Accepted Manuscript : https://doi.org/10.1016/j.ejor.2023.03.003

Valid inequalities and extended formulations for lot-sizing and scheduling problem with sequence-dependent setups

Younsoo Lee[a], Kyungsik Lee[a,*]

[a] *Department of Industrial Engineering, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 08826, Republic of Korea*

Abstract

In this paper, we propose new valid inequalities and extended formulations for the lot-sizing and scheduling problem with sequence-dependent setups, which are derived by investigating the single-period substructure of the problem. Specifically, we derive two new families of valid inequalities and identify their facet-defining conditions. Additionally, we demonstrate that these inequalities can be separated in polynomial time. After introducing the existing extended formulations for the problem, we provide new extended formulations adapting decision variables representing the time-flow and compare the theoretical strengths of the various formulations and valid inequalities, including the proposed ones. Finally, we conduct computational experiments to demonstrate the effectiveness of the proposed inequalities and formulations. The test results indicate that the proposed inequalities and extended formulations facilitate tightening the linear programming relaxation bounds.

Keywords: Production, Lot-sizing and scheduling problem, Sequence-dependent setup, Valid inequality, Extended formulation

1. Introduction

1.1. Problem description

Traditional capacitated lot-sizing problems determine the sizes of the production lots of products for each period within a given planning horizon (Manne, 1958). The goal is to meet the dynamic demand at the minimum cost while satisfying constraints such as limited production capacity and setup requirements. There are plenty of variations of the problem taking into account additional characteristics (Quadt & Kuhn, 2008). One important extension is to consider the *sequence-dependent setup* which indicates that the setup cost and time depend on both the item that was produced previously and that which will be produced subsequently. In the presence of the sequence-dependent setup, the problem needs to determine not only the lot sizes but also the

*Corresponding author.
Email addresses: `fifays@snu.ac.kr` (Younsoo Lee), `optima@snu.ac.kr` (Kyungsik Lee)

sequence of the production because it affects both the total cost and available production capacity. In this paper, we consider the *lot-sizing and scheduling problem with sequence-dependent setups* (LSP-SQ). The formal description and mathematical formulation of the problem will be provided in Section 1.3.

LSP-SQ has been widely investigated over the recent decades because of its importance in both industrial and academic communities. Particularly, there is abundant research on the modeling of LSP-SQ (refer to Pochet & Wolsey, 2006; Copil et al., 2017, for the comprehensive review on different models). Among the various modeling frameworks, we consider the so-called *big bucket models* which are natural extensions of the conventional capacitated lot-sizing model (Manne, 1958). In big bucket models, the planning horizon is divided into multiple periods of length equal to the granularity of demand occurrence, for example, days. In each period, multiple items can be produced, and the production sequence should be determined. As mentioned by Gupta & Magnusson (2005), for big bucket models, the production sequence within each period is represented as a cycle which is illustrated in Section 1.3.

Regarding computational complexity, LSP-SQ is an NP-hard problem. In particular, it is well-known that even with a single item, the capacitated lot-sizing problem without the scheduling decisions is already NP-hard (Bitran & Yanasse, 1982). Moreover, even with the given lot-sizing decisions, the remaining scheduling problem is NP-hard because it can be reduced to the traveling salesman problem (TSP). Because of the difficulty inherited from both, LSP-SQ is strongly NP-hard.

One possible approach to tackle NP-hard problems is to exploit the substructures of the problems using polyhedral analysis. There are many cases where valid inequalities and extended formulations which are the results of the polyhedral analysis, play a major role in improving the practical solvability of NP-hard problems (Wolsey, 2020).

Regarding LSP-SQ, some research has been conducted on substructures. However, most of these studies address the single-item substructure, where sequence-dependent setups between different items cannot be incorporated. In this regard, we study the *single-period substructure* of LSP-SQ incorporating the sequence-dependency, which is formally described in Section 2.

The contributions of this study are as follows. We propose new families of valid inequalities for LSP-SQ which are derived from a polyhedral study on the single-period substructure. We analyze their strengths and facet-defining conditions. Additionally, we demonstrate that the separation problems of these inequalities can be solved in polynomial time. Subsequently, we introduce new extended formulations adapting decision variables representing the time-flow. Their strengths are analyzed and compared with those of the existing formulations, both theoretically and computationally. The results of the computational experiments on both single-period and multi-period instances demonstrate the effectiveness of the newly proposed inequalities and extended formulations.

1.2. Literature review

Research on LSP-SQ includes studies dealing with various practical industrial problems (for example Ríos-Solís et al., 2020; Lee & Lee, 2020) and devising efficient solution approaches for general problem instances (for example Guimarães et al., 2013; Carvalho & Nascimento, 2022). In our literature review, we focus on polyhedral studies, such as valid inequalities and extended formulations of LSP-SQ and related problems. For a comprehensive review including the industrial problems and various solution approaches, refer to Zhu & Wilhelm (2006) and Copil et al. (2017).

Most polyhedral studies on the substructure of LSP-SQ have primarily considered the *single-item* structure. For instance, Barany et al. (1984) studied a single-item uncapacitated lot-sizing problem (ULSP) and provided a complete linear description of the convex hull of the ULSP in its original space using valid inequalities, denoted as (l, S)–inequalities. Krarup & Bilde (1977) and Eppen & Martin (1987) proposed extended formulations for ULSP which can provide an optimal solution by solving their linear programming (LP) relaxations, based on the facility location and shortest path reformulations, respectively. Subsequently, Küçükyavuz & Pochet (2009) provided an explicit description of the convex hull of the ULSP with backlogging in the original space. Leung et al. (1989) and Van Vyve (2007) addressed a single-item capacitated problem with constant capacity. They analyzed the polyhedral structure, proposed facet-defining inequalities, and devised polynomial-time solution algorithms. The results of the aforementioned studies on single-item substructures have been successfully adapted to generalized problems with multiple items or time-varying capacity.

Contrary to the single-item cases, there have been only a limited number of studies on the *single-period* substructure, and even those studies did not consider the sequence-dependent setups. Miller et al. (2003b) studied the single-period relaxation of a capacitated lot-sizing problem with sequence-independent setups. By incorporating multiple items competing for a limited production capacity, the authors derived valid inequalities and their facet-defining conditions. They considered a special case in Miller et al. (2003a), where the demand and setup times were constant for all items. In this special case, the authors proposed a polynomial-time algorithm and derived an extended formulation. More recently, Doostmohammadi & Akartunalı (2018) studied the two-period substructure and derived facet-defining inequalities, but assumed zero setup times. Unfortunately, these results cannot be directly applied to LSP-SQ because the sequence-dependent setups are not considered. To the best of our knowledge, there is a lack of studies on the single-period substructure of LSP-SQ incorporating sequence-dependent setups.

Meanwhile, the solution set of the single-period LSP-SQ is closely related to that of routing-type problems such as the *capacitated vehicle routing problem* (CVRP) (Toth & Vigo, 2002) in that, both problems select a subset of items to produce (or customers to visit) from the given sets and determine the sequence of the production (or sequence of the visit). Particularly, because the sequencing decision in a single period of LSP-SQ is to determine the order of production of items which are decided to produce, it resembles the routing decision in CVRP which determines the order of visits of customers who are decided to visit. Therefore, the valid inequalities and extended formulations of CVRP are also relevant to those of LSP-SQ. Gouveia (1995) studied the projection of single-commodity flow formulations of CVRP. As a result of the projection, they derived valid inequalities called multi-star inequalities. These results were generalized by Letchford et al. (2002) and Letchford & Salazar-González (2006). Letchford et al. (2002) introduced generalized multi-star inequalities and reported their computational effects. Letchford & Salazar-González (2006) surveyed various formulations of CVRP and inequalities derived by the projection, and analyzed the relations between them. Later, Letchford & Salazar-González (2015) provided stronger formulations. These results are relevant to our problem, as discussed in Section 3. However, in contrast to CVRP, the single-period LSP-SQ should make additional decisions regarding the production amount. In this respect, the results of CVRP are not sufficient for LSP-SQ.

