \geq v(\{s\} \cup \hat{L} | q_{i-1} = 0, \forall i \in M \setminus \{\hat{m}_j, \ldots, \hat{m}_{|\hat{M}|}\}). \end{aligned}$$

Using the same logic as the base case, we know that among the production arcs $[m_j, \ell]$ the arc $\hat{m}_j$ can send the largest flow on the backlog arc $(\hat{m}_j, \hat{m}_j - 1)$. Let $m_l = \hat{m}_j$ for $m_l \in M$, then

$$\begin{aligned} v(\{s\} \cup \hat{L} | q_{i-1} = 0, \forall i \in M \setminus \{\hat{m}_{j+1}, \ldots, \hat{m}_{|\hat{M}|}\}) \\ - v(\{s\} \cup \hat{L} | q_{i-1} = 0, \forall i \in M \setminus \{\hat{m}_j, \ldots, \hat{m}_{|\hat{M}|}\}) \\ \leq v(\{\hat{m}_j\} | q_{m_i-1} = 0, \forall i \in [1, l-1]) - v(\{\hat{m}s_j\} | q_{m_i-1} = 0, \forall i \in [1, l])) \quad (4.31) \end{aligned}$$

which shows that the induction holds.

Consequently, we show that inequality (4.25) is valid since

$$\begin{aligned} i^*_{k-1} + \sum_{j \in \hat{L}^+} y^*_j - i_\ell - r_{k-1} &\leq v(\{s\} \cup \hat{L}^+ | q_{t-1} = 0, \forall t \in M \setminus \hat{M}) \\ &\leq u_{k-1} + \sum_{j \in \hat{L}^+} \rho_j + \sum_{t \in \hat{M}} \delta_t. \end{aligned}$$

□

Proposition 4.13. The inequality

$$i_{k-1} + \sum_{j \in L^+} y_j + \sum_{j \in H} \gamma_j (1 - z_j) \leq u_{k-1} + \sum_{j \in L^+} \rho_j x_j + i_\ell + r_{k-1} + \sum_{j \in M} \delta_j q_{j-1}, \quad (4.32)$$

where ρ_j for $j \in L^+$ and δ_j for $j \in M$ are defined as in (4.26) and (4.27) respectively and

$$\gamma_{h_i} = \begin{cases} \min\{(u_{k-1} - d_{kh_i})^+, d_{h_i+1,h_{i+1}}\} & \text{if } m(h_i) = m(h_{(i+1)}), \\ \min\{(u_{k-1} - d_{kh_i})^+, d_{h_i+1,m(h_i)-1}\} & \text{if } m(h_i) < m(h_{(i+1)}) \end{cases} \quad (4.33)$$

is valid for LSBIB.

Proof. The coefficients of γ_{h_i} can be written in terms of the functions v and ρ.

$$\begin{aligned} \gamma_{h_i} = \rho_s(L(h_i) | z_{h_j} = 0, \forall j \in [i+1, \tau], q_{j-1} = 0, \forall j \in M) \\ - \rho_s(L(h_i) | z_{h_j} = 0, \forall j \in [i, \tau], q_{j-1} = 0, \forall j \in M). \end{aligned}$$

Similarly, the definitions for δ_j and ρ_j are provided in (4.28) and (4.29).

Similar to the lot-sizing without backlog arcs, these coefficients depend on the order of variables in sets H and M. In this chapter, we assume that the elements of H and M are in increasing order.

Let $(x^*, y^*, i^*, r^*, q^*, z^*)$ be a feasible solution to LSBIB. For feasible solutions where $z_t^* = 1$ for all $t \in [k, \ell]$, inequality (4.32) is equivalent to (4.25) and valid for LSBIB.

Let $\hat{L}^+ := \{j \in L^+ : x_j^* = 1\}$, $\hat{H} := \{j \in H : z_j^* = 0\}$ and $\hat{M} := \{j \in M : q_{j-1}^* = 1\}$. Recall that $S^+ = \{(k-1, k)\}$ and let us represent the inventory arc $(k-1, k)$ with index s. From the definition of the function v with objective $i_{k-1} + \sum_{j \in L^+} y_j - i_\ell - r_{k-1}$, we know that

$$i_{k-1}^* + \sum_{j \in L^+} y_j^* - i_\ell^* - r_{k-1}^* \leq v(\{s\} \cup \hat{L}^+ | q_{j-1} = 0, \forall j \in M \setminus \hat{M},\ z_j = 0, \forall j \in \hat{H}).$$

Let

$$R := u_{k-1} + \sum_{j \in \hat{L}^+} \rho_j + \sum_{t \in \hat{M}} \delta_t - \sum_{t \in \hat{H}} \gamma_t$$

be the right hand side of inequality (4.32) at this solution. In this proof, we show that

$$R \geq v(\{s\} \cup \hat{L}^+ | q_{j-1} = 0, \forall j \in M \setminus \hat{M},\ z_j = 0, \forall j \in \hat{H}).$$

The first part of the proof follows similar to that of Proposition E.3. For notational convenience let

$$\rho_s(L(t) | z_t = 0, q_{j-1} = 0, \forall j \in M) = \rho_s(L(t) | z_t = 0, M)$$

and

$$v(S^+|z_t = 0, q_{j-1} = 0, \forall j \in M) = v(S^+|z_t = 0, M).$$

Let $h_{\max} = \max\{j \in \hat{H}\}$. Then, using submodularity and the path structure of LSBIB, we observe the following

$$\begin{aligned}
v(\{s\}) - \rho_s(L(h_{\max})|z_{h_{\max}+1} = 0, M) + \rho_s(L(h_{\max})|z_{h_{\max}} = 0, M)& \\
+ \sum_{j \in \hat{L}(h_{\max})} \rho_j(\{s\}|M)& \quad (4.34)\\
\geq v(\{s\} \cup \hat{L}(h_{\max})|M) - \rho_s(L(h_{\max})|z_{h_{\max}+1} = 0, M)& \\
+ \rho_s(L(h_{\max})|z_{h_{\max}} = 0, M)& \quad (4.35)\\
\geq v(\{s\} \cup \hat{L}(h_{\max})|M) - \rho_s(\hat{L}(h_{\max})|z_{h_{\max}+1} = 0, M)& \\
+ \rho_s(L(h_{\max})|z_{h_{\max}} = 0, M)& \quad (4.36)\\
\geq v(\{s\} \cup \hat{L}(h_{\max})|M) - \rho_s(\hat{L}(h_{\max})|M) + \rho_s(L(h_{\max})|z_{h_{\max}} = 0, M)& \quad (4.37)\\
= v(\hat{L}(h_{\max})|M) + \rho_s(L(h_{\max})|z_{h_{\max}} = 0, M)& \quad (4.38)\\
= v(\{s\} \cup \hat{L}(h_{\max})|z_{h_{\max}} = 0, M).& \quad (4.39)
\end{aligned}$$

Inequality (4.35) is obtained using the submodularity of the function v and the fact that

$$v(\{s\}) = v(\{s\}|M) = u_{k-1}.$$

Inequality (4.37) holds since for any $h_j \in H$ and $h_i \geq h_j$,

$$\rho_s(L(h_j)|z_{h_i} = 0, M) \leq \rho_s(L(h_j)|M)$$

due to the path structure of LSBIB. Similarly, we reach the equality of (4.37) and (4.38) since $v(\hat{L}(h_{\max})|M) = v(\hat{L}(h_{\max})|z_{h_{\max}} = 0, M)$. Using the inequality (4.34)–(4.39), we observe that

$$\begin{aligned}
R - \sum_{t \in \hat{M}} \delta_t \geq & v(\{s\} \cup \hat{L}(h_{\max})|z_{h_{\max}} = 0, M) \\
& - \sum_{i \in \hat{T} \setminus \{h_{\max}\}} \big(\rho_s(L(h_i)|z_{h_{i+1}} = 0, M) - \rho_s(L(h_i)|z_{h_i} = 0, M)\big) \\
& + \sum_{i \in \hat{L}^+ \setminus \hat{L}(h_{\max})} \rho_i(\{s\}|M).
\end{aligned}$$

For the rest of the proof, we use an induction approach. First, we introduce some notation for simplification: let $\hat{H}(h_j) = \hat{H} \cap \{h_j, \ldots, h_\tau\}$ and let $v(C|\hat{H}(h_j), M) := v(C|z_i = 0, \forall i \in \hat{H}(h_j), q_{i-1} = 0, \forall i \in M)$ and $\rho_l(C|\hat{H}(h_j), M) := \rho_l(C|z_i = 0, \forall i \in \hat{H}(h_j), q_{i-1} = 0, \forall i \in M)$.

Assuming that the following is true

$$\begin{aligned} R - \sum_{t \in \hat{M}} \delta_t \geq & v(\{s\} \cup \hat{L}(h_{j+1}) | \hat{H}(h_{j+1}), M) \\ & - \sum_{i \in \hat{H} \setminus [h_{j+1}, h_\tau]} (\rho_s(L(h_i) | z_{h_{i+1}} = 0, M) - \rho_s(L(h_i) | z_{h_i} = 0, M)) \\ & + \sum_{i \in \hat{L}^+ \setminus \hat{L}(h_{j+1})} \rho_i(\{s\} | M) \end{aligned}$$

for iteration $j+1$, we extend the result for h_j. First, we make the following observation

$$\rho_i(\{s\}|M) \geq \rho_i(\{s\}|z_{h_{j+1}} = 0, M) = \rho_i(\{s\}|z_{h_i} = 0, \forall i \in [j+1, \tau], M)$$

for any $i \in L^+ \setminus L(h_{j+1})$. Then,

$$\begin{aligned} v(\{s\} \cup \hat{L}(h_{j+1}) | \hat{H}(h_{j+1}), M) - \rho_s(L(h_j) | h_{j+1} = 0) + \rho_s(L(h_j) | z_{h_j} = 0, M) & \\ + \sum_{i \in \hat{L}(h_j) \setminus \hat{L}(h_{j+1})} \rho_i(\{s\} | M) & \quad (4.40) \\ \geq v(\{s\} \cup \hat{L}(h_j) | \hat{H}(h_{j+1}), M) - \rho_s(L(h_j) | h_{j+1} = 0, M) & \\ + \rho_s(L(h_j) | z_{h_j} = 0, M) & \quad (4.41) \\ \geq v(\hat{L}(h_j) | \hat{H}(h_{j+1}), M) + \rho_s(L(h_j) | z_{h_j} = 0, M) & \quad (4.42) \\ = v(\{s\} \cup \hat{L}(h_j) | \hat{H}(h_j), M). & \quad (4.43) \end{aligned}$$

Inequalities (4.41)–(4.42) are obtained from the observation above and submodularity of the function v and the equality of (4.42) to (4.43) is due to the path structure of LSBIB. As a result of the induction, we have proved that

$$R \geq v(\{s\} \cup \hat{L} | \hat{H}, M) + \sum_{t \in \hat{M}} \delta_t.$$

In the second step of the proof, we show that

$$v(\{s\} \cup \hat{L} | \hat{H}, M) + \sum_{t \in \hat{M}} \delta_t \geq v(\{s\} \cup \hat{L} | \hat{H}, M \setminus \hat{M}). \quad (4.44)$$

In the proof of Proposition 4.12, we discuss that δ_j only depends on the backlog arcs $\{(k+1, k), \ldots, (j, j-1)\}$ and not on the inventory arcs. Consequently, we observe that

$$\begin{aligned} \delta_{m_j} &= v(\{m_j\} | q_{m_i - 1} = 0, \forall i \in [1, j-1]) - v(\{m_j\} | q_{m_i - 1} = 0, \forall i \in [1, j]) \\ &= v(\{m_j\} | q_{m_i - 1} = 0, \forall i \in [1, j-1], \hat{H}) - v(\{m_j\} | q_{m_i - 1} = 0, \forall i \in [1, j], \hat{H}). \end{aligned}$$

Then, showing the relationship in (4.44) is a repetition of the proof of Proposition 4.12 for a lot-sizing network where the upper-bounds of inventory arcs in $\hat{H}$ are all zero. As a result, we conclude that

$$i^*_{k-1} + \sum_{j \in L^+} y_j - i_\ell - r_{k-1} \leq v(\{s\} \cup \hat{L}^+ | q_{j-1} = 0, \forall j \in M \setminus \hat{M},\ z_j = 0, \forall j \in \hat{H}) \leq R$$

and inequality (4.32) is valid for LSBIB. □

Proposition 4.14. Inequality (4.32) can be strengthened by partitioning the set M into two subsets M_1 and M_2 where $M_1 = \{t \in M : \delta_t < b_{t-1}\}$ and $M_2 = M \setminus M_1$. Then,

$$i_{k-1} + \sum_{j \in L^+} y_j + \sum_{j \in H} \gamma_j (1 - z_j) \leq u_{k-1} + \sum_{j \in L^+} \rho_j x_j + i_\ell + r_{k-1} + \sum_{j \in M_1} \delta_j q_{j-1} + \sum_{j \in M_2} r_{j-1}, \tag{4.45}$$

where ρ_j for $j \in L^+$, δ_j for $j \in M$ and γ_j for $j \in H$ are defined in (4.26), (4.27) and (4.33), respectively, is also valid for LSBIB.

Proof. We use a simple reformulation argument to show validity of inequality (4.45). Suppose we break a backlog arc $(t, t-1)$ into two clone arcs: one arc incoming to node $t-1$ and one arc outgoing from node t. Then, we force the flows on these clone arcs to be equal to each other by adding non-anticipativity constraints. Now, consider the submodular function v where the objective function coefficients of the outgoing backlog clones are -1 while the coefficients of the incoming backlog clones are 0. To obtain inequality (4.45), the backlog arcs $(j, j-1)$ for all $j \in M_2$ are cloned. Since the coefficients of x_t for $t \in L^+$ assume that the backlog arcs $(j, j-1)$ for all $j \in M$ are closed, the coefficients for $t \in L^+$ as inequalities (4.25) and (4.32).

Let $(x^*, y^*, i^*, r^*, q^*, z^*)$ be a feasible solution to LSBIB and let $\hat{L}^+ := \{j \in L^+ : x^*_j = 1\}$, $\hat{H} := \{j \in H : z^*_j = 0\}$ and $\hat{M} := \{j \in M : q^*_{j-1} = 1\}$. Then, using the definition of v with the objective function $i_{k-1} + \sum_{j \in L^+} y_j - \sum_{j \in M_2} r_{j-1} - i_\ell - r_{k-1}$, we observe that

$$\begin{aligned} & i^*_{k-1} + \sum_{j \in L^+} y^*_j - i^*_\ell - r^*_{k-1} - \sum_{j \in M_2} r^*_{j-1} \\ & \qquad \leq v(\{s\} \cup \hat{L}^+ | q_{j-1} = 0, \forall j \in M_1 \setminus \hat{M},\ z_j = 0, \forall j \in \hat{H}). \end{aligned}$$

As a result, it suffices to show that

$$\begin{aligned} & u_{k-1} + \sum_{j \in \hat{L}} \rho_j + \sum_{j \in \hat{M} \cap M_1} \delta_j - \sum_{j \in \hat{H}} \gamma_j \\ & \qquad \geq v(\{s\} \cup \hat{L}^+ | q_{j-1} = 0, \forall j \in M_1 \setminus \hat{M},\ z_j = 0, \forall j \in \hat{H}) \end{aligned}$$

to prove the validity of inequality (4.45). We omit the remainder of the proof since it follows very similarly to the proof of Proposition 4.13. □

Proposition 4.15. The inequality

$$i_{k-1} + \sum_{j\in L^+} y_j + \sum_{j\in H} \gamma_j(1-z_j) \leq u_{k-1} + \sum_{j\in L^+} \rho_j x_j + \sum_{j\in M} \delta_j q_{j-1} + r_{k-1} \tag{4.46}$$

where δ_j for $j \in M$ and γ_j for $j \in H$ are defined as in (4.27) and (4.33), respectively and

$$\rho_j = \begin{cases} \min\{d_{kj} + u_j - u_{k-1}, u_j + d_j\} \text{ if } j \in M, \\ \min\{d_{kj} + u_j - u_{k-1}, u_j + d_j\} + \min\{(d_{k,j-1} - u_{k-1})^+, d_{m_i,j-1}\} \quad \text{if} \\ \qquad m_i < j < m_{i+1}, i = 0, \dots, |M| \end{cases}$$

is valid for LSBIB.

Proof. The proof follows very similar to that of Proposition 4.32 with the exception that the function v is computed using the objective function $i_{k-1} + \sum_{j\in L^+} y_j - r_{k-1}$. □

Let $\bar{d}_{ij} = d_{ij} + b_{k-1}$ if $i = k$ and d_{ij} otherwise.

Proposition 4.16. The inequality

$$i_{k-1} + \sum_{j\in L^+} y_j + \sum_{j\in H} \gamma_j(1-z_j) \leq u_{k-1} + \sum_{j\in L^+} \rho_j x_j + \sum_{j\in M} \bar{\delta}_j q_{j-1} \tag{4.47}$$

where γ_j for $j \in H$ is defined as in (4.33),

$$\rho_j = \begin{cases} \min\{b_{k-1} + d_{kj} + u_j - u_{k-1}, u_j + d_j\} \text{ if } j \in M, \\ \min\{b_{k-1} + d_{kj} + u_j - u_{k-1}, u_j + d_j\} \\ \quad + \min\{(b_{k-1} + d_{k,j-1} - u_{k-1})^+, \bar{d}_{m_i,j-1}\} \quad \text{if } m_i < j < m_{i+1},\ i = 0, \dots, |M| \end{cases}$$

and

$$\bar{\delta}_{m_i} = \min\{b_{m_i-1}, \bar{d}_{m_{i-1},m_i-1}\}, \quad \forall\ m_i \in M \tag{4.48}$$

is valid for LSBIB.

Proof. The proof follows very similar to that of Proposition 4.32 with the exception that the function v is computed using the objective function $i_{k-1} + \sum_{j\in L^+} y_j$. □

Remark 4.5. If $M = L^+$ and the backlog arc capacities are zero, then inequality (4.47) reduces to inequality (19) of Atamtürk and Küçükyavuz (2005).

Proposition 4.17. The inequality

$$i_{k-1} + \sum_{j \in L^+} y_j + \sum_{j \in H} \gamma_j (1 - z_j) \leq u_{k-1} + \sum_{j \in L^+} \rho_j x_j + \sum_{j \in M} \bar{\delta}_j q_{j-1} + i_\ell, \tag{4.49}$$

where γ_j for $j \in H$ and $\bar{\delta}_j$ for $j \in M$ are defined as in (4.33) and (4.48), respectively and

$$\rho_j = \begin{cases} \min\{b_{k-1} + d_{kj} + u_j - u_{k-1}, b_{k-1} + d_{k\ell} - u_{k-1}, u_j + d_j, d_{j\ell}\} \text{ if } j \in M, \\ \min\{b_{k-1} + d_{kj} + u_j - u_{k-1}, b_{k-1} + d_{k\ell} - u_{k-1}, u_j + d_j, d_{j\ell}\} \\ \quad + \min\{(b_{k-1} + d_{k,j-1} - u_{k-1})^+, \bar{d}_{m_i,j-1}\} \quad \text{if} \quad m_i < j < m_{i+1},\ i = 0, \ldots, |M| \end{cases}$$

is valid for LSBIB.

Proof. The proof follows very similar to that of Proposition 4.32 with the exception that the function v is computed using the objective function $i_{k-1} + \sum_{j \in L^+} y_j - i_\ell$. □

Remark 4.6. If $M = L^+$ and the backlog arc capacities are zero, then inequality (4.49) reduces to inequality (9) of Atamtürk and Küçükyavuz (2005).

Remark 4.7. The procedure described in Proposition 4.14 can be applied to inequalities (4.46), (4.47), (4.49) by partitioning the set M into two subsets:

$$M_1 = \{t \in M : \delta_t < b_{t-1}\}$$

and $M_2 = M \setminus M_1$. If $j \in M_1$, then the backlog arc $(j, j-1)$ appears in the right hand side of these inequalities as $\delta_j q_{j-1}$ and as r_{j-1}, otherwise. Notice that, any partition of M into M_1 and M_2 would conserve the feasibility.

The partition M_1 and M_2 that maximizes the violation of these inequalities by a given point $(\bar{x}, \bar{y}, \bar{i}, \bar{r}, \bar{q}, \bar{z}) \in \mathbb{R}^{6n}$ can be found in linear time. Simply, add a backlog arc $(j, j-1)$ to M_2 if and only if

$$\bar{r}_{j-1} < \delta_j \bar{x}_{j-1}.$$

Let $\hat{d}_i = (d_{i\ell} - b_\ell)^+$ for $i \geq m_{|M|}$ and d_i otherwise and let $\hat{d}_{ij} = \sum_{t=i}^{j} \hat{d}_t$ if $i \leq j$ and 0 otherwise. Moreover, let $d'_{ij} = \hat{d}_{ij} + b_{k-1}$ if $i = k$ and $\hat{d}_{ij}$ otherwise,

Proposition 4.18. The inequality

$$\begin{aligned} i_{k-1} + r_\ell + \sum_{j \in L^+} y_j + \sum_{j \in H} \gamma_j (1 - z_j) \\ \leq \min\{u_{k-1} + b_\ell, b_{k-1} + d_{k\ell} + u_\ell\} + \sum_{j \in L^+} \rho_j x_j + \sum_{j \in M} \hat{\delta}_j q_{j-1}, \end{aligned} \tag{4.50}$$

where

$$\gamma_{h_i} = \begin{cases} \min\{(u_{k-1} - \hat{d}_{kh_i})^+, \hat{d}_{h_i+1,h_{i+1}}\} & \text{if } m(h_i) = m(h_{(i+1)}), \\ \min\{(u_{k-1} - \hat{d}_{kh_i})^+, \hat{d}_{h_i+1,m(h_i)-1}\} & \text{if } m(h_i) < m(h_{(i+1)}) \end{cases} \quad (4.51)$$

$$\rho_j = \begin{cases} \min\{b_{k-1} + \hat{d}_{kj} + u_j - u_{k-1}, u_j + \hat{d}_j\} \text{ if } j \in M, \\ \min\{b_{k-1} + \hat{d}_{kj} + u_j - u_{k-1}, u_j + \hat{d}_j\} + \min\{(b_{k-1} + \hat{d}_{k,j-1} - u_{k-1})^+, d'_{m_i,j-1}\} \quad \text{if} \\ \qquad m_i < j < m_{i+1}, i = 0, \ldots, |M| \end{cases}$$

and

$$\hat{\delta}_{m_i} = \min\{b_{m_i-1}, d'_{m_{i-1},m_i-1}\}, \quad \forall\, m_i \in M \quad (4.52)$$

is valid for LSBIB.