Guimarães et al. (2014) reviewed various formulations of LSP-SQ and proposed classification criteria. There are various formulations, such as GSEC-based formulation with exponentially many constraints, pattern-based formulations (Guimarães et al., 2013) with exponentially many variables, and single/multi-commodity flow formulations. They conducted extensive computational experiments to compare their computational performance and reported that, on average, the single-commodity flow formulation showed the best performance. However, the theoretical strengths of the formulations and their relationships were not investigated.

Recently, Lee & Lee (2021) proposed a time-flow formulation for the *general lot-sizing and scheduling problem model* (Fleischmann & Meyr, 1997) which is a hybrid of the big bucket and small bucket models. As the authors mentioned, their time-flow formulation can be extended to the big bucket models. Accordingly, we propose new time-flow formulations for the big bucket models and compare both the theoretical strength and computational performance.

1.3. Mathematical Formulation of LSP-SQ

We provide a generic mathematical formulation of big bucket models LSP-SQ with sets of items $\mathcal{I} = \{1, ..., I\}$ and periods $\mathcal{T} = \{1, \dots, T\}$. Throughout the exposition, $i, j \in \mathcal{I}$, $t \in \mathcal{T}$ are used as indices. We additionally define a fictitious item 0 to represent the start and end of the production sequence, and let $\mathcal{I}_0 = \mathcal{I} \cup \{0\}$.

Let hc_{it}, bc_{it}, pc_{it}, and d_{it} denote the unit inventory holding cost, backlogging cost, production cost, and demand for item i and period t, respectively. The setup cost and time between items i and j in period t are denoted by sc_{ijt} and st_{ijt}, respectively. For notational convenience, we also define st_{i0t} and st_{0it} and let their values be zero. The production capacity of period t, given in time units, is denoted by K_t, whereas the unit production time of item i is denoted by a_i.

Let s_{it} and b_{it} be the decision variables representing the inventory and backlog amounts of item i at the beginning of period t, respectively. The initial inventory and backlog amounts of item i are denoted by s_{i0} and b_{i0}, respectively, and are assumed to be zero. Variable x_{it} represents the production amount of item i in period t. The binary variable y_{it} is equal to one if item i is produced

Table 1: Nomenclature

Sets	
$\mathcal{I}$	Set of items which are indexed by i and j; $\mathcal{I} = \{1, \ldots, I\}$
$\mathcal{I}_0$	Set of items including the fictitious item 0; $\mathcal{I}_0 = \mathcal{I} \cup \{0\}$
$\mathcal{T}$	Set of time periods which are indexed by t; $\mathcal{T} = \{1, \ldots, T\}$
Parameters	
hc_{it}	Inventory holding cost of item i in period t
bc_{it}	Backlogging cost of item i in period t
pc_{it}	Production cost of item i in period t
d_{it}	Demand of item i in period t
sc_{ijt}	Cost incurred when setup occurs from item i to j in period t
st_{ijt}	Time needed for setup from item i to j
K_t	Production capacity of period t given in time unit
a_i	Production time per unit of item i
Variables	
s_{it}	Inventory amount of item i at the end of period t, $s_{i0} = 0$
b_{it}	Backlog amount of item i at the end of period t, $b_{i0} = 0$
x_{it}	Production amount of item i in period t
y_{it}	$= 1$ if item i is produced in period t
z_{ijt}	$= 1$ if setup from item i to j occurs in period t
z_{0it} (z_{i0t})	$= 1$ if item i is the first (last) produced item in period t

in period t. The binary variable z_{ijt} is equal to one if the setup from item i to item j occurs in period t. Moreover, let z_{0it} and z_{i0t} be the binary variables which represent whether item i is the first and last item produced in period t, respectively. The notations used are summarized in Table 1. We use boldface to denote matrices; for example, $\mathbf{x} = (x_{it})_{i\in\mathcal{I},t\in\mathcal{T}}$ is a matrix of variables representing the production amount of each item at every period. The generic mathematical formulation of the big bucket models of LSP-SQ (Guimarães et al., 2014) can be written as follows:

$$\text{minimize} \quad \sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}}\Big(hc_{it}s_{it} + bc_{it}b_{it} + pc_{it}x_{it}\Big) + \sum_{(i,j)\in\mathcal{A}}\sum_{t\in\mathcal{T}} sc_{ijt}z_{ijt} \tag{1a}$$

$$\text{subject to} \quad s_{it-1} - b_{it-1} + x_{it} = d_{it} + s_{it} - b_{it} \qquad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{1b}$$

$$\sum_{i\in\mathcal{I}} a_i x_{it} + \sum_{(i,j)\in\mathcal{A}} st_{ijt}z_{ijt} \le K_t \qquad \forall t \in \mathcal{T} \tag{1c}$$

$$x_{it} \le K_t y_{it} \qquad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{1d}$$

$$z_{i0t} = z_{0it+1} \qquad \forall i \in \mathcal{I}, t \in \mathcal{T} \setminus \{T\} \tag{1e}$$

$$\sum_{j\in\mathcal{I}_0\setminus\{i\}} z_{jit} = \sum_{j\in\mathcal{I}_0\setminus\{i\}} z_{ijt} = y_{it} \qquad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{1f}$$

$$\text{Do not include cycles without item 0} \qquad \forall t \in \mathcal{T} \tag{1g}$$

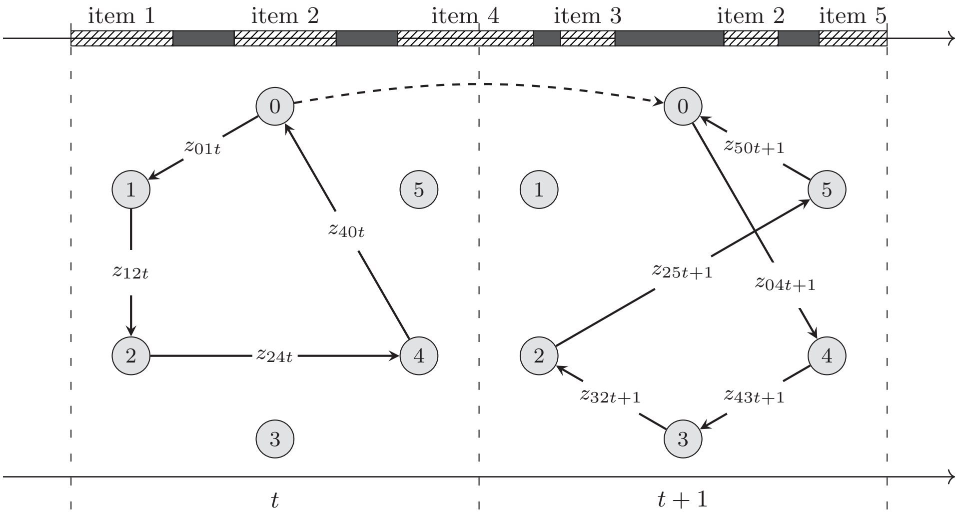

Figure 1: Production sequence and its representation as a big bucket model with a cycle for each period

$$x_{it},\ s_{it} \geq 0,\ y_{it} \in \{0,1\} \qquad \forall i \in \mathcal{I}, \forall t \in \mathcal{T} \qquad (1h)$$

$$z_{ijt} \in \{0,1\} \qquad \forall (i,j) \in \mathcal{A}, \forall t \in \mathcal{T} \qquad (1i)$$

Objective function (1a) is the sum of the inventory holding, backlogging, production, and setup costs, the total of which must be minimized. Constraints (1b) are balance equations between the demand, inventory, backlog, and production amounts. Constraints (1c) ensure that the sum of the production and setup times does not exceed the available capacity in each period. Constraints (1d) indicate that an item can only be produced if the corresponding setup occurs. Constraints (1e) indicate that the setup for the last item in the previous period is carried over to the next period. Constraints (1f) logically link the binary variables to ensure a balanced flow of setups. Constraints (1h)–(1i) ensure the domain of variables.

In a big bucket model, the production sequence within each period is represented as a cycle including item 0. For instance, let us consider the production plan for two periods t and $t+1$ as shown in Figure 1. In period t, items 1, 2, and 4 are produced, whereas items 3 and 5 are not produced. Items within this period are produced in a sequence of $1-2-4$ which can be represented as a cycle shown in the left-hand side of Figure 1. Similarly, the production sequence of $4-3-2-5$ within period $t+1$ also can be represented as a cycle. As shown in the figure, each item corresponds to a node, whereas the setup between the two items corresponds to an arc. Throughout this paper, the terms item and node are used interchangeably, and setup and arc are also used as such.