Proof. In this inequality, the cover set is $S^+ = \{(k-1, k), (\ell+1, \ell)\}$ and for convenience, let the indices s_1 and s_2 represent arcs $(k-1, k)$ and $(\ell+1, \ell)$, respectively. The objective function used to compute v is $i_{k-1} + r_\ell + \sum_{j \in L^+} y_j$. The coefficients ρ_j, γ_j and $\hat{\delta}_j$ are derived from the submodular function v where

$$\rho_j = \rho_j(S^+ | q_{i-1} = 0, \forall i \in M), \quad (4.53)$$

$$\begin{aligned} \gamma_{h_i} = {} & \rho_{s_1}(L(h_i) \cup \{s_2\} | z_{h_j} = 0, \forall j \in [i+1, \tau], q_{j-1} = 0, \forall j \in M) \\ & - \rho_{s_1}(L(h_i) \cup \{s_2\} | z_{h_j} = 0, \forall j \in [i, \tau], q_{j-1} = 0, \forall j \in M) \end{aligned}$$

and

$$\hat{\delta}_{m_i} = v(\{m_j, s_2\} | q_{m_i-1} = 0, \forall i \in [1, j-1]) - v(\{m_j, s_2\} | q_{m_i-1} = 0, \forall i \in [1, j]). \quad (4.54)$$

Let $(x^*, y^*, i^*, r^*, q^*, z^*)$ be a feasible solution to LSBIB. Let $\hat{L}^+ := \{j \in L^+ : x_j^* = 1\}$, $\hat{H} := \{j \in H : z_j^* = 0\}$ and $\hat{M} := \{j \in M : q_{j-1}^* = 1\}$. From the definition of the function v, we know that

$$i_{k-1}^* + r_\ell^* + \sum_{j \in L^+} y_j^* \leq v(\{s_1, s_2\} \cup \hat{L}^+ | q_{j-1} = 0, \forall j \in M \setminus \hat{M},\ z_j = 0, \forall j \in \hat{H}).$$

Let

$$R := \min\{u_{k-1} + b_\ell, b_{k-1} + d_{k\ell} + u_\ell\} + \sum_{j \in \hat{L}^+} \rho_j + \sum_{t \in \hat{M}} \delta_t - \sum_{t \in \hat{H}} \gamma_t$$

be the right hand side of inequality (4.50) at this solution. Moreover, in this inequality

$$v(S^+) = \min\{u_{k-1} + b_\ell, b_{k-1} + d_{k\ell} + u_\ell\}.$$

The rest of the proof is equivalent to the proof of Proposition (4.16), where the upper bound of the backlog arc $(\ell+1, \ell)$ is saturated into the nodes in $[m_{|M|}, \ell]$. In other words, the demand values are taken as $\hat{d}_j$ instead of d_j. □

Proposition 4.19. The inequality

$$i_{k-1} + r_\ell + \sum_{j \in L^+} y_j + \sum_{j \in H} \gamma_j (1 - z_j) \leq \min\{u_{k-1} + b_\ell, d_{k\ell} + u_\ell\} + \sum_{j \in L^+} \rho_j x_j + \sum_{j \in M} \hat{\delta}_j q_{j-1} + r_{k-1}, \quad (4.55)$$

where γ_j for $j \in H$ is defined as in (4.51) and

$$\rho_j = \begin{cases} \min\{\hat{d}_{kj} + u_j - u_{k-1}, u_j + \hat{d}_j\} \text{ if } j \in M, \\ \min\{\hat{d}_{kj} + u_j - u_{k-1}, u_j + \hat{d}_j\} + \min\{(\hat{d}_{k,j-1} - u_{k-1})^+, \hat{d}_{m_i,j-1}\} \quad \text{if} \\ \qquad m_i < j < m_{i+1}, i = 0, \ldots, |M| \end{cases}$$

and

$$\hat{\delta}_{m_i} = \min\{b_{m_i - 1}, \hat{d}_{m_{i-1}, m_i - 1}\}, \quad \forall\, m_i \in M \quad (4.56)$$

is valid for LSBIB.

Proof. The proof follows very similar to that of Proposition 4.18 with the exception that the function v is computed using the objective function $i_{k-1} + r_\ell + \sum_{j \in L^+} y_j - r_{k-1}$. □

Proposition 4.20. The inequality

$$i_{k-1} + r_\ell + \sum_{j \in L^+} y_j + \sum_{j \in H} \gamma_j (1 - z_j) \leq \min\{u_{k-1} + b_\ell, d_{k\ell} + b_{k-1}\} + \sum_{j \in L^+} \rho_j x_j + \sum_{j \in M} \hat{\delta}_j q_{j-1} + i_\ell, \quad (4.57)$$

where γ_j for $j \in H$ and $\hat{\delta}_j$ for $j \in M$ are defined as in (4.51) and (4.56), respectively and

$$\rho_j = \begin{cases} \min\{b_{k-1} + \hat{d}_{kj} + u_j - u_{k-1}, b_{k-1} + \hat{d}_{k\ell} - u_{k-1}, u_j + \hat{d}_j, \hat{d}_{j\ell}\} \text{ if } j \in M, \\ \min\{b_{k-1} + \hat{d}_{kj} + u_j - u_{k-1}, b_{k-1} + \hat{d}_{k\ell} - u_{k-1}, u_j + \hat{d}_j, \hat{d}_{j\ell}\} \\ \quad + \min\{(b_{k-1} + \hat{d}_{k,j-1} - u_{k-1})^+, d'_{m_i,j-1}\} \quad \text{if} \quad m_i < j < m_{i+1}, i = 0, \ldots, |M| \end{cases}$$

is valid for LSBIB.

Proof. The proof follows very similar to that of Proposition 4.18 with the exception that the function v is computed using the objective function $i_{k-1}+r_\ell+\sum_{j\in L^+} y_j - i_\ell$. $\square$

Remark 4.8. The procedure described in Remark 4.7 and Proposition 4.14 can be applied to inequalities (4.50), (4.55), (4.57) by partitioning the set M into two subsets:

$$M_1 = \{t \in M : \bar{\delta}_t < b_{t-1}\}$$

and $M_2 = M \setminus M_1$. If $j \in M_1$, then the backlog arc $(j, j-1)$ appears in the right hand side of these inequalities as $\delta_j q_{j-1}$ and as r_{j-1}, otherwise. Notice that, any partition of M into M_1 and M_2 would conserve the feasibility.

The partition M_1 and M_2 that maximizes the violation of these inequalities by a given point $(\bar{x}, \bar{y}, \bar{i}, \bar{r}, \bar{q}, \bar{z}) \in \mathbb{R}^{6n}$ can be found in linear time. Simply, add a backlog arc $(j, j-1)$ to M_2 if and only if

$$\bar{r}_{j-1} < \bar{\delta}_j \bar{x}_{j-1}.$$

$\square$

Let $T = \{t_1, \ldots, t_{|T|}\}$ be a subset of backlog arc indices in an increasing order. In this convention, $t_i = j$ refers to the backlog arc $(j+1, j)$. For a backlog arc $(j+1, j)$, we define $n(j) = \max\{t \in L^+ \cup \{(k-1)\} : t \leq j\}$. Moreover, let $K = \cup_{i=1}^{r} n(t_i)$. Alternatively, we represent this set as $K = \{k_1, \ldots, k_{|K|}\}$. For convenience, let $k_0 = k$, $k_{|K|+1} = \ell$ and $t_0 = k$, $t_{|T|+1} = \ell$. Finally, let $\bar{L}(t) = L^+ \cap [k, t]$.

Proposition 4.21. The inequality

$$r_\ell + \sum_{t\in L^+} y_t + \sum_{j\in T} \phi_j(1-q_j) \leq b_\ell + \sum_{j\in L^+} \rho_j x_j + \sum_{j\in K} \psi_j z_j + i_\ell + r_{k-1}, \tag{4.58}$$

where

$$\rho_j = \begin{cases} \min\{d_{j\ell} + b_{j-1} - b_\ell, d_{k\ell} - b_\ell, d_j + b_{j-1}, d_{kj}\} \text{ if } j \in K, \\ \min\{d_{j\ell} + b_{j-1} - b_\ell, d_{k\ell} - b_\ell, d_j + b_{j-1}, d_{kj}\} + \min\{(d_{j+1,\ell} - b_\ell)^+, d_{j+1,k_{i+1}}\} \quad \text{if} \\ \qquad k_i < j < k_{i+1}, i = 0, \ldots, |K|, \end{cases}$$

$$\phi_{t_i} = \begin{cases} \min\{(b_\ell - d_{t_i+1,\ell})^+, d_{t_{i-1}+1,t_i}\} & \text{if } n(t_{i-1}) = n(t_i), \\ \min\{(b_\ell - d_{t_i+1,\ell})^+, d_{n(t_i)+1,t_i}\} & \text{if } n(t_{i-1}) < n(t_i) \end{cases} \tag{4.59}$$

and

$$\psi_{k_i} = \min\{u_{k_i}, d_{k_i+1,k_{i+1}}\}, \quad \forall\, k_i \in K$$

is valid for LSBIB.

Proof. In this inequality, the cover set is $S^+ = \{(\ell+1, \ell)\}$ and for convenience, let the index s represent the backlog arc $(\ell+1, \ell)$. In this inequality, the objective function used to compute v is $r_\ell + \sum_{j \in L^+} y_j - i_\ell - r_{k-1}$. The coefficients ρ_j, ϕ_j and ψ_j are derived from the submodular function v where

$$\rho_j = \rho_j(S^+ | z_i = 0, \forall i \in K), \tag{4.60}$$

$$\begin{aligned} \phi_{t_j} = \rho_s(\bar{L}(t_j) \cup \{s\} | q_{t_i} = 0, \forall i \in [1, j-1], z_i = 0, \forall i \in K) \\ - \rho_s(\bar{L}(t_j) | q_{t_i} = 0, \forall i \in [1, j], z_i = 0, \forall i \in K) \end{aligned}$$

and

$$\psi_{k_j} = v(\{k_j\} | z_{k_i} = 0, \forall i \in [j+1, |K|]) - v(\{k_j\} | z_{k_i} = 0, \forall i \in [j, |K|]). \tag{4.61}$$

Inequality (4.58) is equivalent to inequality (4.32) for the mirrored image of the node set $[k, \ell]$. More formally, consider renaming nodes $k, k+1, \ldots, \ell$ as $\bar{\ell}, \bar{\ell}-1, \ldots, \bar{k}$, respectively. Moreover, rename the backlogging variables $r_\ell, r_{\ell-1}, \ldots, r_{k-1}$ as $\bar{i}_{k-1}, \bar{i}_k, \ldots, \bar{i}_\ell$ and rename the inventory variables $i_{k-1}, i_k, \ldots, i_\ell$ as $\bar{r}_\ell, \bar{r}_{\ell-1}, \ldots, \bar{r}_{k-1}$, respectively. Then, inequality (4.32) for the node set $[\bar{k}, \bar{\ell}]$, where $S^+ = \{(\bar{k}-1, \bar{k})\}$ is equivalent to inequality (4.58), where $S^+ = \{(\ell+1, \ell)\}$. Consequently, the proof of this proposition is a symmetric version of the proof of Proposition 4.32. □

Proposition 4.22. The inequality

$$r_\ell + \sum_{t \in L^+} y_t + \sum_{j \in T} \phi_j (1 - q_j) \leq b_\ell + \sum_{j \in L^+} \rho_j x_j + \sum_{j \in K} \psi_j z_j + i_\ell, \tag{4.62}$$

where ϕ_j is defined as in (4.59),

$$\rho_j = \begin{cases} \min\{d_{j\ell} + b_{j-1} - b_\ell, d_j + b_{j-1}\} \text{ if } j \in K, \\ \min\{d_{j\ell} + b_{j-1} - b_\ell, d_j + b_{j-1}\} + \min\{(d_{j+1,\ell} - b_\ell)^+, d_{j+1,k_{i+1}}\} \quad \text{if} \\ \qquad k_i < j < k_{i+1}, i = 0, \ldots, |K|, \end{cases}$$

and

$$\psi_{k_i} = \min\{u_{k_i}, d_{k_i+1,k_{i+1}}\}, \quad \forall\ k_i \in K$$

is valid for LSBIB.

Proof. The proof follows very similar to that of Proposition 4.22 with the exception that the function v is computed using the objective function $r_\ell + \sum_{j\in L^+} y_j - i_\ell$. Moreover, notice that inequality (4.62) is symmetric to inequality (4.46). $\square$

Proposition 4.23. The inequality

$$r_\ell + \sum_{t\in L^+} y_t + \sum_{j\in T} \phi_j(1-q_j) \le b_\ell + \sum_{j\in L^+} \rho_j x_j + \sum_{j\in K} \psi_j z_j, \tag{4.63}$$

where ϕ_j is defined as in (4.59),

$$\rho_j = \begin{cases} \min\{u_\ell + d_{j\ell} + b_{j-1} - b_\ell, d_j + b_{j-1}\} \text{ if } j\in K, \\ \min\{u_\ell + d_{j\ell} + b_{j-1} - b_\ell, d_j + b_{j-1}\} \\ \qquad + \min\{(u_\ell + d_{j+1,\ell} - b_\ell)^+, \bar{d}_{j+1,k_{i+1}}\} \quad \text{if} \quad k_i < j < k_{i+1}, i = 0,\ldots,|K|, \end{cases}$$

and

$$\psi_{k_i} = \min\{u_{k_i}, \bar{d}_{k_i+1,k_{i+1}}\}, \quad \forall\ k_i \in K$$

with $\bar{d}_{ij} = d_{ij} + u_\ell$ if $j=\ell$ and d_{ij} otherwise, is valid for LSBIB.

Proof. The proof follows very similar to that of Proposition 4.22 with the exception that the function v is computed using the objective function $r_\ell + \sum_{j\in L^+} y_j$. Moreover, notice that inequality (4.63) is symmetric to inequality (4.47). $\square$

Proposition 4.24. The inequality

$$r_\ell + \sum_{t\in L^+} y_t + \sum_{j\in T} \phi_j(1-q_j) \le b_\ell + \sum_{j\in L^+} \rho_t x_j + \sum_{j\in K} \psi_j z_j + r_{k-1}, \tag{4.64}$$

where ϕ_j is defined as in (4.59),

$$\rho_j = \begin{cases} \min\{u_\ell + d_{j\ell} + b_{j-1} - b_\ell, u_\ell + d_{k\ell} - b_\ell, d_j + b_{j-1}, d_{kj}\} \text{ if } j\in K, \\ \min\{u_\ell + d_{j\ell} + b_{j-1} - b_\ell, u_\ell + d_{k\ell} - b_\ell, d_j + b_{j-1}, d_{kj}\} \\ \qquad + \min\{(u_\ell + d_{j+1,\ell} - b_\ell)^+, \bar{d}_{j+1,k_{i+1}}\} \quad \text{if} \quad k_i < j < k_{i+1}, i = 0,\ldots,|K|, \end{cases}$$

and

$$\psi_{k_i} = \min\{u_{k_i}, \bar{d}_{k_i+1,k_{i+1}}\}, \quad \forall\ k_i \in K$$

with $\bar{d}_{ij} = d_{ij} + u_\ell$ if $j=\ell$ and d_{ij} otherwise, is valid for LSBIB.

Proof. The proof follows very similar to that of Proposition 4.22 with the exception that the function v is computed using the objective function $r_\ell + \sum_{j\in L^+} y_j - r_{k-1}$. Moreover, notice that inequality (4.64) is symmetric to inequality (4.49). $\square$

Remark 4.9. The procedure described in Remark 4.7 and Proposition 4.14 can be applied to inequalities (4.58), (4.62), (4.63) and (4.64) by partitioning the set K into two subsets:

$$K_1 = \{t \in K : \psi_t < u_t\}$$

and $K_2 = K \setminus K_1$. If $j \in K_1$, then the inventory arc $(j, j+1)$ appears in the right hand side of these inequalities as $\psi_j z_j$ and as i_j, otherwise. Notice that, any partition of K into K_1 and K_2 would conserve the feasibility.

The partition K_1 and K_2 that maximizes the violation of these inequalities by a given point $(\bar{x}, \bar{y}, \bar{i}, \bar{r}, \bar{q}, \bar{z}) \in \mathbb{R}^{6n}$ can be found in linear time. Simply, add an inventory arc $(j, j+1)$ to K_2 if and only if

$$\bar{i}_j < \psi_j \bar{z}_j.$$

□

Let $\tilde{d}_i = (d_{ki} - u_{k-1})^+$ for $i \leq k_1$ and d_i otherwise. Moreover, let $\tilde{d}_{ij} = \sum_{t=i}^{j} \tilde{d}_t$ if $i \leq j$ and 0 otherwise.

Proposition 4.25. The inequality

$$i_{k-1} + r_\ell + \sum_{j \in L^+} y_j + \sum_{j \in T} \phi_j (1 - q_j) \leq \min\{u_{k-1} + b_\ell, b_{k-1} + d_{k\ell} + u_\ell\} + \sum_{j \in L^+} \rho_j x_j + \sum_{j \in K} \psi_j z_j, \quad (4.65)$$

where

$$\rho_j = \begin{cases} \min\{u_\ell + \tilde{d}_{j\ell} + b_{j-1} - b_\ell, \tilde{d}_j + b_{j-1}\} \text{ if } j \in K, \\ \min\{u_\ell + \tilde{d}_{j\ell} + b_{j-1} - b_\ell, \tilde{d}_j + b_{j-1}\} \\ \quad + \min\{(u_\ell + \tilde{d}_{j+1,\ell} - b_\ell)^+, \bar{d}_{j+1,k_{i+1}}\} \quad \text{if} \quad k_i < j < k_{i+1}, i = 0, \ldots, |K|, \end{cases}$$

$$\phi_{t_i} = \begin{cases} \min\{(b_\ell - \tilde{d}_{t_i+1,\ell})^+, \tilde{d}_{t_{i-1}+1,t_i}\} & \text{if } n(t_{i-1}) = n(t_i), \\ \min\{(b_\ell - \tilde{d}_{t_i+1,\ell})^+, \tilde{d}_{n(t_i)+1,t_i}\} & \text{if } n(t_{i-1}) < n(t_i) \end{cases} \quad (4.66)$$

and

$$\psi_{k_i} = \min\{u_{k_i}, \bar{d}_{k_i+1,k_{i+1}}\}, \quad \forall\, k_i \in K$$

with $\bar{d}_{ij} = \tilde{d}_{ij} + u_\ell$ if $j = \ell$ and $\tilde{d}_{ij}$ otherwise, is valid for LSBIB.

Proof. In this inequality, the cover set is $S^+ = \{(k-1, k), (\ell+1, \ell)\}$ and for convenience, let the indices s_1 and s_2 represent arcs $(k-1, k)$ and $(\ell+1, \ell)$, respectively. Here, the objective function used to compute v is $i_{k-1} + r_\ell + \sum_{j \in L^+} y_j$. The coefficients ρ_j, ϕ_j

and ψ_j are derived from the submodular function v where

$$\rho_j = \rho_j(S^+|z_i = 0, \forall i \in K),$$

$$\begin{aligned}\phi_{t_j} = \rho_{s_2}(\bar{L}(t_j) \cup \{s_1\}|q_{t_i} = 0, \forall i \in [1, j-1], z_i = 0, \forall i \in K) \\ - \rho_{s_2}(\bar{L}(t_j) \cup \{s_1\}|q_{t_i} = 0, \forall i \in [1, i], z_i = 0, \forall i \in K)\end{aligned}$$

and

$$\psi_{k_j} = v(\{k_j, s_1\}|z_{k_i} = 0, \forall i \in [j+1, |K|]) - v(\{k_j, s_1\}|z_{k_j} = 0, \forall i \in [j, |K|]).$$

Inequality (4.65) is symmetrical to inequality (4.50). Moreover, the coefficients ϕ_j for $j \in T$, ρ_j for $j \in L^+$ and ψ_j for $j \in K$ are equivalent to the coefficients of inequality (4.63), where the demands at nodes $[k, k_1]$ are saturated with the upper-bound of the inventory arc $(k-1, k)$. Consequently, the proof follows very closely to that of Proposition 4.18. □

Proposition 4.26. The inequality

$$\begin{aligned}i_{k-1} + r_\ell + \sum_{j \in L^+} y_j + \sum_{j \in T} \phi_j(1 - q_j) \\ \leq \min\{u_{k-1} + b_\ell, d_{k\ell} + u_\ell\} + \sum_{j \in L^+} \rho_j x_j + \sum_{j \in K} \psi_j z_j + r_{k-1}, \quad (4.67)\end{aligned}$$

where ϕ_j for $j \in T$ is defined as in (4.66),

$$\rho_j = \begin{cases} \min\{u_\ell + \tilde{d}_{j\ell} + b_{j-1} - b_\ell, u_\ell + \tilde{d}_{k\ell} - b_\ell, \tilde{d}_j + b_{j-1}, \tilde{d}_{kj}\} \text{ if } j \in K, \\ \min\{u_\ell + \tilde{d}_{j\ell} + b_{j-1} - b_\ell, u_\ell + \tilde{d}_{k\ell} - b_\ell, \tilde{d}_j + b_{j-1}, \tilde{d}_{kj}\} \\ \quad + \min\{(u_\ell + \tilde{d}_{j+1,\ell} - b_\ell)^+, \bar{d}_{j+1,k_{i+1}}\} \quad \text{if} \quad k_i < j < k_{i+1}, i = 0, \dots, |K|, \end{cases}$$

and

$$\psi_{k_i} = \min\{u_{k_i}, \bar{d}_{k_i+1,k_{i+1}}\}, \quad \forall\ k_i \in K$$

with $\bar{d}_{ij} = \tilde{d}_{ij} + u_\ell$ if $j = \ell$ and $\tilde{d}_{ij}$ otherwise, is valid for LSBIB.

Proof. The proof follows very similar to that of Proposition 4.25 with the exception that the function v is computed using the objective function $i_{k-1} + r_\ell + \sum_{j \in L^+} y_j - r_{k-1}$. □

Proposition 4.27. The inequality

$$i_{k-1} + r_\ell + \sum_{j \in L^+} y_j + \sum_{j \in T} \phi_j (1 - q_j) \leq \min\{u_{k-1} + b_\ell, d_{k\ell} + b_{k-1}\} + \sum_{j \in L^+} \rho_j x_j + \sum_{j \in K} \psi_j z_j + i_\ell, \quad (4.68)$$

where ϕ_j for $j \in T$ is defined as in (4.66),

$$\rho_j = \begin{cases} \min\{\tilde{d}_{j\ell} + b_{j-1} - b_\ell, \tilde{d}_j + b_{j-1}\} \text{ if } j \in K, \\ \min\{\tilde{d}_{j\ell} + b_{j-1} - b_\ell, \tilde{d}_j + b_{j-1}\} + \min\{(\tilde{d}_{j+1,\ell} - b_\ell)^+, \tilde{d}_{j+1,k_{i+1}}\} \quad \text{if} \\ \qquad k_i < j < k_{i+1}, i = 0, \ldots, |K|, \end{cases}$$

and

$$\psi_{k_i} = \min\{u_{k_i}, \tilde{d}_{k_i+1,k_{i+1}}\}, \quad \forall\ k_i \in K$$

is valid for LSBIB.

Proof. The proof follows very similar to that of Proposition 4.25 with the exception that the function v is computed using the objective function $i_{k-1}+r_\ell+\sum_{j\in L^+} y_j - i_\ell$. □

Remark 4.10. The partition procedure described in Remark 4.9 can be extended to inequalities (4.65), (4.67) and (4.68). □

4.3.1 Strength of lifted inequalities

In this section, we investigate the conditions under which some of these valid inequalities are facet defining for the convex hull of $\mathcal{P}$. Let $[k, \ell] \subseteq [1, n]$, $S^+ \subseteq E^+$, $L^+ \subseteq E^+ \setminus S^+$ and $S^- \subseteq E^-$. For convenience, we introduce some notation in this section. The inventory arc $(k-1, k)$ is presented by index s. Let the elements of the set L^+ be represented by $\{l_1, \ldots, l_L\}$ and let $\varepsilon > 0$ be a sufficiently small value. We also define the following indices:

$$p = \min\{j \in L^+ : u_{k-1} \leq d_{kj}\},$$

$$\hat{t} = \min\{j \in [k, \ell+1] \setminus L^+\}$$

and

$$\tilde{t} = \max\{j \in L^+ : j \leq p\}.$$

Moreover, let $\mathbf{e}_j$, $\mathbf{f}_j$, $\mathbf{g}_j$, $\mathbf{h}_j$, $\mathbf{k}_j$ and $\mathbf{m}_j$ be the unit vectors corresponding to variables y_j, i_j, r_j, x_j, z_j and q_j respectively.

Theorem 4.28. For a node set $[k, \ell]$, let $H \subseteq [k-1, \ell-1]$, $M = L^+$ in Proposition 4.13. Recall that in $S^+ = \{(k-1, k)\}$ and $S^- = \emptyset$ in inequality (4.32). If $d_j > 0$

for all $j \in [k,\ell]$, then the conditions below are sufficient for inequality (4.32) to be facet-defining for conv($\mathcal{P}$):

(i) $(k-1) \cup \{[\ k, p-1\ ] \cap L^+\} \subseteq H$;

(ii) for all $j \in L^+$, $\rho_j(\{s\}) > 0$;

(iii) $u_{\tilde{t}} > d_{\tilde{t}+1,\ell}$;

(iv) $u_{l_1} \geq d_{l_1+1,\ell}$;

(v) if $u_{k-1} = d_{k,p-1}$, then $l_1 < p$, otherwise $l_1 \leq p$;

(vi) $\hat{t} \in [k,p]$, $u_{\hat{t}} \geq d_{\hat{t}+1,\ell}$ and $b_{\hat{t}-1} \geq d_{k\hat{t}-1}$;

(vii) if $\ell < n$, then $u_{l_L} \geq d_{l_L+1,\ell+1}$;

(viii) for all $l_j \in L^+$ $b_{l_j-1} > d_{l_{j-1},l_j-1}$.

Proof. We provide $5n-2$ affinely independent points that lie on the face:

$$F = \big\{(\mathbf{y}, \mathbf{i}, \mathbf{r}, \mathbf{x}, \mathbf{z}, \mathbf{q}) \in \mathcal{P}\ : \\ i_{k-1} + \sum_{t \in L^+} (y_t - \rho_j x_j) + \sum_{t \in H} \gamma_j (1 - z_j) - i_\ell - r_{k-1} - \sum_{j \in L^+} \delta_j q_{j-1} = u_{k-1}\big\}.$$

Consider the following base values for variables $(\mathbf{y}, \mathbf{i}, \mathbf{r}, \mathbf{x}, \mathbf{z}, \mathbf{q})$ of LSBIB described below:

$$i_j = \begin{cases} 0,\ j \in [1,k-2] \cup [\ell,n], \\ u_{k-1},\ j = k-1, \\ (u_{k-1} - d_{kj})^+,\ j \in [k,l-1], \end{cases} \qquad y_j = \begin{cases} d_j,\ j \in [1,k-2] \cup [\ell+1,n], \\ d_{k-1} + u_{k-1},\ j = k-1, \\ 0,\ j \in [k,\ell], \end{cases}$$

$$z_j = \begin{cases} 0,\ j \in [1,k-2] \cup [\ell,n], \\ 1,\ j \in [k-1,\ell-1], \end{cases} \qquad x_j = \begin{cases} 1,\ j \in [1,k-1] \cup [\ell+1,n], \\ 0,\ j \in [k,\ell], \end{cases}$$

$$r_j = 0,\ j \in [1,n],\ q_j^* = 0,\ j \in [1,n].$$

Let w_0 be the vector representation of the solution above:

$$w_0 = \sum_{j \in [1,k-2]} (d_j \mathbf{e}_j + \mathbf{h}_j) + \sum_{j \in [\ell+1,n]} (d_j \mathbf{e}_j + \mathbf{h}_j) + (d_{k-1} + u_{k-1})\mathbf{e}_{k-1} + \mathbf{h}_{k-1} + \\ \sum_{j \in [k,\ell-1]} \mathbf{k}_j + \sum_{j \in [k-1,p-1]} (u_{k-1} - d_{kj})\mathbf{f}_j.$$

Since $u_{k-1} < d_{k\ell}$, w_0 is not feasible for LSBIB. In the remainder of the proof we perturb w_0 in various ways to obtain $5n-2$ affinely independent points.

First, we give $2|L^+|$ feasible points represented by w_j and $\bar{w}_j$ for $j \in L^+$.

(i) For $j \in L^+$: If $j \leq p$, then

$$w_j = w_0 + \mathbf{h}_j + (d_{k\ell} - u_{k-1})\mathbf{e}_j + \sum_{t\in[j,p-1]} (d_{t+1,\ell} - (u_{k-1} - d_{kt}))\mathbf{f}_t + \sum_{t\in[p,\ell]} d_{t+1,\ell}\mathbf{f}_t;$$

otherwise let

$$\begin{aligned} w_j = w_0 + \mathbf{h}_j + \mathbf{h}_{\hat{t}} + d_{j\ell}\mathbf{e}_j + (d_{k,j-1} - u_{k-1})\mathbf{e}_{\hat{t}} \\ + \sum_{t\in[\hat{t},p-1]} (d_{t+1,j-1} - (u_{k-1-d_{kt}}))\mathbf{f}_t + \sum_{t\in[p,j-1]} d_{t+1,j-1}\mathbf{f}_t + \sum_{t\in[j,\ell]} d_{t+1,\ell}\mathbf{f}_t, \end{aligned}$$

where $\hat{t}$ is defined as in condition (vi).

(ii) For $j \in L^+$:

$$\bar{w}_j = \begin{cases} w_j + \varepsilon\mathbf{e}_j + \varepsilon\sum_{t\in[j,\ell]}\mathbf{f}_t - \varepsilon\mathbf{e}_{\ell+1} + \mathbf{k}_\ell, & \text{if } j \geq \tilde{t}, \\ w_j + \varepsilon\mathbf{e}_j - \varepsilon\mathbf{e}_{k-1} - \varepsilon\sum_{t\in[k-1,j-1]}\mathbf{f}_t, & \text{otherwise.} \end{cases}$$

Next, we give $2 \times |[k,\ell] \setminus L^+|$ affinely independent feasible points w_j and $\bar{w}_j$ corresponding to $j \in [k,\ell] \setminus L^+$ that are on the face F.

(iii) $w_j = w_{l_1} + \mathbf{h}_j$ and

(iv) First, recall that $\hat{t} \leq p$ from condition (vi). Then, the next set of solutions corresponding to $j \in [k,\ell] \setminus L^+$ can be summarized by:

$$\bar{w}_j = \begin{cases} w_0 + \mathbf{h}_j + (d_{k\ell} - u_{k-1})\mathbf{e}_j + \sum_{t\in[j,p-1]}(d_{t+1,\ell} - (u_{k-1} - d_{kt}))\mathbf{f}_t + \\ \qquad \sum_{t\in[p,\ell]} d_{t+1,\ell}\mathbf{f}_t \quad \text{if} \quad j \leq p, \\ \bar{w}_{\hat{t}} + \mathbf{h}_j - \varepsilon\mathbf{e}_{\hat{t}} + \varepsilon\mathbf{e}_j - \varepsilon\sum_{t\in[\hat{t},j-1]}\mathbf{f}_t \quad \text{if} \quad j > p. \end{cases}$$

Next, we give $2|[1,k-1]|$ points corresponding to production arcs $j \in [1,k-1]$.

(v) If $j > 1$, then $w_j = w_{l_1} - \mathbf{h}_j - d_j\mathbf{e}_j + d_j\mathbf{e}_{j-1} + d_j\mathbf{f}_{j-1} + \mathbf{k}_{j-1}$ and if $j = 1$, then $w_j = w_{l_1} - \mathbf{h}_j - d_j\mathbf{e}_j + d_j\mathbf{e}_{j+1} + d_j\mathbf{g}_j + \mathbf{m}_j$.

(vi) If $j < k-1$, then $\bar{w}_j = w_{l_1} + \varepsilon\mathbf{e}_j - \varepsilon\mathbf{e}_{j+1} + \mathbf{k}_j + \varepsilon\mathbf{f}_j$ and if $j = k-1$ and $k > 2$, then $\bar{w}_j = w_{l_1} + \varepsilon\mathbf{e}_j - \varepsilon\mathbf{e}_{j-1} + \mathbf{m}_{j-1} + \varepsilon\mathbf{g}_{j-1}$. Finally, if $k = 2$, then $\bar{w}_{k-1} = w_{l_1} - \varepsilon\mathbf{e}_{k-1} + \varepsilon\mathbf{e}_{l_1} - \varepsilon\sum_{t\in[k-1,l_1-1]}\mathbf{k}_t$. Note that, if $k = 1$, then this set of points are irrelevant.

Now, we give $2|[\ell+1,n]|$ points corresponding to production arcs $j \in [\ell+1,n]$.

(vii) if $j = \ell+1$, then $w_j = w_{l_L} - \mathbf{h}_j - d_j\mathbf{e}_j + d_j\mathbf{e}_{l_L} + d_j\sum_{t\in[l_L,\ell]}\mathbf{f}_t + \mathbf{k}_\ell$ and otherwise $w_j = w_{l_L} - \mathbf{h}_j - d_j\mathbf{e}_j + d_j\mathbf{e}_{j-1} + d_j\mathbf{f}_{j-1} + \mathbf{k}_{j-1}$.

(viii) if $j=\ell+1$, then $\bar{w}_j = w_{l_L} + \varepsilon \mathbf{e}_{l_L} - \varepsilon \mathbf{e}_j + \varepsilon \sum_{t\in[l_L,\ell]} \mathbf{f}_t + \mathbf{k}_\ell$ and if $j > \ell+1$, then $\bar{w}_j = w_{l_L} + \varepsilon \mathbf{e}_j - \varepsilon \mathbf{e}_{j+1} + \varepsilon \mathbf{f}_j + \varepsilon \mathbf{k}_j$.

Next, we give $3|[1,k-2]| + 3|[\ell,n-1]|$ points corresponding to inventory arcs $(j,j+1)$ and backlog arcs $(j+1,j)$ for $j \in [1,k-2] \cup [\ell,n-1]$ represented by $\tilde{w}_j$, $\hat{w}_j$ and $\breve{w}_j$. Then, for $j \in [1,k-2] \cup [\ell,n-1]$, let

(ix) $\tilde{w}_j = w_{l_1} + \mathbf{k}_j$,

(x) $\hat{w}_j = w_{l_1} + \mathbf{m}_j$ and

(xi) if $j \neq \ell$, then let $\breve{w}_j = w_{l_1} + \mathbf{k}_j + \mathbf{m}_j + \varepsilon \mathbf{f}_j + \varepsilon \mathbf{g}_j$, otherwise let

$$\breve{w}_j = \bar{w}_{\hat{t}} - \sum_{t\in[\hat{t},\ell-1]} \varepsilon \mathbf{f}_t - \varepsilon \mathbf{e}_{\hat{t}} + \varepsilon \mathbf{g}_\ell + \mathbf{m}_\ell.$$

Next, we give points on F that correspond to the inventory arcs in $[k-1,\ell]$. We represent these points by $\tilde{w}_j$ for $(j,j+1)$, for $j \in [k-1,\ell-1]$. Let $H = \{h_1,\ldots,h_{|H|}\} \subseteq [k-1,p-1]$. Without loss of generality, we assume $h_{|H|} \leq p-1$ since $\gamma_{h_j} = 0$ if $h_j \geq p$.

(xi) If $j \in [p,\ell-1]$, then let

$$\tilde{w}_j = \sum_{t\in[1,k-2]\cup[\ell+1,n]} (d_t \mathbf{e}_t + \mathbf{h}_t) + (d_{kj} - u_{k-1})\,\mathbf{e}_{\hat{t}} + \mathbf{h}_{\hat{t}} + \sum_{t\in[k-1,\ell-1]\setminus\{j\}} \mathbf{k}_t + (d_{k-1} + u_{k-1})\mathbf{e}_{k-1} + \mathbf{h}_{k-1} + d_{j+1,\ell}\mathbf{e}_{j+1} + \sum_{t\in[k,\hat{t}-1]} (u_{k-1} - d_{kt})\mathbf{f}_t + \sum_{t\in[\hat{t},j-1]} d_{t,j-1}\mathbf{f}_t + \sum_{t\in[j,\ell-1]} d_{t+1,\ell}\mathbf{f}_t$$

Recall that $m(j) = \min\{t \in L^+ \ : \ t \geq j\}$. If $j \in [k-1,p-1]$, $j \in H$ and $m(j) \leq \ell$, then

$$\tilde{w}_j = \sum_{t\in[1,k-2]\cup[\ell+1,n]} (d_t \mathbf{e}_t + \mathbf{h}_t) + d_{k-1,j}\mathbf{e}_{k-1} + \mathbf{h}_{k-1} + \sum_{t\in[k-1,j-1]} (d_{t+1,j}\mathbf{f}_t + \mathbf{k}_t) + \sum_{t\in[j,m(j)-1]} (d_t \mathbf{e}_t + \mathbf{h}_t) + \mathbf{h}_{m(j)} + d_{m(j),\ell}\mathbf{e}_{m(j)} + \sum_{t\in[j+1,\ell-1]} \mathbf{k}_t + \sum_{t\in[m(j),\ell-1]} d_{t+1,\ell}\mathbf{f}_t.$$

If $j \in [k-1,p-1]$, $j \in H$ and $m(j) = \ell+1$, then

$$\tilde{w}_j = \sum_{t\in[1,k-2]\cup[\ell+1,n]} (d_t \mathbf{e}_t + \mathbf{h}_t) + d_{k-1,j}\mathbf{e}_{k-1} + \mathbf{h}_{k-1} + \sum_{t\in[k-1,j-1]} (d_{t+1,j}\mathbf{f}_t + \mathbf{k}_t) + \sum_{t\in[j,\ell]} (d_t \mathbf{e}_t + \mathbf{h}_t) + \sum_{t\in[j+1,\ell-1]} \mathbf{k}_t.$$

Finally, consider the case where $j \in [k-1, p-1]$ and $j \notin H$ and let $h_L = \max\{t \in H : t < j\}$. If, $j \in [h_L+1, m(h_L)-1]$, then, let

$$\tilde{w}_j = \tilde{w}_{h_L} - \mathbf{k}_j.$$

Note that, $j \in [h_L+1, m(h_L)-1]$ always holds, since $m(h_L) \in H$ if $m(h_L) \leq p-1$ due to condition (i). Consequently, if $j \notin [h_L+1, m(h_L)-1]$, then we would reach a contradiction with the definition of h_L.

Next, we give points on F that correspond to the backlog arcs in $[k, \ell]$. We represent these points by $\hat{w}_j$ for arcs $(j+1, j)$, where $j \in [k-1, \ell-1]$.

(xii) If $j+1 \notin L^+$, then consider the points

$$\hat{w}_j = w_{l_1} + \mathbf{m}_j.$$

For $j \in L^+$, we use indices l_j. Let $l_0 := k-1$ and $j = 1, \ldots, |L^+|$. If $l_{j-1} \in [p, \ell-1]$, then let

$$\hat{w}_{l_j} = \sum_{t \in [1,k-2] \cup [\ell+1,n]} (d_t \mathbf{e}_t + \mathbf{h}_t) + \left(d_{k,l_{j-1}} - u_{k-1}\right) \mathbf{e}_{\hat{t}} + \mathbf{h}_{\hat{t}} + \sum_{t \in [k-1,\ell-1] \setminus \{l_{j-1}\}} \mathbf{k}_t + (d_{k-1} + u_{k-1}) \mathbf{e}_{k-1} + \mathbf{h}_{k-1} + d_{l_{j-1}+1,\ell} \mathbf{e}_{l_j} + \sum_{t \in [k,\hat{t}-1]} (u_{k-1} - d_{kt}) \mathbf{f}_t + \sum_{t \in [\hat{t}, l_{j-1}-1]} d_{t,j-1} \mathbf{f}_t + \sum_{t \in [l_{j-1}+1, l_j-1]} \left(d_{t,l_j-1} \mathbf{g}_t + \mathbf{m}_t\right) + \sum_{t \in [l_j, \ell-1]} d_{t+1,\ell} \mathbf{f}_t$$

If $l_{j-1} \in [k-1, p-1]$, then

$$\hat{w}_{l_j} = \sum_{t \in [1,k-2] \cup [\ell+1,n]} (d_t \mathbf{e}_t + \mathbf{h}_t) + d_{k-1,l_{j-1}} \mathbf{e}_{k-1} + \mathbf{h}_{l_j} + d_{l_{j-1}+1,\ell} \mathbf{e}_{l_j} + \sum_{t \in [k-1,\ell-1] \setminus \{l_{j-1}\}} \mathbf{k}_t + \sum_{t \in [k-1, l_{j-1}-1]} d_{t+1,l_{j-1}} \mathbf{f}_t + \sum_{t \in [l_{j-1}+1, l_j-1]} \left(d_{t,l_j-1} \mathbf{g}_t + \mathbf{m}_t\right) + \sum_{t \in [l_j, \ell-1]} d_{t+1,\ell} \mathbf{f}_t.$$

Note that, $l_{j-1} \in H$ due to sufficiency condition (i).

Now, we give the set of points $\breve{w}_j$ where $j \in [k-1, \ell-1]$:

(xiii) If $j+1 \notin L^+$, then consider the points:

$$\breve{w}_j = w_{l_1} + \mathbf{m}_j + \varepsilon \mathbf{g}_j + \varepsilon \mathbf{f}_j.$$

For $j \in L^+$, we use indices l_j, where $j = 1, \ldots, |L^+|$ as in the definitions of $\hat{w}_j$. If $l_{j-1} \in [p, \ell-1]$, then define

$$\check{w}_{l_j} = \sum_{t\in[1,k-2]\cup[\ell+1,n]} (d_t\mathbf{e}_t + \mathbf{h}_t) + \left(d_{k,l_{j-1}} - u_{k-1}\right)\mathbf{e}_{\hat{t}} + \mathbf{h}_{\hat{t}} + \sum_{t\in[k-1,\ell-1]\setminus\{l_{j-1}\}} \mathbf{k}_t +$$
$$(d_{k-1} + u_{k-1} - \varepsilon)\mathbf{e}_{k-1} + \mathbf{h}_{k-1} + \left(d_{l_{j-1}+1,\ell} + \varepsilon\right)\mathbf{e}_{l_j} + \sum_{t\in[k,\hat{t}-1]} (u_{k-1} - d_{kt} - \varepsilon)\mathbf{f}_t +$$
$$\sum_{t\in[\hat{t},l_{j-1}-1]} (d_{t,j-1} - \varepsilon)\,\mathbf{f}_t + \sum_{t\in[l_{j-1}+1,l_j-1]} \left((d_{t,l_j-1} + \varepsilon)\mathbf{g}_t + \mathbf{m}_t\right) + \sum_{t\in[l_j,\ell-1]} d_{t+1,\ell}\mathbf{f}_t.$$

Suppose that, $l_{j-1} \in [k-1, p-1]$. Due to the sufficiency condition (i), we assume $l_{j-1} \in H$. Then, consider the solution:

$$\check{w}_{l_j} = \sum_{t\in[1,k-2]\cup[\ell+1,n]} (d_t\mathbf{e}_t + \mathbf{h}_t) + (d_{k-1,l_{j-1}} - \varepsilon)\mathbf{e}_{k-1} + \mathbf{h}_{l_j} + (d_{l_{j-1}+1,\ell} + \varepsilon)\mathbf{e}_{l_j}$$
$$+ \sum_{t\in[k-1,\ell-1]\setminus\{l_{j-1}\}} \mathbf{k}_t + \sum_{t\in[k-1,l_{j-1}-1]} (d_{t+1,l_{j-1}} - \varepsilon)\mathbf{f}_t +$$
$$\sum_{t\in[l_{j-1}+1,l_j-1]} \left((d_{t,l_j-1} + \varepsilon)\mathbf{g}_t + \mathbf{m}_t\right) + \sum_{t\in[l_j,\ell-1]} d_{t+1,\ell}\mathbf{f}_t.$$

So far, we provided $5n-3$ affinely independent points. Let $\hat{w}_0$ be the last affinely independent point that is on face F:

$$\hat{w}_0 = \hat{w}_{l_1} + \varepsilon\mathbf{e}_{l_1} + \varepsilon \sum_{t\in[k-1,l_1-1]} \mathbf{g}_t + \mathbf{m}_{k-1}.$$

□

In the next theorem, we provide sufficient conditions for inequality (4.58). First, we define the following indices:

$$q = \max\{j \in L^+ : b_\ell \le d_{j\ell}\},$$

$$\check{t} = \max\{j \in [k-1,\ell] \setminus L^+\}$$

and

$$t' = \min\{j \in L^+ : j \ge q\}.$$

Theorem 4.29. For a node set $[k,\ell]$, let $T \subseteq [k,\ell]$, $K = L^+$ in Proposition 4.21. Recall that in $S^+ = \{(\ell+1,\ell)\}$ and $S^- = \emptyset$ in inequality (4.58). If $d_j > 0$ for all $j \in [k,\ell]$, then the conditions below are sufficient for inequality (4.58) to be facet-defining for $\text{conv}(\mathcal{P})$:

(i) $(\ell+1,\ell) \cup \{[\,q+1,\ell\,] \cap L^+\} \subseteq H$;

(ii) for all $j \in L^+$, $\rho_j(\{(\ell+1,\ell)\}) > 0$;

(iii) $b_{t'-1} > d_{k,t'-1}$;

(iv) $b_{l_L-1} \geq d_{k,l_L-1}$;

(v) if $b_\ell = d_{q+1,\ell}$, then $l_L > q$, otherwise l_L $geqq$;

(vi) $\check{t} \in [q, \ell]$, $b_{\check{t}-1} \geq d_{k,\check{t}-1}$ and $u_{\check{t}} \geq d_{\check{t}+1,\ell}$;

(vii) if $k > 1$, then $b_{l_1-1} \geq d_{k-1,l_1-1}$;

(viii) for all $l_j \in L^+$, $u_{l_j} > d_{l_j+1,l_{j+1}}$.