Additionally, we define a directed graph $\mathcal{G} = (\mathcal{I}_0, \mathcal{A}_0)$, where the arc set is defined as $\mathcal{A}_0 := \{(i,j) : i \in \mathcal{I}_0, j \in \mathcal{I}_0, i \neq j\}$. We also define $\mathcal{A} := \{(i,j) : i \in \mathcal{I}, j \in \mathcal{I}, i \neq j\}$. For sets of nodes $S, T \subseteq \mathcal{I}_0$, we denote $E(S : T)$ as the set of arcs (i,j), such that $i \in S$ and $j \in T$. Using the definition of $E(S:T)$, we also can define $E(S) := E(S:S)$, the set of arcs with both endpoints in S, $\delta^+(S) := E(S:\mathcal{I}_0 \setminus S)$, set of outgoing arcs from S, $\delta^-(S) := E(\mathcal{I}_0 \setminus S : S)$, set of incoming arcs to S, and $\delta(S) := \delta^+(S) \cup \delta^-(S)$, set of arcs with one endpoint in S and another in $\mathcal{I}_0 \setminus S$.

As mentioned previously, a production sequence within each period is represented as a cycle on $\mathcal{G}$, including item 0. To ensure the validity of the cycle, the constraints (1g) which prevent cycles without item 0 are necessary. One possible option for the constraints (1g) is the *generalized subtour elimination constraints* (GSECs, Toth & Vigo, 2002) which are written as follows:

$$\sum_{(j,i)\in\delta^-(S)} z_{jit} \geq y_{kt} \qquad \forall S \subseteq \mathcal{I},\ k \in S,\ t \in \mathcal{T} \tag{2}$$

GSECs ensure that an item k contained in S can be produced only if at least one incoming arc from the outside of S is selected. Note that there are several other alternatives, such as the Miller–Tucker–Zemlin formulation (Miller et al., 1960) or the single-commodity flow formulation (Gavish & Graves, 1978) which can also eliminate invalid cycles using additionally defined decision variables. These formulations, with additional variables other than those used in the LSP-SQ model (1), are denoted as *extended formulations*.

The remainder of this paper is organized as follows. In Section 2, a single-period substructure of LSP-SQ is formally presented, and the basic polyhedral properties are provided. In Section 3, we provide new families of valid inequalities and discuss their properties. In Section 4, we propose new extended formulations and compare them with the existing formulations. In Section 5, we present the results of the computational experiments. In Section 6, we provide concluding remarks and discuss possible future extensions of this study.

2. Single-period Substructure

The single-period substructure of LSP-SQ and its properties are provided in this section. As its name indicates, this substructure is derived by relaxing the constraints of the LSP-SQ formulation (1) which impose relations between different periods. By relaxing these constraints, the problem can be decomposed into several single-period substructures defined for $t \in \mathcal{T}$. A similar form of relaxation was introduced by Miller et al. (2003a) for LSP with sequence-independent setups. Following Miller et al. (2003a), we introduce the variables s_i^- and s_i^+ which represent the shortage and surplus for the demand of item i, respectively, to ensure the feasibility of the problem. Because we consider only a single period, we omit period index t.

$$\text{minimize} \quad \sum_{i\in\mathcal{I}} \left(hc_i s_i^+ + bc_i s_{it}^- + pc_i x_i \right) + \sum_{(i,j)\in\mathcal{A}} sc_{ij} z_{ij} \tag{3a}$$

$$\text{subject to} \quad s_i^- + x_i = d_i + s_i^+ \qquad \forall i \in \mathcal{I} \tag{3b}$$

$$\sum_{i\in\mathcal{I}} a_i x_i + \sum_{(i,j)\in\mathcal{A}} st_{ij} z_{ij} \le K \tag{3c}$$

$$x_i \le u_i y_i \qquad \forall i \in \mathcal{I} \tag{3d}$$

$$\sum_{i\in\mathcal{I}} z_{0i} = 1 \tag{3e}$$

$$\sum_{j\in\mathcal{I}_0\setminus\{i\}} z_{ji} = \sum_{j\in\mathcal{I}_0\setminus\{i\}} z_{ij} = y_i \qquad \forall i \in \mathcal{I} \tag{3f}$$

$$\sum_{(i,j)\in\delta^+(S)} z_{ij} \ge y_k \qquad \forall k \in S, S \subseteq \mathcal{I} \tag{3g}$$

$$x_i, s_i^-, s_i^+ \ge 0,\ y_i \in \{0,1\} \qquad \forall i \in \mathcal{I} \tag{3h}$$

$$z_{ij} \in \{0,1\} \qquad \forall (i,j) \in \mathcal{A}_0 \tag{3i}$$

Let $\mathcal{X} = \{(\mathbf{x}, \mathbf{s}^+, \mathbf{s}^-, \mathbf{y}, \mathbf{z}) \in \mathbb{R}_+^{3I} \times \mathbb{B}^{I^2+2I} :$ satisfies constraints (3b)$-$(3i)$\}$ be the solution set of problem (3) where $\mathbb{R}_+$ and $\mathbb{B}$ represent the nonnegative real number and $\{0,1\}$ region, respectively. Note that the demand constraints (3b) can always be satisfied owing to the variables s_i^- and s_i^+, and the solution set is not empty. In addition, to enrich the analysis, we introduce the parameter u_i, the upper bound on the production amount of item $i \in \mathcal{I}$. The individual upper bounds can be dropped by letting $u_i = K$, $\forall i \in \mathcal{I}$. We denote $\mathcal{X}_0$ as the solution set of problem (3) with $u_i = K$, $\forall i \in \mathcal{I}$. We use `GSEC`s (3g) to prevent invalid cycles.

Before presenting the basic polyhedral properties, we make some assumptions. Firstly, it can be shown that the problem (3) and the corresponding solution set $\mathcal{X}$ defined with general a_i value can be transformed into equivalent problem and solution set with $a_i = 1$, respectively, without loss of generality. This can be demonstrated by considering variables $x_i' = a_i x_i$, $s_i^{-\prime} = a_i s_i^-$, $s_i^{+\prime} = a_i s_i^+$, and the modified coefficients $d_i' = a_i d_i$ and $u_i' = a_i u_i$. By replacing the original variables and coefficients $(x_i, s_i^-, \ldots)$ with the newly defined variables and coefficient $(x_i', s_i^{-\prime}, \ldots)$, it can be shown that we can let $a_i = 1$ for all $i \in \mathcal{I}$ without loss of generality. Therefore, we set $a_i = 1$ for all $i \in \mathcal{I}$. We also assume that $0 < st_{ij} \le K$, $\forall (i,j) \in \mathcal{A}$ and $u_i \le K$, $\forall i \in \mathcal{I}$.

2.1. Basic polyhedral properties

Before presenting our main results, we firstly analyze the basic properties of the convex hull of $\mathcal{X}$, that is, $conv(\mathcal{X})$. The technical proofs of the following propositions are given in Appendix A.

Proposition 2.1. *Dimension of $conv(\mathcal{X})$ is $I^2 + 2I - 1$.*

Proposition 2.2. *Constraint* (3c) *is a facet-defining inequality of $conv(\mathcal{X})$ if $u_i = K$ for all $i \in \mathcal{I}$, that is, it defines a facet of $conv(\mathcal{X}_0)$.*

Proposition 2.3. *For a given $i \in \mathcal{I}$, if $u_i \le K - \max\{st_{ij}, st_{ji}\}$ for all $j \in \mathcal{I} \setminus \{i\}$, constraint* (3d) *defines a facet of $conv(\mathcal{X})$.*

When the condition given in Proposition 2.3 does not hold, the constraint (3d) does not define the facets of $conv(\mathcal{X})$. Intuitively, if u_i is greater than the given value, for instance $u_i = K$, it may be not possible for the production amount of item i being equal to u_i. In those cases, it is impossible for the constraint (3d) to define a facet of $\mathcal{X}_0$. Instead, the constraint (3d) can be tightened to be the facet-defining inequality of $conv(\mathcal{X}_0)$, as shown in Proposition 2.4.

Proposition 2.4. *For a given $i \in \mathcal{I}$, the following tightened upper bound constraint* (4) *is a facet-defining inequality of $conv(\mathcal{X})$ when $u_i = K$, that is, it defines a facet of $conv(\mathcal{X}_0)$.*

$$x_i \leq Ky_i - \sum_{j \in \mathcal{I} \setminus \{i\}} (st_{ij}z_{ij} + st_{ji}z_{ji}) \qquad \forall i \in \mathcal{I} \tag{4}$$

3. New Valid Inequalities

3.1. S-STAR inequality

In this section, we propose new families of valid inequalities for $\mathcal{X}$ and identify their facet-defining conditions. For a subset of items $S \subseteq \mathcal{I}$, the first inequality is defined as follows.

$$\sum_{i \in S} x_i + \sum_{(i,j) \in \delta(S)} st_{ij}z_{ij} + \sum_{(i,j) \in E(S)} st_{ij}z_{ij} \leq K \sum_{(i,j) \in \delta^-(S)} z_{ij} \qquad \forall S \subseteq \mathcal{I} \tag{5}$$

The inequality (5) is called the `S-STAR` inequality because the left-hand side terms form the shape of a star centered on S, as illustrated in Figure 2. The dark nodes represent the items included in S. The arcs included in $E(S)$ are represented with the dotted arcs, whereas those in $\delta(S)$ are represented with the solid arcs. Note that the arcs are presented in bidirectional ways to present both setups between any two items. The inequality indicates that the sum of production amounts of items in S and time for setups related to items in S cannot be greater than $K \sum_{(i,j) \in \delta^-(S)} z_{ij}$. This inequality can also eliminate all cycles without item 0 which indicates that the `S-STAR` inequality can replace `GSEC` (3g). This is demonstrated in the following proposition.

Proposition 3.1. *For a given $S \subseteq \mathcal{I}$, the inequality* (5) *is valid for $\mathcal{X}$. In addition, these inequalities are sufficient to eliminate all cycles which do not include item* 0.

Proof. For a given feasible solution of $\mathcal{X}$, if $\sum_{(i,j) \in \delta^-(S)} z_{ij} \geq 1$, the inequality trivially holds as the right-hand side becomes greater than or equal to the capacity K. On the other hand, if $\sum_{(i,j) \in \delta^-(S)} z_{ij} = 0$, there are no incoming arcs to the nodes in S which indicates that the items in S cannot be produced, and the corresponding setups that start or end with the items in S cannot be conducted. Therefore, the left-hand side value also becomes zero and the inequality holds.

To demonstrate that the `S-STAR` inequality can prevent any invalid cycles, suppose that we are given a cycle that does not include item 0, and let N be the set of nodes in the cycle ($|N| \geq 2$). Then, from the definition, there are no incoming and outgoing arcs for N. Therefore, $\sum_{(i,j) \in \delta^-(N)} z_{ij} = 0$, and the right-hand side of inequality (5) defined by N becomes zero. On the other hand, as $\sum_{(i,j)\in E(N)} z_{ij} = |N|$ and $st_{ij} > 0$, $\forall (i,j) \in \mathcal{A}$, the left-hand side is greater than 0, violating inequality (5). This shows that invalid cycles can be eliminated by adding the `S-STAR` inequalities. □

We note that the `S-STAR` inequality is closely related to the *generalized large multistar* (`GLM`) inequality proposed for CVRP by Gouveia (1995) and Letchford et al. (2002), which can be presented as follows:

$$\sum_{i\in S} d_i y_i + \sum_{(i,j)\in\delta^+(S)} d_j z_{ij} + \sum_{(i,j)\in\delta^-(S)} d_i z_{ij} \leq K \sum_{(i,j)\in\delta^-(S)} z_{ij} \qquad \forall S \subseteq \mathcal{I} \tag{GLM}$$

To demonstrate the relation between two inequalities more explicitly, consider the following generic inequality which subsumes both `S-STAR` and `GLM` inequalities, although it has nonlinear terms $x_i z_{ij}$.

$$\sum_{i\in S} x_i + \sum_{(i,j)\in\delta^+(S)} (st_{ij} + x_j) z_{ij} + \sum_{(i,j)\in\delta^-(S)} (x_i + st_{ij}) z_{ij} + \sum_{(i,j)\in E(S)} st_{ij} z_{ij} \leq K \sum_{(i,j)\in\delta^-(S)} z_{ij} \qquad \forall S \subseteq \mathcal{I} \tag{6}$$

In CVRP, the value of x_i which indicates the delivery amount for customer i, cannot have any value between 0 and u_i, but must be either 0 or d_i, that is, $x_i = d_i y_i$. In addition, no setup time is considered in CVRP. In this regard, by setting $st_{ij} = 0$ and $x_i = d_i y_i$ for inequality (6), we can obtain the `GLM` inequality immediately because $x_i z_{ij} = d_i y_i z_{ij} = d_i z_{ij}$ (note that $y_i z_{ij}$ can be

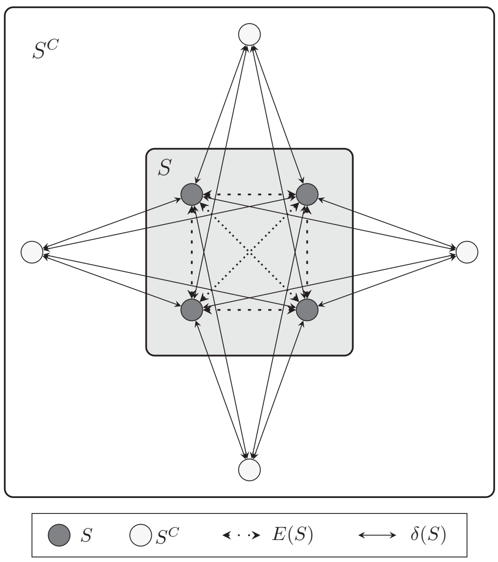

Figure 2: Illustration of the `S-STAR` inequality

linearized as z_{ij}). On the other hand, by dropping variables x_j and x_i in the second and third terms of inequality (6), respectively, we can obtain the S-STAR inequality.

In contrast to the GLM inequalities, the facet-defining conditions of the S-STAR inequalities can be identified. Specifically, if the individual upper bound for each item is not imposed, the S-STAR inequality becomes a facet-defining inequality of $conv(\mathcal{X})$.

Proposition 3.2. *Given $S \subseteq \mathcal{I}$, the S-STAR inequality defines a facet of $conv(\mathcal{X})$ when $u_i = K$, for all $i \in \mathcal{I}$. In other words, it defines a facet of $conv(\mathcal{X}_0)$.*

Proof. See Appendix A. □

3.1.1. Separation of S-STAR inequality

Because the S-STAR inequality is defined for each node subset $S \subseteq \mathcal{I}$, there are exponentially many inequalities. Therefore, it is not practical to add them all to the formulation initially. Rather, they are used as cutting planes, that is, they are dynamically added by solving a separation problem that identifies violated inequalities given a fractional solution. The separation problem of the S-STAR inequality, given a fractional solution $(\bar{\mathbf{x}}, \bar{\mathbf{s}}^+, \bar{\mathbf{s}}^-, \bar{\mathbf{y}}, \bar{\mathbf{z}})$, is to find a subset $N \subseteq \mathcal{I}$ such that

$$\sum_{i \in N} \bar{x}_i + \sum_{(j,i) \in E(N)} st_{ji} \bar{z}_{ji} + \sum_{(i,j) \in \delta(N)} st_{ij} \bar{z}_{ij} > K \sum_{(i,j) \in \delta^+(N)} \bar{z}_{ij},$$

$$\text{or equivalently,} \quad \sum_{i \in N} \Big(\bar{x}_i + \sum_{j \in \mathcal{I}_0 \setminus \{i\}} st_{ji} \bar{z}_{ji} \Big) + \sum_{i \in N} \sum_{j \notin N} \big(st_{ij} - K \big) \bar{z}_{ij} > 0.$$