Proof. Inequality (4.58) can be considered as a mirror image of inequality (4.32), where the ordering of nodes $[1, n]$ is changed to $\{n, \ldots, 1\}$ and inventory and backlog arcs are flipped. Consequently, the $5n - 2$ affinely independent points provided in Theorem 4.28 can be transformed in the same fashion so that they are on the face induced by inequality (4.58). □

4.3.2 Finding violated inequalities

In this section, we discuss how to find lifted path pack inequalities introduced in Section 4.3 that are violated by a given point $(\bar{x}, \bar{y}, \bar{s}, \bar{r}, \bar{q}, \bar{z}) \in \mathbb{R}^{6n}$. For a given $[k, \ell] \subseteq [1, n]$, and the sets S^+ and S^-, the set L^+ that maximizes the violation of these inequalities can be found in $O(n^2)$ time. We explain this procedure using inequality (4.32), where the set $M = L^+$ for simplification. We rearrange this inequality such that for a given $[k, \ell]$ selection and the fractional solution, the right hand side is a constant value:

$$\sum_{j \in L^+} \left(y_j - \rho_j x_j - \delta_j q_{j-1}\right) + \sum_{j \in H} \gamma_j (1 - z_j) \leq u_{k-1} + i_\ell + r_{k-1} - i_{k-1}.$$

The separation problem aims to find the sets L^+ and H that maximize the left hand side of the inequality above. In Atamtürk and Küçükyavuz (2005), the authors model this problem as a longest path algorithm. We follow a similar approach here.

We construct a graph consisting of $O(n^2)$ nodes and $O(n^3)$ arcs. Let the nodes be represented by pairs such as $[i', j]$ and $[i', j']$. The first item of the pair keeps track of the last time period that was selected to be in L^+. If the path passes by the node $[i', j]$, then the arc $(j, j+1)$ is selected to be in H and the last production arc in L^+ is i'. Similarly, if the path passes by the node $[i', j']$, then the production arc $j' \in L^+$ and the previously chosen item in L^+ is i'. Note that, tracking i' now allows us to compute the coefficient $\delta_{j'}$. More formally the nodes of this temporary graph can be listed as follows:

1) $[i', j] \; : \; i' \in [k-1, \ell-1], \quad j \in [i'+1, \ell-1]$,

2) $[i', j'] \; : \; i' \in [k-1, \ell-1], \quad j' \in [i'+1, \ell]$,

3) $[k-1]$: select the inventory arc $(k-1,k)$ to be in H,

4) $[(k-1)']$: arc $(k-1,k)$ is not in H,

5) $[0]$: dummy source node.

Similarly, the arcs connecting the nodes described above are as follows:

1) $[i',j] \to [i',t]$, for $i' \in [k-1,\ell-2]$, $j \in [i'+1,\ell-2]$ and $t \in [j+1,\ell-1]$ with cost

$$(1-\bar{z}_j) \times \left[\min\{(u_{k-1}-d_{kj})^+, d_{j+1,t}\}\right],$$

2) $[i',j'] \to [j',t']$, for $i' \in [k-1,\ell-1]$, $j' \in [i'+1,\ell]$ and $t' \in [j'+1,\ell]$ with cost

$$\bar{y}_{j'} + \min\{d_{kj'}+u_{j'}-u_{k-1}, d_{k\ell}-u_{k-1}, u_{j'}+d_{j'}, d_{j'\ell}\} \times \bar{x}_{j'} \\ - \min\{\bar{r}_{j'-1}, \bar{q}_{j'-1} \times \min\{b_{j'-1}, d_{i'j'-1}\}\},$$

3) $[i',j] \to [i',t']$, for $i' \in [k-1,\ell-2]$, $j \in [i'+1,\ell-2]$ and $t' \in [i',\ell]$ with cost

$$(1-\bar{z}_j) \times \left[\min\{(u_{k-1}-d_{kj})^+, d_{j+1,t'-1}\}\right],$$

4) $[i',j'] \to [j',t]$, for $i' \in [k-1,\ell-1]$, $j' \in [i'+1,\ell]$ and $t \in [j'+1,\ell-1]$ with cost

$$\bar{y}_{j'} + \min\{d_{kj'}+u_{j'}-u_{k-1}, d_{k\ell}-u_{k-1}, u_{j'}+d_{j'}, d_{j'\ell}\} \times \bar{x}_{j'} \\ - \min\{\bar{r}_{j'-1}, \bar{q}_{j'-1} \times \min\{b_{j'-1}, d_{i'j'-1}\}\},$$

5) $[k-1] \to [(k-1)',j]$, for $j \in [k,\ell-1]$ with cost

$$(1-\bar{z}_{k-1}) \times [\min\{u_{k-1}, d_{kj}\}],$$

6) $[k-1] \to [(k-1)',j']$, for $j \in [k,\ell]$ with cost

$$(1-\bar{z}_{k-1}) \times [\min\{u_{k-1}, d_{k,j'-1}\}],$$

7) $[(k-1)'] \to [(k-1)',j']$, for $j' \in [k,\ell]$ with cost zero,

8) $[(k-1)'] \to [(k-1)',j]$, for $j \in [k,\ell-1]$ with cost zero,

9) $[0] \to [k-1]$ with cost zero,

10) $[0] \to [(k-1)']$ with cost zero.

Consequently the longest path from the dummy source node [0] gives the sets L^+ and H that maximize the violation of inequality (4.32) by a given fractional solution $(\bar{x}, \bar{y}, \bar{s}, \bar{r}, \bar{q}, \bar{z}) \in \mathbb{R}^{6n}$. Since the graph described here is a directed acyclic graph consisting of $O(n^3)$ arcs and $O(n^2)$ nodes, the longest path can be found in $O(n^3)$ time. This structure can be repeated for all extended path pack inequalities where the costs are updated.

4.4 Computational Study

We test the effectiveness of the cuts introduced in this chapter by embedding them in a branch-and-cut framework. The experiments are ran on a Linux workstation with 3.60 GHz Intel® Xeon® CPU E5-1650 and 32 GB of RAM with 1 hour time limit and 1 GB memory limit. The branch-and-cut algorithm is implemented in C++ using IBM's Concert Technology of CPLEX (version 12.5). The experiments are ran with one hour limit on elapsed time and 1 GB limit on memory usage. The number of threads is set to one and the dynamic search is disabled. We also turn off heuristics and preprocessing as the purpose is to see the impact of the inequalities by themselves.

Instance Generation

Let n be the total number of time periods and f be the ratio of the fixed cost to the variable cost associated with a production arc. The parameter c controls how large the production arc capacities are with respect to average demand. All parameters are generated from a discrete uniform distribution. The demand for each node is drawn from the range $[1, 30]$. Let $\bar{d}$ be the average demand over all time periods. Inventory and backlogging upper-bounds are drawn from $[1.0 \times \bar{d}, 2.0 \times \bar{d}]$ and $[0.3 \times \bar{d}, 0.8 \times \bar{d}]$, respectively. The variable costs v_t^p, v_t^i and v_t^b are drawn from the ranges $[1, 10]$, $[1, 10]$ and $[1, 20]$, respectively. Finally, fixed ordering costs f_t^p are set equal to $f \times v_t^p$, fixed inventory holding cost is $f_t^i = f$, for all $t \in [1, n]$ and fixed backlogging cost is set to $f_t^b = 2 \times f$, for all $t \in [1, n]$. Using these parameters, we generate five random instances for each combination of $n \in \{50, 100, 150, 200\}$, $f \in \{100, 200, 500, 1000\}$ and $c \in \{2, 5, 10\}$.

Results

We report multiple performance measures. Let z_{INIT} be the objective function value of the initial LP relaxation and z_{ROOT} be the objective function value of the LP relaxation after all the valid inequalities added. Moreover, let z_{UB} be the objective function value of the best feasible solution found within time/memory limit among all experiments for an instance. Let `init gap`$= 100 \times \frac{z_{\text{UB}} - z_{\text{INIT}}}{z_{\text{UB}}}$, `root gap`$= 100 \times \frac{z_{\text{UB}} - z_{\text{ROOT}}}{z_{\text{UB}}}$. We compute the improvement of the relaxation due to adding valid inequalities as `gap`

$\texttt{imp}= 100 \times \frac{\texttt{init gap}-\texttt{root gap}}{\texttt{init gap}}$. We also measure the optimality gap at termination as $\texttt{end gap} = \frac{z_{\text{UB}}-z_{\text{LB}}}{z_{\text{UB}}}$, where z_{LB} is the value of the best lower bound given by CPLEX. We report the average number of valid inequalities added at the root node under column `cuts`, average elapsed time in seconds under `time`, average number of branch-and-bound nodes explored under `nodes`. If there are instances that are not solved to optimality within the time/memory limit, we report the the end gap averaged over unsolved instances under `end gap` and the number of unsolved instances under `unslvd` next to `time` results. All numbers except initial gap, end gap and time are rounded to the nearest integers.

In Table 4.1, we present the results for path pack inequalities introduced in Section 4.2 under columns `spi`. Under columns `lspi`, we show results for both path pack inequalities and lifted path pack inequalities introduced in Section 4.3. For comparison, under `base` columns, we present the results for experiments where no valid inequalities are added. Consequently, in these experiments, the gap improvement and the number of cuts added are zero for all instances. We observe that the introduction of the inventory and backlog fixed charge variables to path pack inequalities, the average gap improvement increases by 6%. Moreover, the number of branch-and-bound nodes explored decreases by 90%. The average elapsed time is more than double without the lifted inequalities. All of the `base` experiments terminated early due to the 1GB memory limit for instances with $n = 100$. As a result, we observe a lower average elapsed time with a 22% average optimality gap.

In Table 4.2, we investigate the classes of inequalities introduced in Sections 4.2 and 4.3 deeper. Under `lspi`, we present the same results as in Table 4.1 where all classes of inequalities are used. Under columns $[i, j]$, we use a only a certain subset of inequalities. In this naming scheme, $i \in \{1, 2, 3\}$ and $j \in \{0, 1, 2, 3\}$, where $i = 1$ implies that the set is $S^+ = \{(k-1, k)\}$, $i = 2$ implies $S^+ = \{(\ell+1, \ell)\}$, $i = 3$ implies $S^+ = \{(k-1, k), (\ell+1, \ell)\}$ and similarly, $j = 0$ implies $S^- = \emptyset$, $j = 1$ implies $S^- = \{(k, k-1)\}$, $j = 2$ implies $S^- = \{(\ell+1, \ell)\}$ and $j = 3$ implies $S^- = \{(k, k-1), (\ell+1, \ell)\}$. More specifically in columns `[1,0]`, inequalities (4.10) and (4.32), in `[1,1]`, inequalities (4.18) and (4.49), in `[1,2]`, inequalities (4.16) and (4.46), in `[1,3]`, inequalities (4.17) and (4.47), in `[2,0]`, inequalities (4.15) and (4.58), in `[2,1]`, inequalities (4.19) and (4.62), in `[2,2]`, inequalities (4.21) and (4.64), in `[2,3]`, inequalities (4.20) and (4.63), in `[3,1]`, inequalities (4.24), (4.57) and (4.68), in `[3,2]`, inequalities (4.23), (4.55) and (4.67) and finally in `[3,3]`, inequalities (4.22), (4.50) and (4.65) are used. We observe that each (S^+, S^-) selection perform relatively well on their own, however when all cuts used simultaneously, the performance is dominating both in terms of time and gap improvement. For this instance set, column `[1,0]` (the set selection $S^+ = \{(k-1, k)\}$ and $S^- = \emptyset$) give the best performance and in comparison, its symmetrical case `[2,0]` (the selection $S^+ = \{(\ell+1, \ell)\}$ and $S^+ = \emptyset$) performs worse. This difference in performance is because the backlog arc capacities are generated to be smaller and it reflects on the effect of the cuts.

In Table 4.3, we investigate the marginal contribution of path pack inequalities of Sections 4.2 and 4.3 when added on top of CPLEX's network cuts: multi-commodity flow (MCF), flow cover and flow path inequalities. Under the columns titled `lspix`, we report results for computations where both path pack inequalities and CPLEX's network cuts are used. Similarly, under `cpx` columns, we report results for computations where only CPLEX's network cuts are used. In this table, the columns `cuts` report the total number of cuts added to the branch-and-bound tree averaged over instances. On average, about 82% of the cuts added under `lspix` column are path pack inequalities. When path pack inequalities are used in addition to CPLEX's network cuts, the gap improvement increases by 7%, the number of branch-and-bound nodes explored decreases by four orders of magnitude and the average time spent decreases by three orders of magnitude. Moreover, all of the instances were solved to optimality within our time/memory limits when path pack inequalities were used.

Table 4.1: Performance of lifted submodular path inequalities.

n	f	c	init gap	gap imp		nodes			cuts		time (endgap:unsolved)		
				lspi	spi	lspi	spi	base	lspi	spi	lspi	spi	base
50	100	2	76.5	95%	90%	200	2326	7733726	2670	2244	1	2	368
		5	77.2	95%	90%	319	6277	11449395	2583	2117	1	4	574
		10	77.6	94%	89%	435	3075	7174514	2777	2340	1	2	349
	200	2	84.7	95%	90%	534	5449	22236405	2850	2309	2	7	1049
		5	85.4	93%	87%	1631	12566	6551607	2805	2215	5	14	287
		10	86.6	93%	87%	589	5287	5827073	2723	2230	2	5	276
	500	2	92.2	93%	89%	1815	9845	17122540	2782	2351	4	9	819
		5	92.6	92%	86%	490	4151	3784405	2926	2111	2	3	178
		10	93.1	93%	89%	5608	46488	16530846	2912	2327	11	37	818
	1000	2	95.6	93%	87%	723	7415	8718542	3045	2342	4	9	422
		5	95.5	93%	88%	869	4704	9696194	3012	2246	3	5	493
		10	95.6	92%	87%	2204	33519	9683838	3020	2237	10	41	439
100	100	2	76.9	95%	90%	26204	623083	11962729	13218	10108	781 (1.1,1)	2825 (0.7,3)	753 (38.9,5)
		5	76.8	95%	90%	13631	407698	11998279	13382	10908	241	2220 (0.1,2)	771 (39.6,5)
		10	77.1	95%	91%	18190	522983	11999348	14402	11312	403	3600 (0.1,5)	779 (40.0,5)
	200	2	85.5	94%	90%	30257	677990	11972480	13955	10101	1229 (0.5,1)	3600 (0.2,5)	751 (43.4,5)
		5	85.9	94%	89%	22741	604804	12272094	13192	9907	863	3199 (0.1,4)	790 (45.6,5)
		10	86.6	94%	89%	20963	578362	11749288	14922	11021	1177 (0.7,1)	3139 (0.5,3)	753 (42.1,5)
	500	2	93.2	92%	87%	39155	818058	12097739	15173	10272	1558 (4.5,1)	3600 (0.9,5)	809 (49.8,5)
		5	93.4	92%	88%	51935	707097	12116225	16119	10906	2269 (0.7,2)	3600 (0.5,5)	771 (50.2,5)
		10	93.6	93%	88%	63620	748193	12039418	15205	10335	1984	3559 (0.3,4)	784 (44.9,5)
	1000	2	96.2	92%	87%	34574	653858	12225463	16054	10708	2946 (1.0,4)	3491 (1.1,4)	784 (48.3,5)
		5	96.3	93%	87%	31879	636670	12222040	17441	12052	1704 (0.6,2)	3600 (0.7,5)	780 (42.5,5)
		10	96.2	93%	88%	36508	730331	12391726	17043	11507	2077 (2.0,2)	3600 (1.1,5)	816 (48.9,5)
Average:			**87.9**	**94%**	**88%**	**16878**	**327093**	**11314830**	**8925**	**6509**	**720 (0.2,1)**	**1674 (1.1,4)**	**642 (22.3,3)**

Table 4.2: Performance of each class of `lspi` broken down.

n	f	c	gap imp												time (sec)											
			lspi	[1,0]	[1,1]	[1,2]	[1,3]	[2,0]	[2,1]	[2,2]	[2,3]	[3,1]	[3,2]	[3,3]	lspi	[1,0]	[1,1]	[1,2]	[1,3]	[2,0]	[2,1]	[2,2]	[2,3]	[3,1]	[3,2]	[3,3]
50	100	2	95%	87%	79%	64%	64%	81%	72%	82%	74%	77%	89%	81%	1	13	337	660	502	102	867	22	121	320	5	103
		5	95%	88%	79%	62%	62%	76%	69%	78%	70%	74%	87%	78%	1	123	342	1197	1486	399	733	64	177	602	10	204
		10	94%	86%	80%	66%	65%	80%	73%	81%	72%	76%	86%	79%	1	15	66	484	1413	227	403	97	186	366	30	173
	200	2	95%	88%	79%	64%	63%	79%	71%	82%	72%	76%	88%	78%	2	17	279	842	739	192	508	88	235	695	10	171
		5	93%	85%	79%	65%	64%	77%	69%	79%	69%	75%	84%	76%	5	93	809	667	1183	504	1400	306	447	387	39	1273
		10	93%	85%	78%	63%	63%	74%	67%	78%	68%	74%	87%	77%	2	29	115	1081	950	355	1148	77	185	355	10	403
	500	2	93%	89%	82%	68%	67%	76%	69%	79%	71%	77%	88%	78%	4	21	85	274	391	794	951	183	464	350	11	762
		5	92%	87%	81%	68%	66%	77%	69%	77%	69%	76%	84%	78%	2	3	26	168	200	82	470	57	130	52	6	77
		10	93%	88%	80%	69%	68%	77%	71%	78%	72%	77%	87%	78%	11	82	859	1526	1091	1080	2185	255	998	1628	21	1048
	1000	2	93%	89%	81%	67%	66%	74%	70%	77%	69%	77%	86%	78%	4	30	308	1278	616	227	760	226	117	329	20	297
		5	93%	90%	84%	69%	68%	76%	71%	78%	72%	78%	88%	79%	3	29	313	797	807	773	789	176	294	765	14	405
		10	92%	87%	81%	68%	67%	76%	69%	78%	69%	76%	86%	77%	10	50	386	916	1227	782	958	242	667	772	22	312
Average			**93%**	**88%**	**80%**	**66%**	**65%**	**77%**	**70%**	**79%**	**71%**	**76%**	**87%**	**78%**	**4**	**42**	**327**	**824**	**884**	**460**	**931**	**149**	**335**	**552**	**16**	**436**

Table 4.3: Marginal contribution of path pack inequalities to CPLEX's network cuts.

n	f	c	init gap	gap imp		nodes		cuts		time (endgap:unsolved)	
				lspix	cpx	lspix	cpx	lspix	cpx	lspix	cpx
100	100	2	76.9	99%	94%	59	67454	848	247	0	10
		5	76.8	99%	94%	14	15810	909	255	0	3
		10	77.1	99%	93%	19	35567	902	268	0	6
	200	2	85.5	99%	93%	40	11114	881	251	0	2
		5	85.9	99%	92%	41	103095	864	251	0	17
		10	86.6	99%	93%	20	8537	866	246	0	1
	500	2	93.2	98%	91%	100	43221	861	238	0	7
		5	93.4	98%	93%	159	8181	883	263	0	1
		10	93.6	98%	94%	57	5537	884	252	0	1
	1000	2	96.2	98%	92%	63	15549	888	249	0	3
		5	96.3	98%	92%	1087	15236	892	241	1	3
		10	96.2	98%	92%	90	29127	897	254	0	5
150	100	2	76.7	99%	93%	458	1614054	1316	387	1	391
		5	77.9	99%	93%	38	619615	1313	365	0	147
		10	76.4	99%	93%	28	135341	1292	375	0	33
	200	2	86.6	99%	93%	54	1911337	1298	367	0	491
		5	86.2	99%	94%	60	184399	1328	367	0	44
		10	85.8	98%	92%	341	1277649	1340	380	1	315
	500	2	93.5	98%	92%	413	949530	1318	365	1	231
		5	93.4	99%	92%	72	4064991	1324	356	0	963
		10	93.6	98%	93%	186	1004958	1340	362	0	241
	1000	2	96.3	98%	91%	195	2963744	1325	360	0	717
		5	96.3	98%	92%	254	2310102	1335	369	1	553
		10	96.4	99%	92%	71	646165	1329	357	0	154
200	100	2	76.6	99%	93%	165	4397657	1726	509	1	1182 (0.2,1)
		5	76.8	99%	94%	126	3448956	1760	484	1	918 (0.0,1)
		10	77.3	99%	93%	91	11081170	1777	487	0	2885 (0.0,4)
	200	2	86.8	99%	93%	79	5192691	1794	486	1	1435 (0.1,1)
		5	86.2	98%	93%	750	3870962	1805	480	2	1060 (0.0,1)
		10	86.7	99%	93%	246	5639377	1769	486	1	1660 (0.1,1)
	500	2	93.7	98%	93%	751	9542705	1794	487	2	2448 (0.1,4)
		5	93.6	98%	92%	510	9506335	1800	490	2	2583 (0.4,4)
		10	93.5	98%	91%	742	8517221	1834	493	2	2074 (0.5,4)
	1000	2	96.5	98%	92%	2053	9359661	1814	489	5	2548 (0.1,4)
		5	96.5	98%	91%	448	10253848	1869	504	1	2828 (0.4,4)
		10	96.6	99%	92%	1541	6089233	1783	489	4	1555 (0.8,3)
Average:			**88.3**	**99%**	**92%**	**317**	**2915004**	**1332**	**370**	**1**	**764 (0.2,1)**

Chapter 5

Summary of Thesis and Conclusions

In this thesis, we study various fixed-charge networks by formulating them as linear mixed-integer programs. We propose different classes of valid inequalities and show their strength both theoretically and computationally. These valid inequalities are derived from submodular inequalities that are initially proposed by Wolsey (1989).

After giving a brief introduction and some preliminary definitions in Chapter 1, we focus on simple directed path structures of capacitated fixed-charge networks in Chapter 2. We show how to efficiently compute an explicit form of the submodular inequalities. We refer to these explicitly expressed submodular inequalities as path cover inequalities. Furthermore, we obtain a generalized class through a simultaneous lifting procedure. In Chapter 3, we focus on slightly more general paths where consecutive nodes j and $j+1$ are connected through a forward path arc $(j, j+1)$ and a backward path arc $(j+1, j)$. In this chapter, we give two explicit descriptions of submodular inequalities and refer them as path cover and path pack inequalities.

We provide necessary and sufficient conditions under which the inequalities introduced in Chapters 2 and 3 are facet defining for the convex hull of the feasible solutions. Moreover, we present extensive computational results that show the effectiveness of these inequalities when used in a branch-and-cut framework.

In Chapter 4, we study single item lot-sizing problems with backlogging and inventory bounds. This class of the lot-sizing problem is a special case of the path structure studied in Chapter 3. Since we assume that the production arcs are uncapacitated, the coefficients of path pack inequalities were obtained parametrically. Using different arc set selections, we give eleven classes of path pack inequalities. Then, we incorporate the binary variables of inventory and backlog arcs to path pack inequalities using a lifting procedure and present computational results.

Bibliography

Agra, A. and Constantino, M. (1999). Lotsizing with backlogging and start-ups: The case of Wagner–Whitin costs. *Operations Research Letters*, 25(2):81–88.

Atamtürk, A. (2001). Flow pack facets of the single node fixed–charge flow polytope. *Operations Research Letters*, 29:107–114.

Atamtürk, A. (2004). Sequence independent lifting for mixed-integer programming. *Operations Research*, 52(3):487–490.

Atamtürk, A., Gómez, A., and Küçükyavuz, S. (2016). Three-partition flow cover inequalities for constant capacity fixed-charge network flow problems. *Networks*, 67:299–315.

Atamturk, A., Küçükyavuz, S., and Tezel, B. (2017). Path cover and path pack inequalities for the capacitated fixed-charge network flow problem. *SIAM Journal on Optimization*, 27(3):1943–1976.

Atamtürk, A. and Küçükyavuz, S. (2005). Lot sizing with inventory bounds and fixed costs. *Operations Research*, 53:711 – 730.

Atamtürk, A. and Muñoz, J. C. (2004). A study of the lot–sizing polytope. *Mathematical Programming*, 99:43–65.

Balas, E. and Zemel, E. (1978). Facets of the knapsack polytope from minimal covers. *SIAM Journal on Applied Mathematics*, 34(1):119–148.

Brahimi, N., Dauzere-Peres, S., Najid, N. M., and Nordli, A. (2006). Single item lot sizing problems. *European Journal of Operational Research*, 168(1):1–16.

Chu, C., Chu, F., Zhong, J., and Yang, S. (2013). A polynomial algorithm for a lot-sizing problem with backlogging, outsourcing and limited inventory. *Computers & Industrial Engineering*, 64(1):200–210.

Constantino, M. (2000). A polyhedral approach to a production planning problem. *Annals of Operations Research*, 96(1-4):75–95.

Federgruen, A. and Tzur, M. (1993). The dynamic lot-sizing model with backlogging: A simple $o(n \log n)$ algorithm and minimal forecast horizon procedure. *Naval Research Logistics*, 40:459–478.

Gade, D. and Küçükyavuz, S. (2011). A note on lot-sizing with fixed charges on stocks: the convex hull. *Discrete Optimization*, 8:385–392.

Ganas, I. and Papachristos, S. (2005). The single-product lot-sizing problem with constant parameters and backlogging: exact results, a new solution, and all parameter stability regions. *Operations research*, 53(1):170–176.

Gomory, R. E. (1969). Some polyhedra related to combinatorial problems. *Linear algebra and its applications*, 2(4):451–558.

Gu, Z. (1994). *Lifted cover inequalities for 0-1 and mixed integer programs*. PhD thesis, Georgia Institute of Technology.

Gu, Z., Nemhauser, G. L., and Savelsbergh, M. W. P. (1999). Lifted flow cover inequalities for mixed 0–1 integer programs. *Mathematical Programming*, 85:439–467.

Gu, Z., Nemhauser, G. L., and Savelsbergh, M. W. P. (2000). Sequence independent lifting in mixed integer programming. *Journal of Combinatorial Optimization*, 4:109–129.

Hille, E. and Phillips, R. S. (1957). *Functional Analysis and Semi-Groups*. American Mathematical Society, Providence.

Karimi, B., Ghomi, S. F., and Wilson, J. (2003). The capacitated lot sizing problem: a review of models and algorithms. *Omega*, 31(5):365–378.

King, V., Rao, S., and Tarjan, R. (1994). A faster deterministic maximum flow algorithm. *Journal of Algorithms*, 17:447–474.

Küçükyavuz, S. and Pochet, Y. (2009). Uncapacitated lot sizing with backlogging: the convex hull. *Mathematical Programming*, 118:151–175.

Ortega, F. and Wolsey, L. A. (2003). A branch-and-cut algorithm for the single-commodity, uncapacitated, fixed-charge network flow problem. *Networks*, 41(3):143–158.

Padberg, M. W., van Roy, T. J., and Wolsey, L. A. (1985). Valid linear inequalities for fixed charge problems. *Operations Research*, 33:842–861.

Pochet, Y. and Wolsey, L. A. (1988). Lot-size models with backlogging: Strong reformulations and cutting planes. *Mathematical Programming*, 40:317–335.

Pochet, Y. and Wolsey, L. A. (1994). Polyhedra for lot-sizing with Wagner–Whitin costs. *Mathematical Programming*, 67:297–323.

Pochet, Y. and Wolsey, L. A. (2006). *Production planning by mixed integer programming*. Springer Science & Business Media.

Rardin, R. L. and Wolsey, L. A. (1993). Valid inequalities and projecting the multicommodity extended formulation for uncapacitated fixed charge network flow problems. *European Journal of Operational Research*, 71:95–109.

Stallaert, J. I. A. (1997). The complementary class of generalized flow cover inequalities. *Discrete Applied Mathematics*, 77:73–80.

van Roy, T. J. and Wolsey, L. A. (1985). Valid inequalities and separation for uncapacitated fixed charge networks. *Operations Research Letters*, 4:105–112.

van Roy, T. J. and Wolsey, L. A. (1986). Valid inequalities for mixed 0–1 programs. *Discrete Applied Mathematics*, 14:199–213.

Van Vyve, M. (2006). Linear-programming extended formulations for the single-item lot-sizing problem with backlogging and constant capacity. *Mathematical Programming*, 108(1):53–77.

Van Vyve, M. (2013). Fixed-charge transportation on a path: optimization, LP formulations and separation. *Mathematical Programming*, 142:371–395.

Van Vyve, M. and Ortega, F. (2004). Lot-sizing with fixed charges on stocks: the convex hull. *Discrete Optimization*, 1:189–203.

Wolsey, L. A. (1976). Facets and strong valid inequalities for integer programs. *Operations Research*, 24(2):367–372.

Wolsey, L. A. (1977). Valid inequalities and superadditivity for 0-1 integer programs. *Mathematics of Operations Research*, 2(1):66–77.

Wolsey, L. A. (1989). Submodularity and valid inequalities in capacitated fixed charge networks. *Operations Research Letters*, 8:119–124.

Wolsey, L. A. (1998). *Integer Programming*. John Wiley and Sons.

Wolsey, L. A. and Nemhauser, G. L. (1999). *Integer and Combinatorial Optimization*. Wiley Series in Discrete Mathematics and Optimization. Wiley.

Wu, T., Shi, L., Geunes, J., and Akartunalı, K. (2011). An optimization framework for solving capacitated multi-level lot-sizing problems with backlogging. *European Journal of Operational Research*, 214(2):428–441.

Zangwill, W. I. (1966). A deterministic multi-period production scheduling model with backlogging. *Management Science*, 13(1):105–119.

Zangwill, W. I. (1969). A backlogging model and a multi-echelon model of a dynamic economic lot size production system-a network approach. *Management Science*, 15(9):506–527.

Zemel, E. (1978). Lifting the facets of zero–one polytopes. *Mathematical Programming*, 15(1):268–277.

Zhong, J., Chu, F., Chu, C., and Yang, S. (2016). Polynomial dynamic programming algorithms for lot sizing models with bounded inventory and stockout and/or backlogging. *Journal of Systems Science and Systems Engineering*, 25(3):370–397.

Appendices

Appendix A

Equivalency of $\mathcal{P}'$ to the maximum flow problem

Recall that $K^+ = \{t \in E^+ : a_t = 1\}$ and $K^- = \{t \in E^- : a_t = 0\}$, where the values a_t are the coefficients of arcs in E in the objective function of $\mathcal{P}'$.

Proposition A.1. If $d_j \geq 0$ for all $j \in V$, then there exists an optimal solution to the optimization problem $\mathcal{P}'$ where $y_t = 0$ for all $t \in E^- \setminus K^-$.

Proof. Given an optimal solution $(\mathbf{y}^*, \mathbf{i}^*)$ to $\mathcal{P}'$, for notational convenience, let $\bar{Y}_j^- := \sum_{t \in E_j^- \setminus L_j^-} y_t^*$ and $\bar{Y}_j^+ := \sum_{t \in S_j^+} y_t^*$ for $j \in V$. Let

$$\epsilon_j' = d_j - \left[\bar{i}_{j-1} + \bar{Y}_j^+\right] + \left[\bar{Y}_j^- + \bar{i}_j\right]$$

be the slack of flow balance constraint (2.6) of node j. Moreover, let p be some node in V with $\bar{Y}_p^- = \epsilon > 0$. Observe that if $\epsilon_p' > 0$, then decreasing the outgoing flow $\bar{Y}_p^-$ by ϵ_p' increases the objective function value by ϵ_p' while preserving feasibility. This operation provides contradiction with the assumption of optimality. Therefore, we assume that $\epsilon_j' = 0$ for nodes j with $\bar{Y}_j^- > 0$.

Then, applying the procedure described in Algorithm 2 to the optimal solution $(\mathbf{y}^*, \mathbf{i}^*)$, we find an optimal solution where $\bar{y}_t = 0$ for all $t \in E^- \setminus K^-$.

□

Remark A.1. If $d_j < 0$ for some $j \in V$, one can represent the supply amount as a dummy arc in E_j^+ with a fixed flow and capacity of $-d_j$ and set the modified demand of node j to be $d_j = 0$.

If the path V has nodes j with $d_j < 0$, then using Remark A.1, we convert it into a path where $d_j \geq 0$ for all $j \in V$. After the modification of V, we note that the dummy supply arcs will always be open, therefore, they will always be included in the set S^+. Let us call the constraints of fixed flow value on the dummy supply arc *fixed-flow constraints* for the sake of conciseness. Notice that the formulation $\mathcal{P}'$ becomes a

Algorithm 2

$(\mathbf{y}^*, \mathbf{i}^*)$: an optimal solution to $\mathcal{P}'$.
$\mathcal{J} = \{k \le j \le \ell : \bar{Y}_j^- > 0\}$.
for $p \in \mathcal{J}$ **do**
  $\epsilon := \bar{Y}_p^-$, $\bar{Y}_p^- = 0$
  for $j = p$ **to** k **do**
    $\Delta = \min\{\epsilon, \bar{Y}_j^+\}$
    $\bar{Y}_j^+ = \bar{Y}_j^+ - \Delta$
    $\bar{i}_{j-1} = \bar{i}_{j-1} - (\epsilon - \Delta)$
    $\epsilon = \epsilon - \Delta$
  end for
end for

relaxation for the modified path because of the missing fixed-flow constraints. In the next proposition, we prove that the optimal objective function value does not change by adding/dropping the fixed-flow constraints to $\mathcal{P}'$.

Proposition A.2. Suppose $d_j < 0$ for some $j \in V$. Let G' be the modification of graph G using Remark A.1. If the optimization problem $\mathcal{P}'$ for G' along with the fixed-flow constraints is feasible, then the optimal objective function value does not change by adding the fixed-flow constraints to $\mathcal{P}'$.

Proof. We need to show that $\mathcal{P}'$ has an optimal solution where the dummy supply arcs has flows at their capacities. Notice that the modification using Remark A.1 makes Proposition A.1 applicable to the modified graph G'. Therefore, there exists an optimal solution $(\mathbf{y}^*, \mathbf{i}^*)$ to $\mathcal{P}'$ where $y_t^* = 0$ for $t \in E^- \setminus K^-$. Let $p \in S_j^+$ represent the index of the dummy supply arc with $c_p = -d_j$. If $y_p^* < c_p$, then satisfying the fixed-flow constraints will require pushing flow through the arcs in $E^- \setminus K^-$. We use Algorithm 3 in order to obtain an optimal solution with $y_p^* = c_p$. Note that each arc in $E_j^- \setminus K^-$ affect the objective function value and constraints of $\mathcal{P}'$ the same way, therefore we merge these outgoing arcs into one in Algorithm 3. We represent the merged flow and capacity by $\bar{Y}_j^- = \sum_{t \in E_j^- \setminus K^-} y_t^*$ and $\bar{C}_j = c(E_j^- \setminus K^-)$ for $j \in V$.

□

Proposition A.2 shows that adding the fixed-flow constraints to $\mathcal{P}'$ formulated for a modified a path via Remark A.1 does not change the optimal objective function value. Consequently, without loss of generality, we assume that $d_j \ge 0$ for all $j \in V$. Then, using the optimality condition in Proposition A.1, the variables y_t for $t \in E^- \setminus K^-$ can be dropped from the formulation $\mathcal{P}'$. As a result, without loss of generality, all arcs in S^+ are leaving a dummy source node s_V, the outgoing arcs $K^- \setminus L^-$ are incoming to a dummy sink node t_V and the demands values are represented as arcs with capacity d_j, incoming to t_V from node $j \in V$ (see Figure 2.2 for a representation).

Algorithm 3

$\mathcal{J}$: Set of supply nodes in V where the nodes are sorted with respect to their order in V.
$(\mathbf{y}^*, \mathbf{i}^*)$: the optimal solution to $\mathcal{P}'$ with $y_t = 0$ for all $t \in E^-$.
for $q \in \mathcal{J}$ **do**
  Let p be the dummy supply node in S_q^+
  $\Delta = c_p - y_p^*$
  for $j = q$ **to** ℓ **do**
    $\bar{Y}_j^- = \bar{Y}_j^- + \min\{\bar{C}_j - \bar{Y}_j^-, \Delta\}$
    $\Delta = \Delta - \min\{\bar{C}_j - \bar{Y}_j^-, \Delta\}$
    $i_j^* = i_j^* + \Delta$
  end for
  if $\Delta > 0$ **then**
    The problem $\mathcal{P}'$ along with fixed-flow constraints is infeasible
  end if
end for

Appendix B

Proofs from Chapter 2

B.1 Proof of Theorem 2.6

1. If for some $t' \in S^+$, $\rho_{t'}(S^+ \setminus \{t'\}, L^-) \geq c_{t'}$, then removing t' from S^+ in inequality (2.10) results in an inequality at least as strong. To see this, let $S' = S^+ \setminus \{t'\}$. Rewriting inequality (2.10), we obtain

$$\begin{aligned}
&\sum_{t \in S'} (y_t + \rho_t(S^+ \setminus \{t\}, L^-)(1 - x_t)) + y_{t'} + \sum_{t \in L^-} \rho_t(S^+, L^- \setminus \{t\}) x_t \\
&\leq v(S^+, L^-) - \rho_{t'}(S^+ \setminus \{t'\}, L^-) + \rho_{t'}(S^+ \setminus \{t'\}, L^-) x_{t'} + y(E^- \setminus L^-) \\
&= v(S', L^-) + \rho_{t'}(S^+ \setminus \{t'\}, L^-) x_{t'} + y(E^- \setminus L^-).
\end{aligned}$$

Consider adding the submodular inequality

$$\sum_{t \in S'} (y_t + \rho_t(S' \setminus \{t\}, L^-)(1 - x_t)) + \sum_{t \in L^-} \rho_t(S', L^- \setminus \{t\}) x_t \leq v(S', L^-) + y(E^- \setminus L^-),$$

and $y_{t'} \leq c_{t'} x_{t'}$. The resulting inequality dominates inequality (2.10). This follows because $\rho_t(S' \setminus \{t\}, L^-) \geq \rho_t(S^+ \setminus \{t\}, L^-)$, from the definition of submodularity of the set function v.

2. If for some $t \in L^-$, $\rho_t(S^+, L^- \setminus \{t\}) \leq -c_t$, then summing the submodular inequality obtained by removing t from L^-, and the inequality $y_t \leq c_t x_t$ results in an inequality at least as strong.

3. If $L^- = \emptyset$ and $\max_{t \in S^+} \rho_t(S^+ \setminus \{t\}, L^-) = 0$, then summing flow balance inequalities (2.1) gives an inequality at least as strong.

4. Suppose at node p we have $c(S_p^+) \geq d_{p\ell}$, $\max_{t \in S_p^+} \rho_t(S^+ \setminus \{t\}, L^-) = 0$ and $L_p^- = \emptyset$.

Then, observe that the submodular path inequality for path V is

$$\begin{aligned}\textstyle\sum_{j=k}^{p} \sum_{t\in S_j^+} \big(y_t + (c_t - \lambda_j)^+(1-x_t)\big) + \sum_{j=p}^{\ell} y(S_j^+) \\ \le d_{k\ell} + \textstyle\sum_{j=k}^{p} \big(\lambda_j x(L_j^-) + y(E_j^- \setminus L_j^-)\big) + \sum_{j=p}^{\ell} y(E_j^-). \qquad \text{(B.1)}\end{aligned}$$

Since $c(S_p^+) \ge d_{p\ell}$, nodes $\{p,\ldots,\ell\}$ do not pull any flow through nodes $\{k,\ldots,p-1\}$ which implies that they do not have any effect on the excess values λ_j for $j \in [k, p-1]$. The submodular path inequality for path $\bar{V} = [k, p-1]$

$$\sum_{j=k}^{p-1} \sum_{t\in S_j^+} \big(y_t + (c_t - \lambda_j)^+(1-x_t)\big) \le d_{k,p-1} + i_{p-1} + \sum_{j=k}^{p-1} \big(\lambda_j x(L_j^-) + y(E_j^- \setminus L_j^-)\big)$$

summed with the flow balance inequalities (2.1) for nodes $j \in [p,\ell]$

$$i_{p-1} + \sum_{j=p}^{\ell} \big(y(E_j^+) - y(E_j^-)\big) \le d_{p,\ell}$$

gives an inequality at least as strong as inequality (B.1).

5. Suppose for some node $k < j \le \ell$, $\alpha_j = d_{k,j-1} + u_{j-1} + c(S_j^+)$ and $\cup_{i=j}^{\ell} S_i^+$ is a path cover for nodes $j,\ldots,\ell$. Let λ_i, $i \in V$ be the excess values computed for path V. We consider dividing path $V = [k,\ell]$ into two sub-paths where $V_1 = [k, j-1]$ and $V_2 = [j,\ell]$ and identify submodular path inequalities for V_1 and V_2.

   Dropping nodes $j,\ldots,\ell$ from path V does not change the excess values λ_i for $i \in V_1$ because $\cup_{i=j}^{\ell} S_i^+$ is a path cover for V_2. In other words, nodes in V_2 do not pull any flow from nodes in V_1 and therefore, they do not affect the excess values λ_i for $i \in V_1$. Then, submodular path inequality for path $V_1 = [k, j-1]$ is

$$\sum_{i\in V_1} \sum_{t\in S_i^+} \big(y_t + (c_t - \lambda_i)^+(1-x_t)\big) \le d_{k,j-1} + i_{j-1} + \sum_{i\in V_1} \big(\lambda_i x(L_i^-) + y(L_i^{--})\big).$$

   Now, we consider the submodular path inequality defined for V_2. If the path arc $(j-1, j)$ with capacity u_{j-1} is added to S_j^+, then λ_i values for $j \in V_2$ remain unchanged because $\alpha_j = d_{k,j-1} + u_{j-1} + c(S_j^+)$ implies that the path arc $(j-1,j)$ with capacity u_{j-1} provides a bottleneck for sending a flow from node k towards node j. In other words, the excess value that can be carried to node j from nodes in V_1 is at least as large as the capacity of the path arc $(j-1,j)$, u_{j-1}. Therefore, the submodular path inequality defined for path $V_2 = [j,\ell]$ is

$$i_{j-1} + \sum_{i\in V_2} \sum_{t\in S_i^+} \big(y_t + (c_t - \lambda_i)^+(1-x_t)\big) \le d_{j\ell} + \sum_{i\in V_2} \big(\lambda_i x(L_i^-) + y(L_i^{--})\big).$$

As a result, submodular path inequalities defined for path V_1 and V_2 summed give an inequality at least as strong as the submodular path inequality defined for path V.