To formulate this problem as an integer program, we define the following variables. For $i \in \mathcal{I}$, let us define the binary variable p_i which is equal to one if the item i is chosen to be a member of N. We let $p_0 = 0$. For $(i,j) \in \mathcal{A}_0$, let us define the binary variable q_{ij} which is equal to one if the arc (i,j) belongs to $\delta^+(N)$, that is, $i \in N$ and $j \notin N$. We also define additional notation to simplify the formulation. For $i \in \mathcal{I}$, let $\alpha_i = \bar{x}_i + \sum_{j \in \mathcal{I}_0 \setminus \{i\}} st_{ij} \bar{z}_{ij}$. For $(i,j) \in \mathcal{A}_0$, let $\gamma_{ij} = \big(K - st_{ij}\big) \bar{z}_{ij}$. Note that the parameters α_i and γ_{ij} are non-negative. Then, the separation problem of the S-STAR inequalities can be formulated as follows.

$$\begin{aligned}
\texttt{(SEP-S-STAR)} \quad \text{maximize} \quad & \sum_{i \in \mathcal{I}} \alpha_i p_i - \sum_{(i,j) \in \mathcal{A}_0} \gamma_{ij} q_{ij} && && \text{(7a)} \\
\text{subject to} \quad & q_{ij} \le p_i && \forall (i,j) \in \mathcal{A}_0 && \text{(7b)} \\
& q_{ij} \le 1 - p_j && \forall (i,j) \in \mathcal{A}_0 && \text{(7c)} \\
& p_i - p_j \le q_{ij} && \forall (i,j) \in \mathcal{A}_0 && \text{(7d)} \\
& p_i \in \{0,1\} && \forall i \in \mathcal{I} && \text{(7e)} \\
& q_{ij} \in \{0,1\} && \forall (i,j) \in \mathcal{A}_0 && \text{(7f)}
\end{aligned}$$

After solving the above separation problem, one can obtain a set of items N whose p_i value is one. If the optimal objective value is greater than zero, the S-STAR inequality defined by N is violated and should be added to the problem. On the other hand, the non-positive objective value indicates that there are no violated inequalities. Because this problem should be solved repeatedly during the cutting plane algorithm, it is important to identify its computational complexity and check whether it can be solved efficiently. We can show that the above separation problem can be solved in a polynomial time which is demonstrated in the following proposition.

Proposition 3.3. *(SEP-S-STAR) is polynomially solvable.*

Proof. The objective coefficients of the variables q_{ij} are non-positive. Therefore, there exists an optimal solution with q_{ij} value as small as possible. In this regard, the value of q_{ij} is determined by constraint (7d), that is, $q_{ij} = p_i - p_j$. Because $p_i - p_j \leq p_i$ and $p_i - p_j \leq 1 - p_j$ always hold, constraints (7b) and (7c) are automatically satisfied and can be dropped. Then, a matrix corresponding to the remaining constraints (7d) is of the form $[I_n|Q]$, where I_n is an identity matrix with $n = I + 1$ rows and columns, and each row of matrix Q contains only two nonzero coefficients of 1 and -1 which indicates that it is totally unimodular (Nemhauser & Wolsey, 1988). Therefore, the problem can be solved by solving its LP relaxation which can be performed in polynomial time. □

3.2. U-STAR inequality

The S-STAR inequality presented above defines a facet if there is no individual upper bound for the production amount of each item. On the other hand, the following valid inequality, denoted as U-STAR inequality, can be beneficial when the non-trivial upper bound for each item is presented.

Proposition 3.4. *For a given $S \subseteq \mathcal{I}$, the inequality*

$$\sum_{i \in S} x_i - \sum_{(i,j) \in E(S)} \lambda_{ij} z_{ij} \leq \sum_{(i,j) \in \delta^-(S)} u_j z_{ij} \tag{8}$$

is valid for $\mathcal{X}$, where $\lambda_{ij} := \min\{K - st_{ij} - u_i, u_j\}$ for all $(i,j) \in \mathcal{A}$.

Proof. See Appendix A. □

Proposition 3.5. *For a given $S \subseteq \mathcal{I}$, the U-STAR inequality (8) defines a facet of conv($\mathcal{X}$) if the following conditions hold.*

1. *For all $i \in S$ and $j \in \mathcal{I} \setminus S$, $u_i + \max\{st_{ij}, st_{ji}\} \leq K$.*
2. *For all $(i,j) \in E(S)$, $K - st_{ij} < u_i + u_j$.*

Proof. See Appendix A. □

Note that when $u_i = K$, the first condition is violated. In this case, the U-STAR inequalities are reduced to the weakened version of the S-STAR inequalities and therefore, are dominated by them.

3.2.1. Separation of U-STAR inequality

Similar to the S-STAR inequality, there are exponentially many number of U-STAR inequalities. The separation problem of the U-STAR inequality for a given fractional solution $(\bar{\mathbf{x}}, \bar{\mathbf{s}}^+, \bar{\mathbf{s}}^-, \bar{\mathbf{y}}, \bar{\mathbf{z}})$, is to find a subset $N \subseteq \mathcal{I}$ such that

$$\sum_{i \in N} \bar{x}_i - \sum_{(i,j) \in E(N)} \min\{K - st_{ij} - u_i, u_j\} \bar{z}_{ij} > \sum_{(i,j) \in \delta^-(N)} u_j \bar{z}_{ij}.$$

Because one can rewrite $\min\{K - st_{ij} - u_i, u_j\} = u_j - [u_j + u_i + st_{ij} - K]^+$, where $[X]^+ = \max\{0, X\}$, the problem is reduced to finding N such that

$$\sum_{i \in N} (u_i \bar{y}_i - \bar{x}_i) - \sum_{(i,j) \in E(N)} [u_j + u_i + st_{ij} - K]^+ \bar{z}_{ij} < 0.$$

To formulate the above separation problem as an integer program, we define additional variables. For $i \in \mathcal{I}$, let us define the binary variable p_i which is equal to one if $i \in N$. For $(i,j) \in \mathcal{A}$, let us define the binary variable q_{ij} which is equal to one if $(i,j) \in E(N)$, that is, $i \in N$ and $j \in N$. In addition, we let $\gamma_i = u_i \bar{y}_i - \bar{x}_i$ for $i \in \mathcal{I}$. For $(i,j) \in \mathcal{A}$, we define $\delta_{ij} = [u_j + u_i + st_{ij} - K]^+ \bar{z}_{ij}$. Note that both γ_i and δ_{ij} are non-negative. Using the notation, the separation problem (SEP-U-STAR) can be formulated. Furthermore, we can show that this problem is also polynomially solvable.

$$\begin{aligned}
\text{(SEP-U-STAR)} \quad \text{minimize} \quad & \sum_{i \in \mathcal{I}} \gamma_i p_i - \sum_{(i,j) \in \mathcal{A}} \delta_{ij} q_{ij} && && \text{(9a)} \\
\text{subject to} \quad & q_{ij} \leq p_i && \forall (i,j) \in \mathcal{A} && \text{(9b)} \\
& q_{ij} \leq p_j && \forall (i,j) \in \mathcal{A} && \text{(9c)} \\
& p_i + p_j - 1 \leq q_{ij} && \forall (i,j) \in \mathcal{A} && \text{(9d)} \\
& p_i \in \{0,1\} && \forall i \in \mathcal{I} && \text{(9e)} \\
& q_{ij} \in \{0,1\} && \forall (i,j) \in \mathcal{A} && \text{(9f)}
\end{aligned}$$

Proposition 3.6. *(SEP-U-STAR) is polynomially solvable.*

Proof. The objective coefficients of the variables q_{ij} are non-positive. Because (SEP-U-STAR) is a minimization problem, in contrast to (SEP-S-STAR), there exists an optimal solution with a q_{ij} value as large as possible. Therefore, the lower bounding constraints (9d) are redundant and can be dropped. Then, each row of the matrix corresponding to the remaining constraints (9b) and (9c) contains only two nonzero coefficients of 1 and -1 which indicates that it is totally unimodular

(Nemhauser & Wolsey, 1988). Therefore, the problem can be solved by solving its LP relaxation within a polynomial time. □

It is well-known that GSEC can be separated in polynomial time by solving maximum-flow problems (Wolsey, 2020). The separation problems (SEP-S-STAR) and (SEP-U-STAR) can also be converted to maximum-flow problems, similar to that of GSEC. Therefore, in our implementation of the separation algorithms, we used the maximum flow algorithms rather than solving LP for all three inequalities for computational efficiency.