$\square$

B.2 Proof of Theorem 2.7

Let us first introduce the effective demand $\bar{d}_j$ and effective path flow $\bar{i}_j$ at each node of the path V. From node ℓ to node k we first calculate:

$$\bar{d}_{j-1} = d_{j-1} + \left(\bar{d}_j - c(S_j^+)\right)^+ \quad \text{and} \quad \bar{i}_{j-1} = \left(\bar{d}_j - c(S_j^+)\right)^+$$

with $\bar{d}_\ell = d_\ell$ and $\bar{i}_\ell = 0$. Then, we reassign effective demand $\bar{d}_j$ to be $\bar{d}_j = \min\{c(S_j^+), \bar{d}_j\}$. We also define effective remaining capacity of path arcs as $\bar{u}_j = u_j - \bar{i}_j$. Note that $\bar{i}_j \leq u_j$ due to the assumption of S^+ defining a path cover. Using this notation, we have

$$\lambda_j = \min\{\bar{u}_{j-1}, \lambda_{j-1}\} + c(S_j^+) - \bar{d}_j.$$

Furthermore, we observe that $\bar{u}_j > 0$ because $\bar{u}_j = 0$ implies $\lambda_{j+1} = 0$ which contradicts with the first necessary condition in Theorem 2.6. We represent the feasible point p as $z^p = (y^p, x^p, i^p)$ or $w^p = (y^p, x^p, i^p)$. Define set $K = \{t \in S^+ \; : \; t = \arg\max_{t \in S_j^+} c_t, \quad j \in [k, \ell]$. First we describe the feasible point $\tilde{z} = (\tilde{y}, \tilde{x}, \tilde{i})$ where

$$\begin{aligned}
&\left(\tilde{y}_t, \tilde{x}_t\right) = \left(c_t - (\lambda_j - \min\{\bar{u}_{j-1}, \lambda_{j-1}\})^+, 1\right), && t \in S_j^+ \cap K,\\
&\left(\tilde{y}_t, \tilde{x}_t\right) = \left(c_t, 1\right), && t \in S^+ \setminus K,\\
&\left(\tilde{y}_t, \tilde{x}_t\right) = \left(0, 0\right), && t \in A \setminus (S^+ \cup I),\\
&\tilde{i}_j = \bar{i}_j, && j \in V.
\end{aligned}$$

The feasible point $\tilde{z}$ satisfies submodular path inequality at equality since $\sum_{t \in S_j^+} \tilde{y}_t = \bar{d}_j$ for all $j \in V$ and $d_{k\ell} = \sum_{j=k}^{\ell} \bar{d}_j$. Without loss of generality, let us call the arc with maximum capacity in S_k^+ arc 1. Recall that $c_1 > \lambda_k$. Let $\epsilon > 0$ be a sufficiently small value. We first introduce the following $2|S^+|$ affinely independent points, where first $|S^+|$ points are represented as $\bar{z}$ and second half are represented by $\hat{z}$.

i) Let $\bar{z}^1 = \tilde{z}$ and let $t \in S_j^+ \setminus \{1\}$. Point $\bar{z}^t$ is obtained by sending extra flow amount ϵ from arc 1 and ϵ less from arc t. Then, $\bar{z}^t$ is described by $\bar{y}_1^t = \bar{y}_1^1 + \epsilon$, $\bar{y}_t^t = \bar{y}_t^1 - \epsilon$, $\bar{y}_m^t = \bar{y}_m^1$ for $m \in A \setminus (I \cup \{1, t\})$, $\bar{i}_s = \bar{i}_s^1 + \epsilon$ for $s = k, \ldots, j-1$ and $\bar{x}^t = \bar{x}^1$.

ii) Let $t \in S_j^+$, point $\hat{z}^t$ is obtained by closing arc t. In order to satisfy inequality (2.10) at equality, we need to send $\tilde{y}_t^1 - (c_t - \lambda_j)^+$ extra amount of flow from arcs in $t \in K \setminus \{t\}$ starting from $\bar{z}^1$. Validity of this operation is guaranteed by the feasibility of $v(S^+ \setminus \{t\}, L^-)$ with value $d_{k\ell} - (c_t - \lambda_j)^+$. Then, $\hat{x}^t_m = 1$ for $m \in S^+ \setminus \{t\}$ and $\hat{x}^t_m = 0$ for $m \in (A \setminus I) \cup \{t\}$. Path flows are then calculated using flow balance: $\hat{i}^t_j = \left(\hat{i}^t_{j-1} + \sum_{m \in S^+_j} \hat{y}^t_m - d_j\right)^+$.

Next we describe $2|E^- \setminus S^+|$ points represented by z and z'.

iii) Let $t \in E^+_j \setminus S^+_j$, the point z^t is obtained by opening arc t. $y^t = \tilde{y}$, $x^t_n = 1$ for $n \in S^+ \setminus \{m\}$, $x^t_n = 0$ otherwise and $i^t = \tilde{i}$.

iv) We now send a flow of ϵ from arc $t \in E^+_j \setminus S^+_j$. Let $y^{t\prime}_m = \tilde{y}_m$ if $m \neq t$ and $y^{t\prime}_t = \tilde{y}_t$. Also let, $x^{t\prime} = x^t$ and $i^{t\prime} = \tilde{i}$.

Next, we define the affinely independent points $\bar{w}^t$ and $\hat{w}^t$ described for arcs $t \in L^-$.

v) Point $\bar{w}^t$ is obtained by opening the arc $t \in L^-_j$ and setting $\bar{x}^t_t = 1$. Starting with feasible point $\tilde{z}$, we send an extra flow of λ_j from arcs in $K \cap S^+_i$ for $i = k, \ldots, j$. This operation conserves feasibility and satisfies the inequality (2.10) at equality because $v(S^+, L^- \setminus \{t\}, S^-)$ is feasible with a value of $d_{k\ell} - \lambda_j$. Extra incoming flow of λ_j is sent from outgoing arc t resulting $\bar{y}^t_t = \lambda_j$. Let $\bar{x}^t_m = 1$ for $m \in S^+ \cup \{t\}$ and 0 otherwise. At points $\bar{w}$, the flow balance constraints hold at equality. Therefore, $\bar{i}^t_j = \left(\bar{i}^t_{j-1} + \sum_{m \in S^+_j} \bar{y}^t_m - d_j\right)^+$.

vi) Let $\hat{w}^t$ have similar properties to $\bar{w}^t$ and let $t \in L^-_j$. The second set of feasible points have $\hat{x}^t = \bar{x}^t$, $\hat{i}^t = \bar{i}^t$, $\hat{y}^t_m = \bar{y}^t_m$ for $m \neq t$ and $\hat{y}^t_t = \lambda_j + \epsilon$.

We now construct the affinely independent points corresponding to arcs $t \in E^- \setminus L^-$.

vii) Starting with feasible point $\tilde{z}$, we obtain point w by opening arc $t \in E^- \setminus L^-$. Let $y^t = \tilde{y}$, $x^t_m = \tilde{x}_m$ for $m \neq t$ and $x^t_t = 1$, $i^t = \tilde{i}$.

viii) Let $t \in L^-_j$. Starting with $\tilde{z}$, we send an extra flow of ϵ from arc 1, $y^{t\prime}_1 = \tilde{y}_1 + \epsilon$, $y^{t\prime}_t = \epsilon$ and $y^{t\prime}_m = \tilde{y}_m$ for $m \neq t, m \neq 1$. Let $x^{t\prime} = x^t$ and $i^{t\prime}_s = \tilde{i}_s + \epsilon$ for $s = k, \ldots j - 1$.

Affinely independent points corresponding to path arcs are represented by v^j and defined below.

ix) Prior to giving the feasible points, we first define a new notion called *minimal cover blocks*. Suppose we have a subpath $B \subseteq M$ with cardinality $|B| = b$ and nodes $B = \{n_1, \ldots, n_b\}$, where without loss of generality n_1 and n_b are the first and the last nodes of subpath B respectively. We say that B is a minimal cover block if (i) $\cup_{j \in B} S^+_j$ is a path cover for B, (ii) $\cup_{j \in B \setminus \{n_1\}} S^+_j$ is not a path cover for $B \setminus \{n_1\}$ and (iii) if $b > 1$, then $S^+_{n_b}$ is not a cover for $\{n_b\}$. Notice that V can be partitioned into a number of minimal cover blocks $B_1, \ldots, B_q$ where $V = \cup_{i=1}^q B_i$.

Let n_1^i be the first node of block B_i and $m^i = K \cap S^+_{n_1^i}$ be the arc with largest capacity incoming to node n_1^i. Moreover, let us call nodes in $B_i = \{n_1^i, \ldots, n_b^i\}$. While finding the point associated with path arc incoming to node j where $j \in B_i$, we start from the feasible point where arc m^i is closed, $\hat{z}^{m^i}$. For each path arc $p = n_1^i - 1, \ldots, n_b^i - 1$, send an extra flow of ϵ from path arcs $s = p, \ldots, n_b^i - 1$. We, then, describe point v^j as: $x^j = \hat{x}^m$, $y^j = \hat{y}^m$ and $i_s^j = \hat{i}_s^m + \epsilon$ for $s = p, \ldots, n_b^i - 1$.

□

B.3 Proof of Theorem 2.10

We do the proof case by case. For each interval, we construct a solution that follows the optimality conditions provided in Section 2.3. We show the feasibility of the solution explicitly only under Case 1, since the procedure is the same for all the cases.

Case 1: $M_{j,i} - \bar{\lambda}_j \leq z \leq M_{j,i} - \bar{\lambda}_j + \lambda_j, \qquad i \in [1, p_j - 1]$.

If $v_l \in S_j^{++} \setminus \{v_{p_j}\}$, then $y_{v_k} = 0$ for $l \leq i$ and $y_{v_l} = c_{v_l}$ for $l > i$, $y_{v_{p_j}} = c_{v_{p_j}} - z + M_{j,i} - \bar{\lambda}_j$. If $v_l \in L_j^-$, then $y_{v_l} = c_{v_l}$ for $l \leq i$, $y_{v_l} = 0$ for $l > i$ and $w = m_j$. Because of the boundaries of z we have,

$$c_{v_{p_j}} - \lambda_j \leq y_{v_{p_j}} \leq c_{v_{p_j}}.$$

Therefore, this solution agrees with the optimality conditions described above.

The solution is also feasible because constraint (2.38) is:

$$w + \sum_{t \in S_j^{++}} y_t + \sum_{t \in L^-} (c_t - y_t) = m_j + M_{j,r_j} - M_{j,i} - z + M_{j,i} - \bar{\lambda}_j = \hat{d}_j - z.$$

The objective function value of this solution is:

$$\begin{aligned}
&\hat{d}_j - m_j + \sum_{t \in L_j^-} \lambda_j x_t + \sum_{t \in S_j^{++}} \lambda_j (1 - x_t) - \sum_{t \in S_j^{++}} (y_t + c_t(1 - x_t)) - \sum_{t \in L_j^-} c_t \\
&= \hat{d}_j - m_j + i\lambda_j - M_{j,r} + z - M_{j,i} + \bar{\lambda}_j \\
&= -\bar{\lambda}_j + i\lambda_j + z - M_{j,i} + \bar{\lambda}_j \\
&= z + i\lambda_j - M_{j,i}
\end{aligned}$$

Case 2: $M_{j,i} + \lambda_j - \bar{\lambda}_j \leq z \leq M_{j,i+1} - \bar{\lambda}_j, \qquad i \in [0, p_j - 1]$.

If $v_l \in S_j^{++}$, then $y_{v_l} = 0$ for $l \leq i + 1$, $y_{v_l} = c_{v_l}$ for $l > i + 1$. If $v_l \in L_j^-$, then $y_{v_l} = c_{v_l}$ for $l \leq i + 1$ and $y_{v_l} = 0$ for $l > i + 1$ and $w = m_j$.

Case 3: $M_{j,i} - \bar{\lambda}_j \leq z \leq M_{j,i} - \bar{\lambda}_j + ml_j, \qquad i \in [p_j, r_j - 1].$

If $v_l \in S_j^{++}$, then $y_{v_l} = 0$. If $v_l \in L_j^-$, then $y_{v_l} = c_{v_l}$ for $l \leq i$, $y_{v_l} = 0$ for $l > i$ and $w = m_j + M_{j,i} - \bar{\lambda}_j - z$.

Case 4: $M_{j,i} - \bar{\lambda}_j + ml_j < z \leq M_{j,i} - \bar{\lambda}_j + ml_j + \varphi_{j,i}, \qquad i \in [p_j, r_j - 1].$

If $v_l \in S_j^{++} \setminus \{v_{p_j}\}$, then $y_{v_l} = 0$, $y_{v_{p_j}} = M_{j,i+1} - \bar{\lambda}_j - z$. If $v_l \in L_j^-$, then $y_{v_l} = c_{v_l}$ for $l \leq i+1$ and $y_{v_l} = 0$ for $l > i+1$ and $w = m_j$.

Case 5: $M_{j,i} - \bar{\lambda}_j + ml_j + \varphi_{j,i} \leq z \leq M_{j,i+1} - \bar{\lambda}_j, \qquad i \in [p_j, r_j - 1].$

If $v_l \in S_j^{++}$, then $y_{v_l} = 0$. If $v_l \in L_j^-$, then $y_{v_l} = c_{v_l}$ for $l \leq i+1$ and $y_{v_l} = 0$ for $l > i+1$ and $w = m_j$.

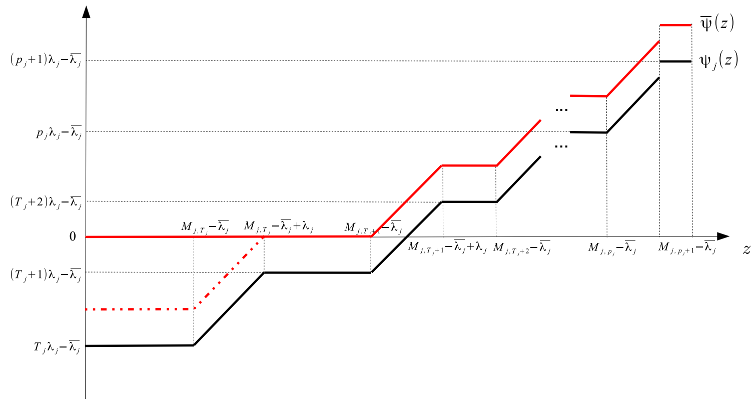
Case 6: $M_{j,r} - \bar{\lambda}_j \leq z \leq \hat{d}_j$.

If $v_l \in S_j^{++}$, then $y_{v_l} = 0$. If $v_l \in L_j^-$, then $y_{v_l} = c_{v_l}$ and $w = m_j + M_{j,r_j} - \bar{\lambda}_j - z$.

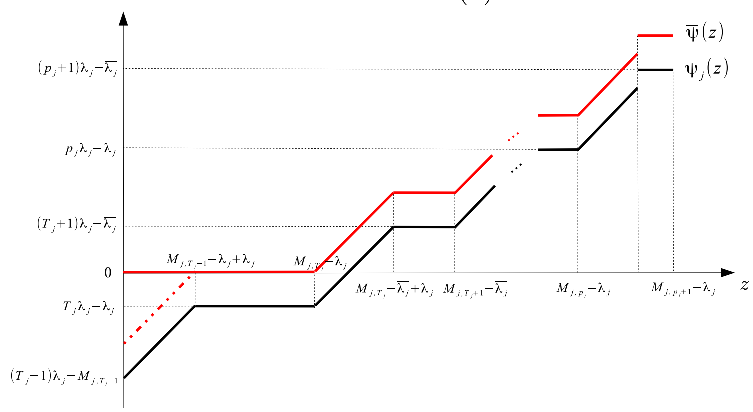
□

B.4 Proof of Theorem 2.11

Figures (B.1a) and (B.1b) represent the two different forms that $\psi_j(z)$ can take if $T_j\lambda_j - \bar{\lambda}_j \leq 0$.



(a)



(b)

Figure B.1: Two different representations of the functions $\bar{\phi}(z)$ and $\phi_j(z)$.

In Figures B.1a and B.1b, the function $\bar{\psi}(z)$ is a superadditive function that conforms to the description in Theorem 13 of Gu et al. (1999). Then, using shifting down argument in Observation 3 in Appendix B.5 and changing a nonnegative piece of the function with non-positive values argument in Observation 4 that is also in Appendix B.5, one can easily verify that $\psi_j(z)$ is superadditive when $T_j\lambda_j - \bar{\lambda}_j \leq 0$. $\square$

B.5 Observations on superadditive functions

Observation 3. If function $f(z)$ is superadditive for $z \in Z$ and $K \leq 0$, then $g(z) = K + f(z)$ is also superadditive for $z \in Z$. In other words, shifting a superadditive function downwards preserves superadditivity.

Proof. Let z_1, z_2 and $z_1 + z_2 \in Z$. We know that $f(z_1) + f(z_2) \leq f(z_1 + z_2)$. Since $K \leq 0$, summing left side with $2 \times K$ and right side with K, ensures that the following inequality holds:

$$K + f(z_1) + K + f(z_2) \leq K + f(z_1 + z_2)$$

$$\implies g(z_1) + g(z_2) \leq g(z_1 + z_2).$$

□

Note that if a superadditive function is summed with a positive constant K, then we need to ensure that

$$K + \max\{g(z_1) + g(z_2) - g(z_1 + z_2) \ : \ z_1, z_2, z_1 + z_2 \in Z\} \leq 0$$

in order to conclude that the resulting function is superadditive as well.

Observation 4. Let $f(x)$ be superadditive and non-positive on $0 \leq x \leq k$ for some $k \geq 0$ and $g(x) \geq 0$ be non-decreasing and superadditive on $x \geq k$. Let $f(k) = g(k) = 0$. Then, the function

$$\phi(x) = \begin{cases} f(x) & 0 \leq x \leq k \\ g(x) & x \geq k \end{cases}$$

is superadditive.

Proof. There are four cases we examine:

Case 1: $0 \leq x_1 \leq k$, $0 \leq x_2 \leq k$ and $x_1 + x_2 \leq k$. Then,

$$\phi(x_1) + \phi(x_2) - \phi(x_1 + x_2) = f(x_1) + f(x_2) - f(x_1 + x_2) \leq 0$$

because f is superadditive on $[0, k]$.

Case 2: $0 \leq x_1 \leq k$, $0 \leq x_2 \leq k$ and $x_1 + x_2 > k$. Then,

$$\phi(x_1) + \phi(x_2) - \phi(x_1 + x_2) = f(x_1) + f(x_2) - g(x_1 + x_2) \leq 0$$

because $f(x_1) \leq 0$, $f(x_2) \leq 0$ and $g(x_1 + x_2) \geq 0$.

Case 3: $0 \leq x_1 \leq k$, $x_2 > k$ and $x_1 + x_2 > k$. Then,

$$\phi(x_1) + \phi(x_2) - \phi(x_1 + x_2) = f(x_1) + g(x_2) - g(x_1 + x_2) \leq 0$$

since $g(x_2) - g(x_1 + x_2) \leq 0$ and $f(x_1) \leq 0$.

Case 4: $x_1 > k$, $x_2 > k$ and $x_1 + x_2 > k$. Then,

$$\phi(x_1) + \phi(x_2) - \phi(x_1 + x_2) = g(x_1) + g(x_2) - g(x_1 + x_2) \leq 0$$

due to the superadditivity of the function g.

□

Observation 5. If $f(x) : \mathbb{R}^n \to \mathbb{R}^m$ is a superadditive non-decreasing function on $\mathbb{R}^n_+$, then $g(x) = \max\{0, f(x)\}$ is a superadditive function as well.

Proof. We would like to show that

$$\max\{f(x_1 + x_2), 0)\} \leq \max\{f(x_1), 0\} + \max\{f(x_2), 0\}. \tag{B.2}$$

Without loss of generality, we assume $x_1 \leq x_2$, then, there are three cases we need to examine:

Case 1: $f(x_1 + x_2) \leq 0$: We have $f(x_1) \leq 0$ and $f(x_2) \leq 0$ since f is non-decreasing. Then, both sides of inequality (B.2) are zero.

Case 2: $f(x_1 + x_2) > 0$, $f(x_1) \leq 0$ and $f(x_2) \geq 0$: Inequality (B.2) becomes $f(x_1 + x_2) \geq f(x_2)$ which holds since f is non-decreasing.

Case 3: $f(x_1 + x_2) > 0$, $f(x_1) \geq 0$ and $f(x_2) \geq 0$: Inequality (B.2) becomes $f(x_1 + x_2) \geq f(x_1) + f(x_2)$ which holds due to superadditivity of f.

□

B.6 Convex lower-bound of $f_j^L(z)$

Largest convex lower-bound function of $f_j^L(z)$ with a nonpositive value at origin is a piecewise linear function with the following generic form:

$$\phi_j(z) = \begin{cases} \frac{\tau\lambda_j - \bar{\lambda}_j - \Gamma}{M_{j,\tau}\bar{\lambda}_j} z + \Gamma & 0 \leq z \leq M_{j,\tau} - \bar{\lambda}_j \\ \frac{\lambda_j}{c_{j_{i+1}}}(z - M_{j,i} + \bar{\lambda}_j) + i\lambda_j - \bar{\lambda}_j & M_{j,i} - \bar{\lambda}_j \leq z \leq M_{j,i+1} - \bar{\lambda}_j \qquad i = \tau, \ldots, r_j - 1 \\ z - M_{j,r_j} + r_j\lambda_j & M_{j,r_j} - \bar{\lambda}_j \leq z \leq \hat{d}_j. \end{cases}$$

Recall that index T_j is defined as $\min\{1 \leq i \leq r_j \ : \ M_{j,i} - \bar{\lambda}_j \geq 0\}$ if $M_{j,r_j} - \bar{\lambda}_j \geq 0$, and r_j otherwise. Moreover, for notational convenience we let $c_{v_{r_j+1}} := \lambda_j$. Next, we find values of Γ and τ explicitly for cases where $T_j\lambda_j - \bar{\lambda}_j > 0$.

Case 1: $T_j \leq p_j$.

Case 1.1: $M_{j,T_j-1} - \bar{\lambda}_j + \lambda_j \leq 0$.

In this case $f_j^L(z)$ has the form in Figure B.2.

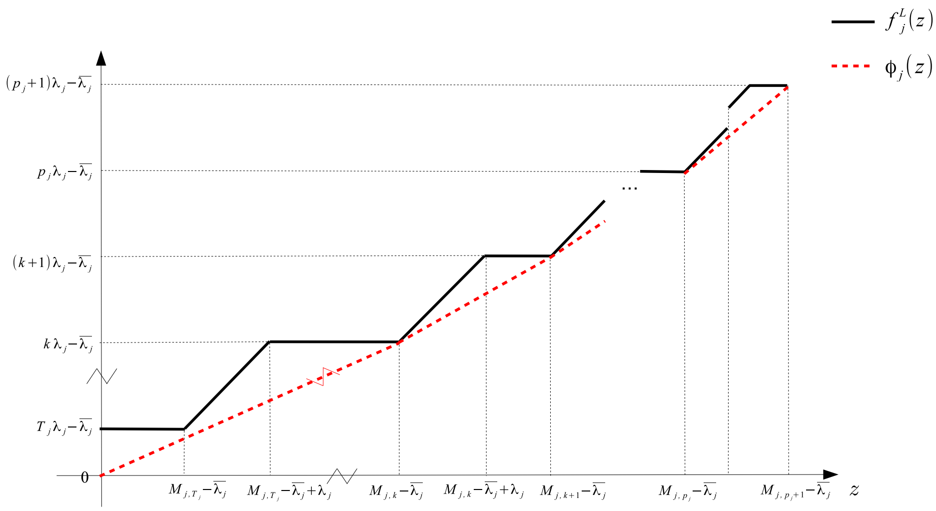


Figure B.2: Representation of Case 1.1.

We select index τ to be the minimum such that the piecewise linear function has an increasing slope. Then,

$$\tau = \min\left\{T_j \le i \le r_j \;:\; \frac{i\lambda_j - \bar{\lambda}_j}{M_{j,i} - \bar{\lambda}_j} \le \frac{\lambda_j}{c_{v_{i+1}}}\right\}$$

and $\Gamma = 0$.

Case 1.2: $M_{j,T_j-1} - \bar{\lambda}_j + \lambda_j > 0$.

The function $f_j(z)$ under this conditions has the form in Figure B.3.

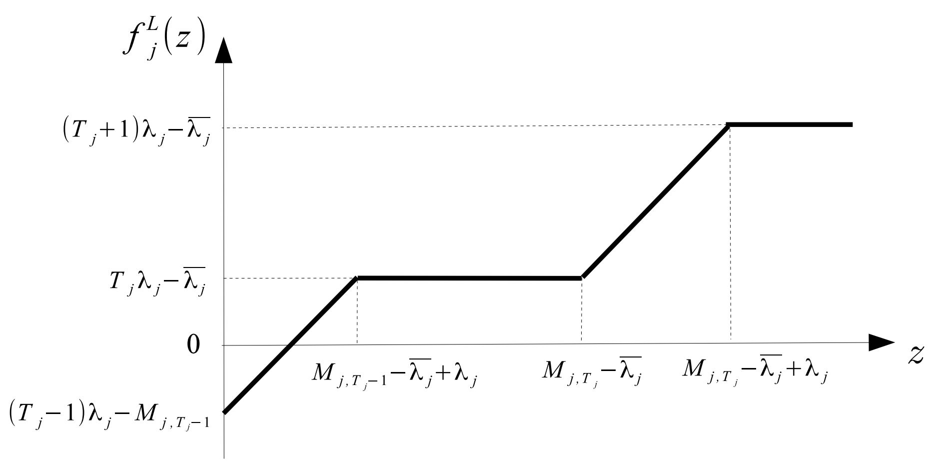


Figure B.3: Representation of Case 1.2.

The largest convex lower bound of $f_j^L(z)$ that satisfies $\phi_j(0) \le 0$ is its convex envelope. We have

$$\tau = \min\left\{T_j \leq i \leq r_j \ :\ \frac{M_{j,t-1} - \bar{\lambda}_j + (i - T_j + 1)\lambda_j}{M_{j,i} - \bar{\lambda}_j} \leq \frac{\lambda_j}{c_{v_{i+1}}}\right\}$$

and $\Gamma = (T_j - 1)\lambda_j - M_{j,T_j-1}$.

Case 2: $T_j > p_j$.

Case 2.1: $M_{j,T_j-1} - \bar{\lambda}_j + ml_j + \varphi_{j,T_j-1} \leq 0$ This is very similar to Case 1.1. Please see Figure B.4 for a representation of $f_j^L(z)$ and it's convex lower bound.

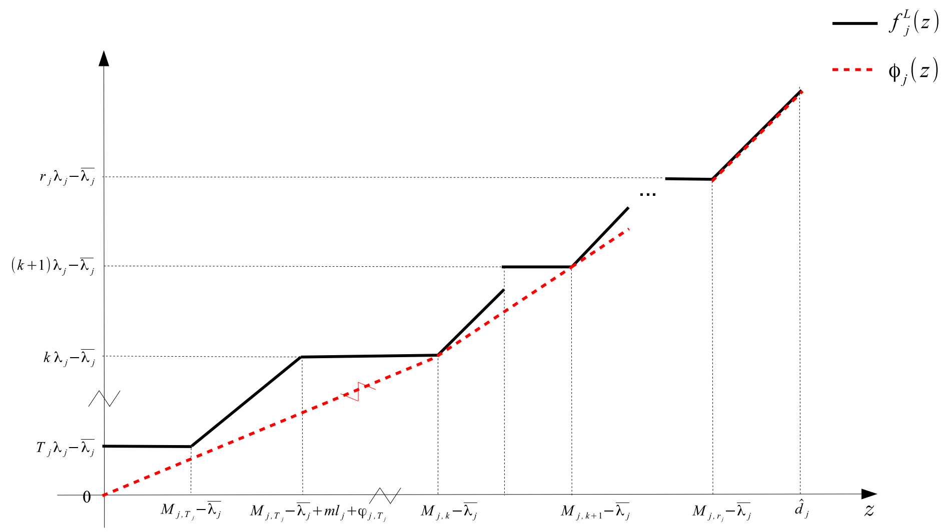


Figure B.4: Representation of Case 2.1.

Values of τ and Γ are the same as in Case 1.1.

Case 2.2: $M_{j,T_j-1} - \bar{\lambda}_j + ml_j + \varphi_{j,T_j-1} > 0$.
The closed form of convex lower bound depends on the following two sub cases.

Case 2.2.1: $M_{j,T_j-1} - \bar{\lambda}_j + ml_j \leq 0$.
Please see Figure B.5 for a representation of this case.

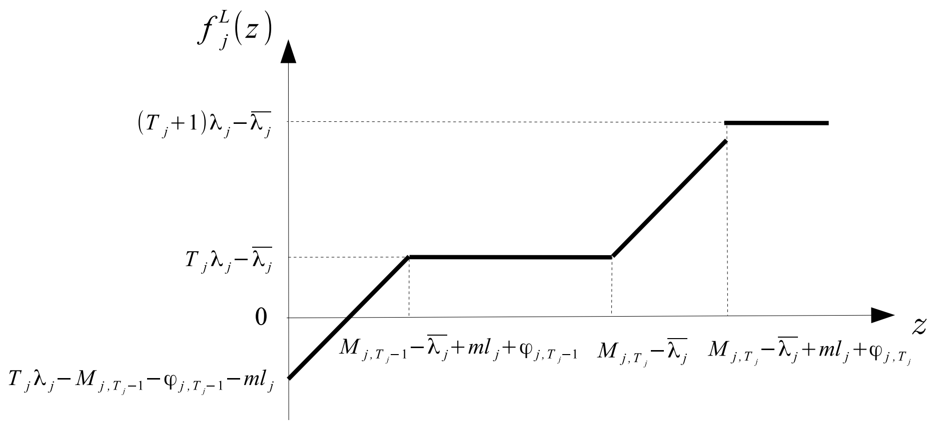


Figure B.5: Representation of Case 2.2.1.

The largest convex lower bound that satisfies $\phi_j(0) \leq 0$ is the convex envelope of $f_j^L(z)$. We have

$$k = \min\left\{T_j \leq i \leq r_j \ : \ \frac{M_{j,T_j-1} - \bar{\lambda}_j + ml_j + \varphi_{j,T_j-1} + (i-T_j)\lambda_j}{M_{j,i} - \bar{\lambda}_j} \leq \frac{\lambda_j}{c_{v_{i+1}}}\right\}$$

and $a = T_j\lambda_j - \varphi_{j,T_j-1} - ml_j - M_{j,T_j-1}$.

Case 2.2.2: $M_{j,T_j-1} - \bar{\lambda}_j + ml_j > 0$.
The representation of $f_j^L(z)$ under this case can be seen in Figure B.6. The convex envelope of the function satisfies $\phi_j(0) \leq 0$. Therefore, it is superadditive. Values of τ and Γ are the same as in Case 1.2.

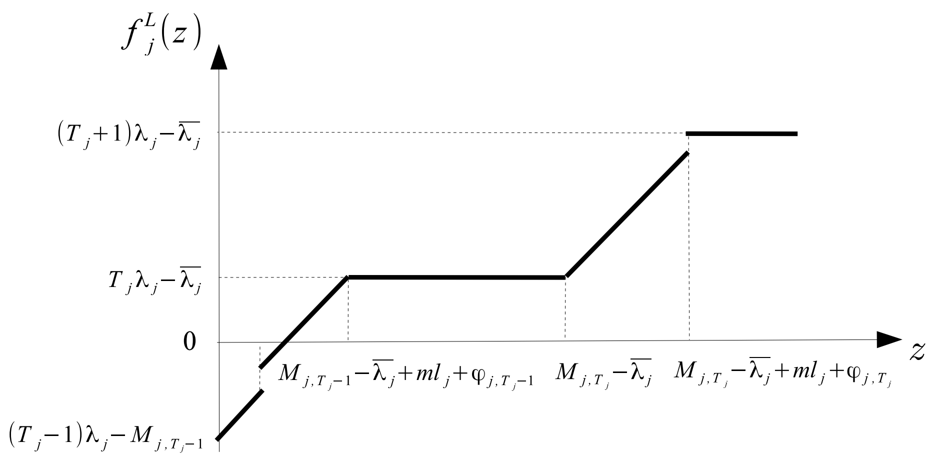


Figure B.6: Representation of Case 2.2.2.

Appendix C

Equivalency of (F3.2) to the maximum flow problem

In Section 3.2, we showed the maximum flow equivalency of $v(S^+, L^-)$ under the assumption that $d_j \geq 0$ for all $j \in N$. In this section, we generalize the equivalency for the paths where $d_j < 0$ for some $j \in N$.

Observation 6. If $d_j < 0$ for some $j \in N$, one can represent the supply amount as a dummy arc incoming to node j (i.e., added to E_j^+) with a fixed flow and capacity of $-d_j$ and set the modified demand of node j to be $d_j = 0$.

Given the node set N with at least one supply node, let $\mathcal{T}(N)$ be the transformed path using Observation 6. Transformation $\mathcal{T}$ ensures that the dummy supply arcs are always open. As a result, they are always in the set S^+. We refer to the additional constraints that fix the flow to the supply value on dummy supply arcs as *fixed-flow constraints*. Notice that, $v(S^+, L^-)$ computed for $\mathcal{T}(N)$ does not take fixed-flow constraints into account. In the next proposition, for a path structure, we show that there exists at least one optimal solution to (F3.2) such that the fixed-flow constraints are satisfied.

Proposition C.1. Suppose that $d_j < 0$ for some $j \in N$. If (F3.2) for the node set N is feasible, then it has at least one optimal solution that satisfies the fixed-flow constraints.

Proof. We need to show that $v(S^+, L^-)$ has an optimal solution where the flow at the dummy supply arcs is equal to the supply values. The transformation $\mathcal{T}$ makes Proposition 3.1 applicable to the modified path $\mathcal{T}(N)$. Let $\mathcal{Y}$ be the set of optimal solutions of (F3.2). Then, there exists a solution $(\mathbf{y}^*, \mathbf{i}^*, \mathbf{r}^*) \in \mathcal{Y}$ where $y_t^* = 0$ for $t \in E^- \setminus (S^- \cup L^-)$. Let $p \in S_j^+$ represent the index of the dummy supply arc with $c_p = -d_j$. If $y_p^* < c_p$, then satisfying the fixed-flow constraints require pushing flow through the arcs in $E^- \setminus L^-$. We use Algorithm 4 to construct an optimal solution with $y_p^* = c_p$.

Note that each arc in $E_k^- \setminus L_k^-$ for $k \in N$ appear in (F3.2) with the same coefficients, therefore we merge these outgoing arcs into one in Algorithm 4. We represent the merged flow and capacity by $\bar{Y}_k^- = \sum_{t \in E_k^- \setminus (S_k^- \cup L_k^-)} y_t^*$ and $\bar{C}_k = c\big(E_k^- \setminus (S_k^- \cup L_k^-)\big)$ for $k \in N$. □

Algorithm 4

$\mathcal{J}$: Set of supply nodes in N where the nodes are sorted with respect to their order in N.
$(\mathbf{y}^*, \mathbf{i}^*, \mathbf{r}^*) \in \mathcal{Y}$: $y_t^* = 0$ for all $t \in E^-$.
for $q \in \mathcal{J}$ **do**
  Let p be the dummy supply arc in S_q^+
  $\Delta = c_p - y_p^*$
  for $j = q$ **to** n **do**
    $\bar{Y}_j^- = \bar{Y}_j^- + \min\{\bar{C}_j - \bar{Y}_j^-, \Delta\}$
    $\Delta = \Delta - \min\{\bar{C}_j - \bar{Y}_j^-, \Delta\}$
    $i_j^* = i_j^* + \Delta$
    if $i_j^* > u_j$ **then**
      $\Delta = i_j^* - u_j$
      $i_j^* = u_j$
      Let $k := j$
      break inner loop
    end if
  end for
  if $\Delta > 0$ **then**
    for $j = k$ **to** 1 **do**
      $\bar{Y}_j^- = \bar{Y}_j^- + \min\{\bar{C}_j - \bar{Y}_j^-, \Delta\}$
      $\Delta = \Delta - \min\{\bar{C}_j - \bar{Y}_j^-, \Delta\}$
      $r_j^* = r_j^* + \Delta$
      if $r_j^* > b_j$ **then**
        $\Delta = r_j^* - b_j$
        break inner loop
      end if
    end for
  end if
  if $\Delta > 0$ **then**
    (F3.2) is infeasible for the node set N.
  end if
end for

Proposition C.1 shows that, under the presence of supply nodes, transformation $\mathcal{T}$ both captures the graph's structure and does not affect (F3.2)'s validity. As a result, Propositions 3.1 and 3.2 become relevant to the transformed path and submodular path inequalities (3.14) and (3.17) are also valid for paths where $d_j < 0$ for some $j \in N$.

Appendix D

Proofs from Chapter 3

D.1 Proof of Lemma 3.3

Recall that $C = S^+ \cup L^-$ and let $C_1 = S^+_{N1} \cup L^-_{N1}$ and $C_2 = S^+_{N2} \cup L^-_{N2}$. In (3.13), we showed that the value of the minimum cut is

$$v(C) = m_i = \min\{\alpha^u_i + \beta^u_i - c(S^+_i), \alpha^d_i + \beta^d_i - d_i - c(S^-_i)\}$$

for all $i \in N$. For node set N_1 and the arc set C_1, the value of the minimum cut is

$$v_1(C_1) = \min\{\alpha^u_{j-1} + b_{j-1}, \alpha^d_{j-1}\}.$$

This is because of three observations: (1) the values $\alpha^{\{u,d\}}_i$ for $i \in [1, j-2]$ are the same for the node sets N_1 and N, (2) for the arc set C_1 the set S^+_{j-1} now includes the backward path arc $(j, j-1)$ and (3) node $j-1$ is the last node of the first path. Similarly, for node set N_2 and the arc set C_2, the value of the minimum cut is

$$v_2(C_2) = \min\{\beta^u_j + u_{j-1}, \beta^d_j\}.$$

For nodes N_2 and the arc set C_2, (1) the values $\beta^{\{u,d\}}_i$ for $i \in [j+1, n]$ are the same for the node sets N_2 and N, (2) for the arc set C_2 the set S^+_j now includes the forward path arc $(j-1, j)$ and (3) node j is the first node of the second path.

Now, if $\alpha^u_j = \alpha^d_{j-1} + u_{j-1} + c(S^+_j)$, then $\alpha^d_j = \alpha^d_{j-1} + d_j + c(S^-_j)$ from equations in (4.3)–(4.4). Then, rewriting $v(C) = m_j$ and $v_1(C_1)$ in terms of α^d_{j-1}:

$$v(C) = \alpha^d_{j-1} + \min\{\beta^u_j + u_{j-1}, \beta^d_j\}$$

and

$$v_1(C_1) = \alpha^d_{j-1}.$$

As a result, the values $v_1(C_1)$ and $v_2(C_2)$ summed gives the value $v(C)$ under the assumption for the value of α_j^u.

Similarly, if $\beta_{j-1}^u = \beta_j^d + b_{j-1} + c(S_{j-1}^+)$, then $\beta_{j-1}^d = \beta_j^d + d_{j-1} + c(S_{j-1}^-)$ from equations in (4.5)–(4.6). Then, rewriting $v(C) = m_{j-1}$ and $v_2(C_2)$ in terms of β_j^d:

$$v(C) = \beta_j^d + \min\{\alpha_{j-1}^u + b_{j-1}, \alpha_{j-1}^d\}$$

and

$$v_2(C_2) = \beta_j^d.$$

As a result, the values $v_1(C_1)$ and $v_2(C_2)$ summed gives the value $v(C)$ under the assumption for the value of β_{j-1}^u.

D.2 Proof of Lemma 3.4

The proof follows closely to that of Lemma 3.3. Let $C = S^+ \cup L^-$, $C_1 = S_{N1}^+ \cup L_{N1}^-$ and $C_2 = S_{N2}^+ \cup L_{N2}^-$. For node set N_1 and the arc set C_1, the value of the minimum cut is

$$v_1(C_1) = \min\{\alpha_{j-1}^u, \alpha_{j-1}^d + u_{j-1}\},$$

where u_{j-1} is added because $c(S_{N1}^-) = c(S_{1j-1}^-) + u_{j-1}$. Similarly, for node set N_2 and the arc set C_2, the value of the minimum cut is

$$v_2(C_2) = \min\{\beta_j^u, \beta_j^d + b_{j-1}\},$$

where b_{j-1} is added because $c(S_{N2}^-) = c(S_{jn}^-) + b_{j-1}$.

Now, if $\alpha_j^d = \alpha_{j-1}^u + b_{j-1} + d_{j-1} + c(S_j^-)$, then $\alpha_j^u = \alpha_{j-1}^u + c(S_j^+)$ from equations in (4.3)–(4.4). Then, rewriting $v(C) = m_j$ and $v_1(C_1)$ in terms of α_{j-1}^u:

$$v(C) = \alpha_{j-1}^u + \min\{\beta_j^u, \beta_j^d + b_{j-1}\}$$

and

$$v_1(C_1) = \alpha_{j-1}^u.$$

As a result, the values $v_1(C_1)$ and $v_2(C_2)$ summed gives the value $v(C)$ under the assumption for the value of α_j^d.

Similarly, if $\beta_{j-1}^d = \beta_j^u + u_{j-1} + d_{j-1} + c(S_{j-1}^-)$, then $\beta_{j-1}^u = \beta_j^u + c(S_{j-1}^+)$ from equations in (4.5)–(4.6). Then, rewriting $v(C) = m_{j-1}$ and $v_2(C_2)$ in terms of β_j^u:

$$v(C) = \beta_j^u + \min\{\alpha_{j-1}^u, \alpha_{j-1}^d + u_{j-1}\}$$

and

$$v_2(C_2) = \beta_j^u.$$

As a result, the values $v_1(C_1)$ and $v_2(C_2)$ summed gives the value $v(C)$ under the assumption for the value of β^d_{j-1}.

D.3 Proof of Lemma 3.5

The proof follows closely to that of Lemmas 3.3 and 3.4. Let $C = S^+ \cup L^-$, $C_1 = S^+_{N1} \cup L^-_{N1}$ and $C_2 = S^+_{N2} \cup L^-_{N2}$. For node set N_1 and the arc set C_1, the value of the minimum cut is

$$v_1(C_1) = \min\{\alpha^u_{j-1}, \alpha^d_{j-1}\}$$

and for node set N_2 and the arc set C_2, the value of the minimum cut is

$$v_2(C_2) = \min\{\beta^u_j + u_{j-1}, \beta^d_j + b_{j-1}\}.$$

Now, if $\alpha^u_j = \alpha^d_{j-1} + u_{j-1} + c(S^+_j)$ and $\beta^d_{j-1} = \beta^u_j + u_{j-1} + d_{j-1} + c(S^-_{j-1})$, then $\alpha^d_j = \alpha^d_{j-1} + d_j + c(S^-_j)$ and $\beta^u_{j-1} = \beta^u_j + c(S^+_j)$. Then, rewriting $v(C) = m_j$, $v_1(C_1)$ and $v_2(C_2)$:

$$v(C) = \alpha^d_{j-1} + \min\{u_{j-1} + \beta^u_j, \beta^d_j\} = \alpha^d_{j-1} + u_{j-1} + \beta^u_j,$$

$$v_1(C_1) = \alpha^d_{j-1} \text{ and } v_2(C_2) = \beta^u_j + u_{j-1}.$$

As a result, the values $v_1(C_1)$ and $v_2(C_2)$ summed gives the value $v(C)$ under the assumption for the values of α^u_j and β^d_{j-1}.