We denote the formulations obtained by replacing the constraints (3g) in formulation (3) with the S-STAR inequality (5), and U-STAR inequality (8) as (S-STAR) and (U-STAR), respectively. In Section 4, we introduce extended formulations which can also be used to model the production sequence and ensure the validity of the cycle.

4. Extended Formulations

4.1. Single-commodity flow formulations

The single-commodity flow formulation, first proposed by Gavish & Graves (1978) for TSP, has been frequently used to model many routing problems. Guimarães et al. (2014) used this formulation to model LSP-SQ. To present this formulation, the variable f_{ij} for $(i, j) \in \mathcal{A}_0$ representing the commodity flow along the arc (i, j) should be defined.

$$\sum_{i \in \mathcal{I}} f_{0i} = \sum_{i \in \mathcal{I}} y_i \tag{10a}$$

$$\sum_{j \in \mathcal{I}_0 \setminus \{i\}} f_{ji} - \sum_{j \in \mathcal{I}_0 \setminus \{i\}} f_{ij} = y_i \qquad \forall i \in \mathcal{I} \tag{10b}$$

$$0 \leq f_{ij} \leq I z_{ij} \qquad \forall (i, j) \in \mathcal{A}_0 \tag{10c}$$

Constraints (10a) indicate that the amount of commodities sent from node 0 is equal to the total number of items produced. Constraints (10b) ensure that if item i is produced, the amount of commodity decreases by one, whereas it does not change if i is not produced. Constraints (10c) represent the relation between the variables $\mathbf{f}$ and $\mathbf{z}$, and impose the upper bound on the amount of the commodity. If $z_{ij} = 1$, f_{ij} represents the amount of commodity that flows along the arc (i, j), whereas $f_{ij} = 0$ if $z_{ij} = 0$. To illustrate this, we recall the production plan of the period t presented in Figure 1 where the items are produced in an order of $1 - 2 - 4$. This sequence can be represented with commodity flow variables as shown in Figure 3. Because three items are produced, three units of commodity are sent from node 0 to 1, that is, $f_{01} = 3$. Then, the amount decreases by one at every node.

By replacing constraints (3g) with (10), the first single-commodity flow formulation which is denoted as (SCF1) can be obtained. The solution set of (SCF1) is defined by the set of constraints

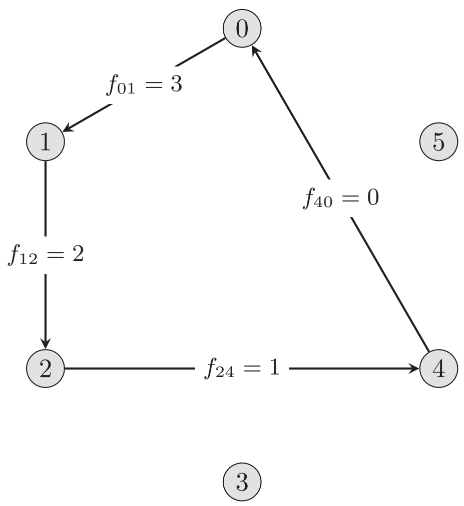

Figure 3: Illustration of the single-commodity flow formulations

$\{(3b) - (3f), (3h), (3i), (10)\}$. It is commonly known (Gouveia, 1995) that a stronger formulation denoted as (SCF2) can be obtained by replacing the bound constraints (10c) of (SCF1) with the tighter constraints (11), that is, the solution set of (SCF2) is defined by the constraints $\{(3b) - (3f), (3h), (3i), (10a) - (10b), (11)\}$.

$$z_{0i} \leq f_{0i} \leq I z_{0i} \qquad \forall i \in \mathcal{I} \tag{11a}$$

$$z_{ij} \leq f_{ij} \leq (I-1) z_{ij} \qquad \forall (i,j) \in \mathcal{A} \tag{11b}$$

$$f_{i0} = 0 \qquad \forall i \in \mathcal{I} \tag{11c}$$

Both (SCF1) and (SCF2) prevent any cycles without node 0. Furthermore, the following relations hold among (SCF1), (SCF2), and (GSEC).

Proposition 4.1. *Let* $z(\texttt{F})$ *be the LP relaxation bound obtained from the given formulation* (F). *Then,*

$$z(\texttt{SCF1}) \leq z(\texttt{SCF2}) \leq z(\texttt{GSEC}).$$

Proof. By projecting both (SCF1) and (SCF2) onto the original space, one can obtain the following inequalities (12) and (13), respectively.

$$\sum_{i \in S} y_i \leq I \Big(\sum_{(i,j) \in \delta^-(S)} z_{ij} \Big) \qquad \forall S \subseteq \mathcal{I} \tag{12}$$

$$\sum_{i \in S} y_i \leq I \Big(\sum_{(i,j) \in \delta^-(S)} z_{ij} \Big) - \sum_{i \in S} \sum_{j \in \mathcal{I} \setminus S} z_{ij} - \sum_{i \in \mathcal{I} \setminus S} \sum_{j \in S} z_{ij} \qquad \forall S \subseteq \mathcal{I} \tag{13}$$

No other inequalities can be obtained by the projection because of Hoffman's circulation theorem (Hoffman, 1976). It is trivial that inequalities (12) are dominated by inequalities (13). We show that inequalities (13) are implied by the `GSEC`. For a given $S \subseteq \mathcal{I}$, by aggregating the `GSEC` for all $i \in S$, we obtain $\sum_{i \in S} y_i \leq |S| \sum_{(i,j) \in \delta^-(S)} z_{ij}$. Therefore, it is sufficient to show that

$$\sum_{i \in S} \sum_{j \in \mathcal{I} \setminus S} z_{ij} + \sum_{i \in \mathcal{I} \setminus S} \sum_{j \in S} z_{ij} \leq (I - |S|) \sum_{(i,j) \in \delta^-(S)} z_{ij}. \tag{14}$$

When $|S| = I$, it naturally holds because $\mathcal{I} \setminus S = \emptyset$. When $|S| = I - 1$, $|\mathcal{I} \setminus S| = 1$, and let k be the only element in $\mathcal{I} \setminus S$. Then,

$$\sum_{i \in S} \sum_{j \in \mathcal{I} \setminus S} z_{ij} + \sum_{i \in \mathcal{I} \setminus S} \sum_{j \in S} z_{ij} = \sum_{i \in S} (z_{ik} + z_{ki})$$

and

$$(I - |S|) \sum_{(i,j) \in \delta^-(S)} z_{ij} = \sum_{i \in S} \sum_{j \in \mathcal{I}_0 \setminus S} z_{ij} = \sum_{i \in S} (z_{ik} + z_{i0}) = \sum_{i \in S} z_{ik} + 1 - z_{k0}$$

as $\sum_{i \in \mathcal{I}} z_{i0} = 1$. Because $\sum_{i \in S} z_{ki} + z_{k0} \leq 1$, inequality (14) also holds. Finally, when $|S| \leq I - 2$, inequality (14) can be written as

$$\sum_{i \in S} \sum_{j \in \mathcal{I} \setminus S} z_{ij} + \sum_{i \in \mathcal{I} \setminus S} \sum_{j \in S} z_{ij} \leq \sum_{(i,j) \in \delta^+(S) \cup \delta^-(S)} z_{ij} = 2 \sum_{(i,j) \in \delta^-(S)} z_{ij} \leq (I - |S|) \sum_{(i,j) \in \delta^-(S)} z_{ij},$$

and therefore, it also holds. □

4.2. Multi-commodity flow formulations

Multi-commodity flow formulations have also been studied for various routing problems. Sarin et al. (2011) presented a multi-commodity flow formulation for LSP-SQ. In contrast to single-commodity flow formulations, multi-commodity flow formulations define one commodity per item. Let us define the binary variable q_{ij}^k for $i \in \mathcal{I}_0$ and $j, k \in \mathcal{I}$ which represents whether arc (i, j) is traversed on the way from node 0 to node k. In other words, $q_{ij}^k = 1$ if the node k is visited after traversing the arc (i, j). Accordingly, the following constraints are formed.

$$\sum_{j \in \mathcal{I}} q_{0j}^k = y_k \qquad \forall k \in \mathcal{I} \tag{15a}$$

$$\sum_{j \in \mathcal{I}_0 \setminus \{k\}} q_{jk}^k = y_k \qquad \forall k \in \mathcal{I} \tag{15b}$$

$$\sum_{j \in \mathcal{I}_0 \setminus \{i\}} q_{ji}^k = \sum_{j \in \mathcal{I} \setminus \{i\}} q_{ij}^k \qquad \forall i, k \in \mathcal{I}, k \neq i \tag{15c}$$

$$q_{kj}^k = 0 \qquad \forall j, k \in \mathcal{I} \tag{15d}$$

$$0 \leq q_{ij}^k \leq z_{ij} \qquad \forall i \in \mathcal{I}_0, j, k \in \mathcal{I} \tag{15e}$$

Constraints (15a) and (15b) ensure that when item k is produced, the corresponding commodity should flow from node 0 to node k. Constraints (15c) are the flow balance constraints. Constraints (15d) and (15e) ensure that the commodity variable can have a nonzero value only when the corresponding arc is traversed. As an illustration, the production sequence in Figure 1 is represented with multi-commodity flow variables in Figure 4. Note that $\mathbf{q}_{ij} = (q_{ij}^1, q_{ij}^2, \ldots, q_{ij}^I)$. For example, because items 1, 2, and 4 are produced after traversing the arc $(0,1)$, $q_{01}^1 = q_{01}^2 = q_{01}^4 = 1$ which is concisely represented as $\mathbf{q}_{01} = (1,1,0,1,0)$. After traversing the arc $(1,2)$, nodes 2 and 4 are visited which results in $\mathbf{q}_{12} = (0,1,0,1,0)$, and so on.

By replacing constraints (3g) with the set of constraints (15), the multi-commodity flow formulation denoted as `(MCF1)` can be obtained, that is, the solution set of `(MCF1)` is defined by the set of constraints $\{(3b)-(3f),(3h),(3i),(15)\}$. It is known that by projecting `(MCF1)` onto the original space, `(GSEC)` is obtained, and they provide the same LP bounds (Padberg & Sung, 1991).

Because `(MCF1)` only uses additional variables q_{ij}^k regarding the sequencing decisions, there are no considerations for lot-sizing decisions. In this regard, to further enhance the LP relaxation bound, we derive the following additional constraints that incorporate both decisions.

$$\sum_{(i,j)\in\mathcal{A}} st_{ij} q_{ij}^k + x_k \le K y_k \qquad \forall k \in \mathcal{I} \tag{16a}$$

$$st_{ij} q_{ij}^j + st_{ji} q_{ji}^i + x_i + x_j \le K(y_i + y_j - z_{ij} - z_{ji}) \qquad \forall (i,j) \in \mathcal{A} \tag{16b}$$

It is not difficult to show that inequalities (16a) and (16b) are valid. Incorporating these inequalities to `(MCF1)`, a tighter formulation `(MCF2)` is obtained, that is, the solution set of `(MCF2)` is defined by the set of constraints $\{(3b)-(3f),(3h),(3i),(15),(16)\}$. Their strength and effectiveness are further investigated through computational experiments. Additionally, we obtain the following

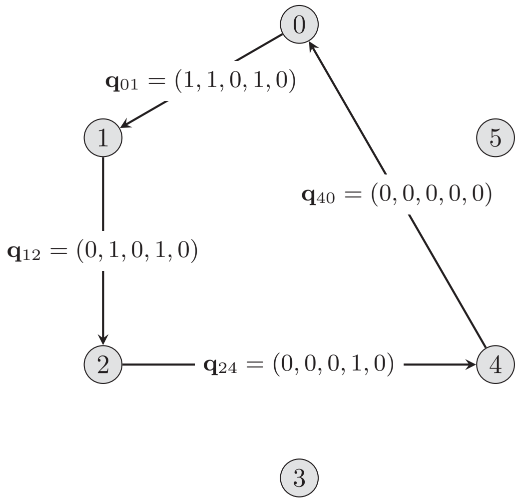

Figure 4: Illustration of the multi-commodity flow formulations

corollary.

Corollary 4.2. $z(\texttt{SCF1}) \leq z(\texttt{SCF2}) \leq z(\texttt{GSEC}) = z(\texttt{MCF1}) \leq z(\texttt{MCF2})$.

4.3. Time-flow formulations

The time-flow formulations are similar to the single-commodity flow formulations, except that they represent the flow of time instead of commodity. To present the formulations, we define a set of time-flow variables w_{ij} for $(i, j) \in \mathcal{A}_0$ which represents the remaining capacity when the setup from item i to item j begins.

$$\sum_{j \in \mathcal{I}_0 \setminus \{i\}} (w_{ji} - st_{ji} z_{ji}) - x_i = \sum_{j \in \mathcal{I}_0 \setminus \{i\}} w_{ij} \qquad \forall i \in \mathcal{I} \tag{17a}$$

$$0 \leq w_{ij} \leq K z_{ij} \qquad \forall (i, j) \in \mathcal{A}_0 \tag{17b}$$

Constraints (17a) are time-flow balance equations which indicate that, by subtracting the setup and production time for item i from the given capacity, one can obtain the remaining capacity. Constraints (17b) impose upper bounds on the time-flow variables.

This is illustrated in Figure 5 where the values of time-flow variables w_{ij} representing the remaining capacity decrease as the flow travels along the arcs. In addition, contrary to the commodity variables which only contain the sequencing information, the time-flow variables can additionally incorporate the production amounts. For instance, let us assume that the capacity and setup times be given as $K = 100$ and $st_{ij} = 15$ for all $(i, j) \in \mathcal{A}$. In addition, let the production amounts are given as $x_1 = x_2 = 25$ and $x_4 = 20$. The flow starts with the full capacity, that is, $w_{01} = 100$. By producing 25 units of item 1, the remaining capacity is $w_{12} = 75$. Then, after conducting the setup from item 1 to 2 and the production of $x_2 = 25$, the remaining capacity is $75 - 15 - 25 = 35$. This property can be beneficial in modeling LSP-SQ where both the lot-sizing and sequencing decisions should be considered.

By replacing constraints (3g) with the set of constraints (17), the first time-flow formulation `(TF1)` can be obtained. The solution set of `(TF1)` is defined by the set of constraints $\{(3b) - (3f), (3h), (3i), (17)\}$. `(TF1)` can be further strengthened with tighter bound constraints (18) instead of (17b) which we call `(TF2)`.

$$st_{ij} z_{ij} \leq w_{ij} \leq K z_{ij} \qquad \forall (i, j) \in \mathcal{A}_0,\ i \neq 0 \tag{18a}$$

$$w_{0i} = K z_{0i} \qquad \forall i \in \mathcal{I} \tag{18b}$$

Finally, we provide the relation between the time-flow formulations and `S-STAR` inequalities.

Proposition 4.3. *Projection of* $(\texttt{TF2})$ *onto the original space results in* $\texttt{S} - \texttt{STAR}$ *inequalities* (5). *Furthermore,* $z(\texttt{TF1}) \leq z(\texttt{TF2}) = z(\texttt{S} - \texttt{STAR})$.

Proof. For a given $S \subseteq \mathcal{I}$, by adding the inequalities (17a) for all $i \in S$ and rearranging the terms, one can obtain

$$\sum_{(j,i)\in E(S)\cup\delta^-(S)} st_{ji}z_{ji} + \sum_{i\in S} x_i = \sum_{i\in S}\Big(\sum_{j\in\mathcal{I}_0\setminus\{i\}} w_{ji} - \sum_{j\in\mathcal{I}_0\setminus\{i\}} w_{ij}\Big) = \sum_{(j,i)\in\delta^-(S)} w_{ji} - \sum_{(i,j)\in\delta^+(S)} w_{ij}.$$

From the bound constraints (18),

$$\sum_{(j,i)\in E(S)\cup\delta^-(S)} st_{ji}z_{ji} + \sum_{i\in S} x_i = \sum_{(j,i)\in\delta^-(S)} w_{ji} - \sum_{(i,j)\in\delta^+(S)} w_{ij} \le K \sum_{(i,j)\in\delta^-(S)} z_{ij} - \sum_{(i,j)\in\delta^+(S)} st_{ij}z_{ij}$$

can be obtained, which is the `S-STAR` inequality for $S \subseteq \mathcal{I}$. This indicates that all feasible solutions of `(TF2)` satisfy the `S-STAR` inequalities. Moreover, due to Hoffman's circulation theorem (Hoffman, 1976) it can be shown that $z($`TF2`$) = z($`S-STAR`$)$. □

There are no other dominance relations between the time-flow formulations and commodity-flow formulations. Their strengths are compared through computational experiments, as discussed in the next section.