D.4 Proof of Lemma 3.6

The proof follows closely to that of Lemmas 3.3 and 3.4. Let $C = S^+ \cup L^-$, $C_1 = S^+_{N1} \cup L^-_{N1}$ and $C_2 = S^+_{N2} \cup L^-_{N2}$. For node set N_1 and the arc set C_1, the value of the minimum cut is

$$v_1(C_1) = \min\{\alpha^u_{j-1} + b_{j-1}, \alpha^d_{j-1} + u_{j-1}\}$$

and for node set N_2 and the arc set C_2, the value of the minimum cut is

$$v_2(C_2) = \min\{\beta^u_j, \beta^d_j\}.$$

Now, if $\alpha^d_j = \alpha^u_{j-1} + b_{j-1} + d_j + c(S^-_j)$ and $\beta^u_{j-1} = \beta^d_j + b_{j-1} + c(S^+_{j-1})$, then $\alpha^u_j = \alpha^u_{j-1} + c(S^+_j)$ and $\beta^d_{j-1} = \beta^d_j + d_j + c(S^-_j)$. Then, rewriting $v(C) = m_j$, $v_1(C_1)$ and $v_2(C_2)$:

$$v(C) = \alpha^u_{j-1} + \min\{\beta^u_j, \beta^d_j + b_{j-1}\} = \alpha^u_{j-1} + \beta^d_j + b_{j-1},$$

$$v_1(C_1) = \alpha^u_{j-1} + b_{j-1} \text{ and } v_2(C_2) = \beta^d_j.$$

As a result, the values $v_1(C_1)$ and $v_2(C_2)$ summed gives the value $v(C)$ under the assumption for the values of α_j^d and β_{j-1}^u.

Appendix E

Lot Sizing with Inventory Bounds and Fixed Costs

In this section, we show that the inequalities by Atamtürk and Küçükyavuz (2005) is a special case of (4.9). Recall that they formulate the lot-sizing with bounded inventory (LSBI) problem as the following optimization problem:

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{n}(f_t x_t + p_t y_t + g_t z_t + h_t i_t) \\
\text{s.t.} \quad & i_{t-1} + y_t - i_t = d_t, \quad t \in [1,n], \\
& 0 \le i_t \le u_t z_t, \quad t \in [1,n] \\
\text{(LSBI)} \quad & 0 \le y_t \le (d_t + u_t) x_t, \quad t \in [1,n], \\
& y \in \mathbb{R}^n, \quad x \in \{0,1\}^n, \\
& i \in \mathbb{R}^{n+1}, \quad z \in \{0,1\}^{n+1}.
\end{aligned}
$$

In the first set of inequalities, they give valid inequalities for LSBI, when $z_t = 1$, for all $t \in [1.n]$. Let $[k,\ell] \subseteq [1,n]$ be a subset of nodes and $L \subseteq [k,\ell]$ be a subset of production arcs. Then the inequalities

$$
i_{k-1} + \sum_{t \in L^+} y_t \le u_{k-1} + \sum_{t \in L^+} \min\{d_{kt} + u_t - u_{k-1}, d_{k\ell} - u_{k-1}, d_{t\ell}\} x_t + i_\ell, \tag{E.1}
$$

$$
i_{k-1} + \sum_{t \in L^+} y_t \le u_{k-1} + \sum_{t \in L^+} (d_{kt} - u_{k-1} + u_t) x_t \tag{E.2}
$$

are valid for LSBI.

Proposition E.1. Inequality (E.1) is weaker than inequality (4.9) for LSBI, where $S^+ = \{(k-1,k)\}$.

Proof. Let $p = \min\{t \in [k,\ell] : u_{k-1} < d_{kt}\}$. Due to the assumption of $u_{t-1} \le d_t + u_t$, for all $t \in [1,n]$, we notice that $v(S^+) = u_{k-1}$. Moreover, notice that the maximum flow that can be sent by a production arc t is $\min\{d_{t\ell}, d_t + u_t\}$. If $t \le p$, then $v(S^+ \cup \{t\}) = d_{kt-1} + \min\{u_t + d_t, d_{t\ell}\}$ and if $t > p$, then $v(S^+ \cup \{t\}) = u_{k-1} + \min\{u_t + d_t, d_{t\ell}\}$. Using the definition of p, one can write $v(S^+ \cup \{t\})$ in a compact form as $\min\{u_{k-1} + \min\{u_t + d_t, d_{t\ell}\}, d_{kt-1} + \min\{u_t + d_t, d_{t\ell}\}\}$. After some algebraic manipulation, we observe that

$$\rho_t(S^+) = (\min\{d_{kt} + u_t - u_{k-1}, d_{k\ell} - u_{k-1}, u_t + d_t, d_{t\ell}\})^+$$

for each $t \in L^+$. □

Proposition E.2. Inequality (E.2) is weaker than inequality (4.9) for LSBI, where $S^+ = \{(k-1, k)\}$.

Proof. Proof follows similar to that of Proposition E. Since $a_t = 0$ for the inventory arc $(\ell, \ell+1)$, the effective demand at node ℓ becomes $d_\ell + u_\ell$. This leads to

$$\rho_t(S^+) = (\min\{d_{kt} + u_t - u_{k-1}, d_{k\ell} + u_\ell - u_{k-1}, u_t + d_t, d_{t\ell} + u_\ell\})^+ .$$

From the assumption of $u_{t-1} \le d_t + u_t$, for all $t \in [1,n]$, we know that $d_{kt} + u_t - u_{k-1} \le d_{k\ell} + u_\ell - u_{k-1}$ and $u_t + d_t \le d_{t\ell} + u_\ell$. Then, for each $t \in L^+$,

$$\rho_t(S^+) = (\min\{d_{kt} + u_t - u_{k-1}, d_t + u_t\})^+ .$$

□

In the second set of inequalities, they introduce inventory fixed charge variables to inequalities (E.1) and (E.2). Recall that $p = \min\{t \in [k,\ell] : u_{k-1} < d_{kt}\}$. For $1 \le k \le \ell \le n$ such that $u_{k-1} \le d_{k\ell}$, let $L \subseteq [k,\ell]$ and $T = \{t_1, t_2, \ldots, t_\tau\} \subseteq [k-1, p-1]$. For $j \in T$, let $s(j) = \min\{L \cup \{\ell+1\} : t > j\}$. Then the inequalities

$$i_{k-1} + \sum_{t \in L^+} y_t + \sum_{t \in T} \gamma_t (1 - z_t) \le u_{k-1} + \sum_{t \in L^+} \min\{d_{kt} + u_t - u_{k-1}, d_{k\ell} - u_{k-1}, d_{t\ell}\} x_t + i_\ell, \quad \text{(E.3)}$$

$$i_{k-1} + \sum_{t \in L^+} y_t + \sum_{t \in T} \gamma_t (1 - z_t) \le u_{k-1} + \sum_{t \in L^+} (d_{kt} - u_{k-1} + u_t) x_t, \quad \text{(E.4)}$$

where

$$\gamma_{t_j} = \begin{cases} u_{k-1} - d_{kt_\tau} & \text{if } j = \tau \text{ and } s(t_j) > p, \\ d_{(t_j+1)(t_{j+1})} & \text{if } j < \tau \text{ and } s(t_j) = s(t_{(j+1)}), \\ d_{(t_j+1)(s(t_j)-1)} & \text{if } (j < \tau \text{ and } s(t_j) < s(t_{(j+1)})) \text{ or } (j = \tau \text{ and } s(t_j) \le p) \end{cases}$$

are valid for LSBI.

Proposition E.3. Let s represent the inventory arc $(k-1,k)$ and let $t \in [k-1,p-1]$. For $S=\{s\}$ and $T=\{(t,t+1)\}$, the inequality

$$i_{k-1} + \sum_{j\in L^+} y_j + \big(\rho_s(L(t)) - \rho_s(L(t)|z_t=0)\big)(1-z_t) \le v(S) + \sum_{j\in L^+} \rho_j(S)x_j \qquad \text{(E.5)}$$

where $L(t) = L^+ \cap [t+1,\ell]$ is equivalent to inequality (E.3) and is valid for LSBI.

Proof. Let $(\mathbf{y}^*, \mathbf{i}^*, \mathbf{x}^*, \mathbf{z}^*)$ be a feasible solution of LSBI and let $\bar{L}_t = L^+ \setminus L(t)$. We show validity of inequality (E.5), under two cases: $z_t^*=1$ and $z_t^*=0$. When $z_t^*=1$, inequality (E.5) is equivalent to inequality (4.9) and is valid for LSBI.

Now, suppose $z_t^*=0$. Inequality (E.5) becomes

$$i^*_{k-1} - i^*_\ell + \sum_{j\in L^+} y^*_j \le v(S) - \rho_s(L(t)) + \rho_s(L(t)|z_t=0) + \sum_{j\in \hat{L}^+} \rho_j(S), \qquad \text{(E.6)}$$

where $\hat{L}^+ = \{t \in L^+ : x_t = 1\}$. Moreover, let $\hat{L}(t) = \hat{L}^+ \cap L(t)$. From the definition of the function v, when $z_t^*=0$, it is guaranteed that

$$i^*_{k-1} - i^*_\ell + \sum_{j\in \hat{L}^+} y^*_j \le v(s \cup \hat{L}^+ | z_t = 0).$$

Moreover, using the structure of the path we observe

$$\rho_s(L(t)|z_t=0) = \rho_s(\hat{L}(t)|z_t=0)$$

and

$$v(\{s\}\cup \hat{L}^+|z_t=0) = \rho_s(\hat{L}(t)|\ z_t=0) + v(\hat{L}(t)|z_t=0)$$

since $t \le p-1$. Then, we make the following observation about the right hand side of inequality (E.6):

$$\begin{aligned}
v(\{s\}) - \rho_s(L(t)) + \rho_s(L(t)|z_t=0) + \sum_{j\in \hat{L}^+} \rho_j(S)& \\
&\ge v(\{s\}\cup \hat{L}(t)) - \rho_s(L(t)) + \rho_s(L(t)|z_t=0) + \sum_{j\in \bar{L}(t)\cap \hat{L}^+} \rho_j(\{s\}) \\
&\ge v(\{s\}\cup \hat{L}(t)) - \rho_s(\hat{L}(t)) + \rho_s(L(t)|z_t=0) + \sum_{j\in \bar{L}(t)\cap \hat{L}^+} \rho_j(\{s\}) \\
&= v(\hat{L}(t)) + \rho_s(L(t)|z_t=0) + \sum_{j\in \bar{L}(t)\cap \hat{L}^+} \rho_j(\{s\}) \\
&= v(\{s\}\cup \hat{L}^+|z_t=0) + \sum_{j\in \bar{L}(t)\cap \hat{L}^+} \rho_j(\{s\}).
\end{aligned}$$

In the first and second inequalities, we use submodularity of the function v and in the last equality, we use the structure of the path. As a result,

$$
\begin{aligned}
i^*_{k-1} - i^*_\ell + \sum_{j\in L^+} y^*_j &\leq v(\{s\} \cup \hat{L}^+ | z_t = 0) \\
&\leq v(\{s\}) - \rho_s(L(t)) + \rho_s(L(t)|z_t = 0) + \sum_{j\in\hat{L}^+} \rho_j(S)
\end{aligned}
$$

and we conclude that inequality (E.5) is valid for LSBI when $z^*_t = 0$. □

As it is pointed out in Atamtürk (2004), the coefficients γ_t are sequence-dependent. Let the sequence of inventory arcs to be lifted be represented as $T = \{t_1, \ldots, t_\tau\}$. Then, one can represent the coefficients as:

$$\gamma_{t_j} = \rho_s(L(t_j)|t_i = 0,\ \forall i \in [j+1, \tau]) - \rho_s(L(t_j)|t_i = 0,\ \forall i \in [j, \tau]).$$

Proposition E.4. Let $t_i, t_j \in T$ for some $i \neq j$. If $t_i > t_j$ and $i < j$, then $\gamma_{t_i} = 0$.

Proof. Recall that the coefficient γ_{t_i} can be represented as $\rho_s(L(t_i)|t_l = 0,\ \forall l \in [i + 1, \tau]) - \rho_s(L(t_i)|t_l = 0,\ \forall l \in [i, \tau])$. Without loss of generality, let $j := \arg\min_{l\in[i+1,\tau]}\{t_l\}$ and suppose $t_j < t_i$. Since $t_j < p$, we know that $\rho_s(L(t_i)|t_l = 0,\ \forall l \in [i, \tau]) = d_{kt_j}$ and $\rho_s(L(t_i)|t_l = 0,\ \forall l \in [i, \tau]) = d_{kt_j}$. Consequently, the coefficient of z_{t_i} is $\gamma_{t_i} = 0$. □

Due to Proposition E.4, we assume that $t_i < t_j$ for all $t_i, t_j \in T$ and $i < j$. In other words, we assume that T is an increasing set.

Remark E.1. For LSBI, the coefficients

$$\rho_s(L(t_i)|t_l = 0,\ \forall l \in [i, \tau]) = \rho_s(L(t_i)|t_i = 0)$$

and

$$\rho_s(L(t_i)|t_l = 0,\ \forall l \in [i+1, \tau]) = \rho_s(L(t_i)|t_{i+1} = 0).$$

In other words, the change in maximum flow by adding the inventory arc $(k-1, k)$ to $L(t_i)$ only depends on the smallest (left-most) inventory arc $(j, j+1)$ that is closed.

Note that the results shown in Propositions E.3, E.4 and Remark E.1 are not general for all network structures and depend heavily on the path structure of LSBI.

Proposition E.5. Let s represent the inventory arc $(k-1, k)$ and let $t \in [k-1, p-1]$. For $S = \{s\}$ and $T = \{(t, t+1)\}$, the inequality

$$i_{k-1} + \sum_{j\in L^+} y_j + \sum_{t\in T} \big(\rho_s(L(t)|t_{j+1} = 0) - \rho_s(L(t)|t_j = 0)\big)(1 - z_t) \leq v(S) + \sum_{j\in L^+} \rho_j(S) x_j \tag{E.7}$$

is equivalent to inequality (E.3) and is valid for LSBI.

Proof. Let $(\mathbf{y}^*, \mathbf{i}^*, \mathbf{x}^*, \mathbf{z}^*)$ be a feasible solution of LSBI. If $z_t^* = 1$ for all $t \in T$, then inequality (E.7) is equivalent to (4.9) and is valid for LSBI. Suppose $z_t^* = 0$ for some $t \in T$. Let $\hat{T} = \{t \in T : z_t^* = 0\}$. From the definition of the function v, we observe

$$i_{k-1}^* - i_\ell^* + \sum_{j \in L^+} y_j^* \leq v(\{s\} \cup \hat{L}^+ | z_i = 0, \forall i \in \hat{T}).$$

Inequality (E.7) when $z_i = 0$ for $i \in \hat{T}$ becomes

$$i_{k-1} + \sum_{j \in L^+} y_j \leq v(S) - \sum_{j \in \hat{T}} \big(\rho_s(L(j)|t_{j+1} = 0) - \rho_s(L(j)|t_j = 0)\big) + \sum_{j \in \hat{L}^+} \rho_j(S). \quad \text{(E.8)}$$

Let R be the right hand side of inequality (E.8) and let $t_{\max} = \max\{j \in \hat{T}\}$. Then, using submodularity and the path structure of LSBI, we observe the following

$$\begin{aligned}
& v(\{s\}) - \rho_s(L(t_{\max})|z_{t_{\max}+1} = 0) + \rho_s(L(t_{\max})|z_{t_{\max}} = 0) + \sum_{j \in \hat{L}(t_{\max})} \rho_j(S) && \text{(E.9)}\\
&\quad \geq v(\{s\} \cup \hat{L}(t_{\max})) - \rho_s(L(t_{\max})|z_{t_{\max}+1} = 0) \\
&\qquad\qquad + \rho_s(L(t_{\max})|z_{t_{\max}} = 0) && \text{(E.10)}\\
&\quad \geq v(\{s\} \cup \hat{L}(t_{\max})) - \rho_s(\hat{L}(t_{\max})|z_{t_{\max}+1} = 0) \\
&\qquad\qquad + \rho_s(L(t_{\max})|z_{t_{\max}} = 0) && \text{(E.11)}\\
&\quad \geq v(\{s\} \cup \hat{L}(t_{\max})) - \rho_s(\hat{L}(t_{\max})) + \rho_s(L(t_{\max})|z_{t_{\max}} = 0) && \text{(E.12)}\\
&\quad = v(\hat{L}(t_{\max})) + \rho_s(L(t_{\max})|z_{t_{\max}} = 0) && \text{(E.13)}\\
&\quad = v(\{s\} \cup \hat{L}(t_{\max})|z_{t_{\max}} = 0). && \text{(E.14)}
\end{aligned}$$

The inequality (E.12) holds since for any $t_j \in T$ and $t_i \geq t_j$,

$$\rho_s(L(t_j)|z_{t_i} = 0) \leq \rho_s(L(t_j))$$

due to the path structure of LSBI. Similarly, we reach the equality of (E.13) and (E.14) since $v(\hat{L}(t_{\max})) = v(\hat{L}(t_{\max})|z_{t_{\max}} = 0)$. Using the inequality (E.10)–(E.14), we observe that

$$\begin{aligned}
R \geq\ & v(\{s\} \cup \hat{L}(t_{\max})|z_{t_{\max}} = 0) \\
& - \sum_{i \in \hat{T} \setminus \{t_{\max}\}} \big(\rho_s(L(t_i)|t_{i+1} = 0) - \rho_s(L(t_i)|t_i = 0)\big) + \sum_{i \in \hat{L}^+ \setminus \hat{L}(t_{\max})} \rho_i(S).
\end{aligned}$$

For the rest of the proof, we use an induction logic. First, we introduce some notation for simplification: let $\hat{T}(t_j) = \hat{T} \cap \{t_j, \dots, t_\tau\}$ and let $v(C|\hat{T}(t_j)) := v(C|z_i =$

$0, \forall i \in \hat{T}(t_j))$ and $\rho_l(C|\hat{T}(t_j)) := \rho_l(C|z_i = 0, \forall i \in \hat{T}(t_j))$.

Assuming that the following is true

$$R \geq v(\{s\} \cup \hat{L}(t_{j+1})|\hat{T}(t_{j+1})) - \sum_{i \in \hat{T} \setminus [t_{j+1}, t_\tau]} (\rho_s(L(t_i)|z_{t_{i+1}} = 0) - \rho_s(L(t_i)|z_{t_i} = 0)) + \sum_{i \in \hat{L}^+ \setminus \hat{L}(t_{j+1})} \rho_i(S)$$

for iteration $j+1$, we extend the result for t_j. First, we make the following observation

$$\rho_i(S) \geq \rho_i(S|z_{t_{j+1}} = 0) = \rho_i(S|z_{t_i} = 0, \forall i \in [j+1, \tau])$$

for any $i \in L^+ \setminus L(t_{j+1})$. Then,

$$\begin{aligned} & v(\{s\} \cup \hat{L}(t_{j+1})|\hat{T}(t_{j+1})) - \rho_s(L(t_j)|t_{j+1} = 0) + \rho_s(L(t_j)|z_{t_j} = 0) + \sum_{i \in \hat{L}(t_j) \setminus \hat{L}(t_{j+1})} \rho_i(S) \\ & \geq v(\{s\} \cup \hat{L}(t_j)|\hat{T}(t_{j+1})) - \rho_s(L(t_j)|t_{j+1} = 0) + \rho_s(L(t_j)|z_{t_j} = 0) \quad \text{(E.15)} \\ & \geq v(\hat{L}(t_j)|\hat{T}(t_{j+1})) + \rho_s(L(t_j)|z_{t_j} = 0) \quad \text{(E.16)} \\ & = v(\{s\} \cup \hat{L}(t_j)|\hat{T}(t_j)). \quad \text{(E.17)} \end{aligned}$$

Inequalities (E.15)–(E.16) are obtained from the observation above and submodularity of the function v and the equality of (E.16) to (E.17) is due to the path structure of LSBI. As a result of the induction, we have proved that

$$R \geq v(\{s\} \cup \hat{L}|\hat{T}).$$

Then, combining with the definition of the function v, we conclude that

$$i^*_{k-1} - i^*_\ell + \sum_{j \in L^+} y^*_j \leq v(\{s\} \cup \hat{L}^+|\hat{T}) \leq R$$

and inequality (E.7) is valid for LSBI.

□

Remark E.2. In Atamtürk and Küçükyavuz (2005), they select $T \subseteq [k-1, p-1]$ since closing any inventory arc $j \geq p$ does not change $v(S^+)$ nor $\rho_t(S^+)$ for any $t \in L^+$. As a result, the lifting coefficient of such an inventory arc is zero.

Example E.1. Consider the example in Figure E.1. Inequality (4.9) for the node set $[2, 5]$ and the arc sets $S^+ = \{(1, 2)\}$ and $L^+ = \{2, 4, 5\}$ is

$$i_1 + y_2 + y_4 + y_5 \leq 40 + 14x_2 + 14x_4 + 14x_5 + i_5,$$

which is equivalent to inequality (E.1) for the same subsets of nodes and arcs.

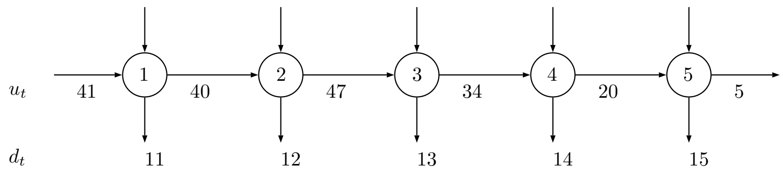


Figure E.1: A lot-sizing example with inventory bounds.

Note that $p = 5$ since $d_{24} \leq 40$ and $d_{25} > 40$. Selecting $T = \{(2,3)\}$, we get the lifting coefficient of 13. We obtain this value by the difference of $\rho_{(1,2)}(\{4,5\}) = 25$ and $\rho_{(1,2)}(\{4,5\}|z_2 = 0) = 12$. Consequently,

$$i_1 + y_2 + y_4 + y_5 + 13(1 - z_2) \leq 40 + 14x_2 + 14x_4 + 14x_5 + i_5,$$

is also valid for LSBI.