5. Computational Experiments

5.1. Experiment settings

We stress that the aim of the computational experiments is to compare the strengths of various inequalities and formulations, rather than to solve real-world LSP-SQ instances. Thus, we compare the LP relaxation bounds of the extended formulations, that is, `(SCF1)`, `(SCF2)`, `(MCF1)`, `(MCF2)`,

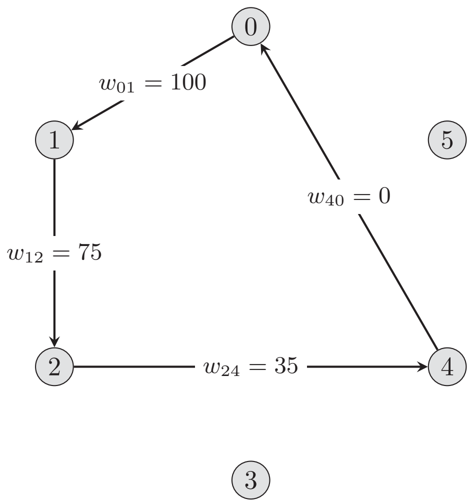

Figure 5: Illustration of the time-flow formulations

(TF1), and (TF2), and the formulations with valid inequalities, that is, (GSEC), (S-STAR), and (U-STAR).

To compute the bound value with a particular type of inequality, we exhaustively separate the violated inequalities at the root node until none is found or the objective value does not improve during the last 100 iterations. Furthermore, for the purpose of comparison, we also report the results when all three types of inequalities are separated which is denoted as (ALL).

All experiments were conducted on an Intel Core 3.10 GHz PC with 16 GB RAM under Windows 10 Pro. The separation algorithms and mathematical formulations were implemented using C++. FICO Xpress 8.12 with its default parameter settings was used as the LP solver.

We use two sets of test instances, that is, single-period and multi-period instances that are generated following the instance-generation scheme proposed by Almada-Lobo et al. (2007) which has been frequently used in the literature. The descriptions of the instances are given below.

5.1.1. Single-period instances

The type of single-period instance is defined by the combination of the number of items (I), capacity utilization parameter (ρ), setup cost parameter (θ), and upper bound parameter (β). These parameters are adopted from the study of Almada-Lobo et al. (2007), except for the last parameter which is additionally defined to determine whether there is an individual upper bound for the production amount of each item ($\beta = 1$) or not ($\beta = 0$).

The demand for item i, d_i, is generated from a discrete uniform distribution $U[40, 60]$, and capacity K is set to $I \cdot d_{avg}/\rho$ where d_{avg} is the average demand. The unit surplus and shortage costs of item i, that is, hc_i and bc_i, respectively, are generated from $U[2, 10]$. In addition, we use a negative unit production cost, i.e., profit $pc_i = -1$, to make production profitable. The setup time st_{ij} is drawn from $U[0.05K, 0.1K]$, and the setup cost is set as $sc_{ij} = \theta \cdot st_{ij}$. The upper bound u_i for item i is generated from $U[d_i + 1, K]$ if $\beta = 1$, while it is set to K if $\beta = 0$. We use the following parameters: $I \in \{5, 15, 25, 35\}$, $\rho = \{0.6, 0.8, 1\}$, $\theta \in \{50, 100\}$, $\beta \in \{0, 1\}$. For each combination, we generated 100 instances, resulting in a total of 4800 single-period instances.

5.1.2. Multi-period instances

Multi-period instances are also generated, similar to the single-period instances. The main difference is in setting the cost parameters which are set as $pc_{it} = 1$, $hc_{it} \sim U[2, 10]$, $bc_{it} \sim U[10, 50]$, and $sc_{ij} = \theta \cdot st_{ij}$. We use the following parameters: $I \in \{5, 15, 25\}$, $T \in \{5, 15, 25\}$, $\rho = \{0.6, 0.8, 1\}$, $\theta \in \{50, 100\}$, $\beta \in \{0, 1\}$. For each combination, we generated 10 instances, resulting in a total of 1080 multi-period instances.

5.2. Experiment results on single-period instances

We compared the strengths of the extended formulations and valid inequalities using the following measures:

Table 2: Test results on single-period instances: LP gap (%)

Factors		*Extended Formulations*						*Valid Inequalities*			
		(SCF1)	(SCF2)	(MCF1)	(MCF2)	**(TF1)**	**(TF2)**	(GSEC)	**(S-STAR)**	(U-STAR)	(ALL)
I	5	29.72	29.61	29.21	23.14	22.12	22.12	29.21	22.12	23.58	21.60
	15	33.66	33.65	33.42	26.52	21.77	21.77	33.42	21.77	23.14	21.04
	25	33.87	33.87	33.73	27.46	20.93	20.92	33.73	20.92	22.43	20.23
	35	34.68	34.68	34.59	28.65	21.02	21.02	34.59	21.02	22.54	20.31
ρ	60	32.55	32.52	32.31	26.10	21.24	21.23	32.31	21.23	22.70	20.60
	80	33.05	33.02	32.82	26.48	21.48	21.47	32.82	21.47	22.95	20.80
	100	33.35	33.32	33.09	26.75	21.67	21.67	33.09	21.67	23.12	20.98
θ	50	49.66	49.63	49.42	43.49	36.49	36.48	49.42	36.48	37.99	35.28
	100	16.30	16.28	16.05	9.40	6.43	6.43	16.05	6.43	7.86	6.31
β	0	48.49	48.45	48.25	36.94	29.44	29.44	48.25	29.44	33.20	29.44
	1	17.48	17.46	17.23	15.95	13.48	13.47	17.23	13.47	12.65	12.15
Average		32.98	32.95	32.74	26.44	21.46	21.46	32.74	21.46	22.92	20.79

- *LP gap* (%): $\frac{\texttt{(OPT)}-z(*)}{\texttt{(OPT)}} \times 100$,
- *Closed gap* (%): $\frac{z(*)-z\texttt{(PURE)}}{\texttt{(OPT)}-z\texttt{(PURE)}} \times 100$,

where (OPT) is an optimal objective value, (PURE) is the basic lower bound obtained using formulation (3) without GSEC (3g), and $z(*)$ is the LP relaxation bound obtained from the given formulation ($*$). Note that, therefore, (PURE) itself is not a valid formulation and is only used for comparison. The *LP gap* and *Closed gap* results for single-period instances are reported in Table 2 and 3, respectively. The test results are averaged for different parameter values, for example, $I = 5$, $I = 15$, $I = 25$, and $I = 35$. In the heading of each table, we highlight the inequalities and formulations that show the best performance using boldface.

First, the following relations established in Section 3 and 4 are verified based on the results:

- $z\texttt{(SCF1)} \leq z\texttt{(SCF2)} \leq z\texttt{(MCF1)} = z\texttt{(GSEC)} \leq z\texttt{(MCF2)}$,
- $z\texttt{(TF1)} \leq z\texttt{(TF2)} = z\texttt{(S-STAR)}$, and
- $z\texttt{(U-STAR)} \leq z\texttt{(S-STAR)}$ when $\beta = 0$.

The newly proposed formulations (TF1) and (TF2) provide considerably tighter LP bounds than the other extended formulations. They can close approximately 40% of the gap, whereas only approximately 3% can be closed by (MCF1). The effect of the constraints added in (MCF2) is considerably significant, whereas the strengthened bounds of (SCF2) and (TF2) are not particularly significant compared with (SCF1) and (TF1), respectively.

(U-STAR) provides slightly better results than (S-STAR) when $\beta = 1$, as expected. Obviously, the tightest bound is obtained when all three types of inequalities are added in (ALL). From Table 2 and 3, it can be shown that the overall performance of (S-STAR) is better than (U-STAR)

Table 3: Test results on single-period instances: Closed gap (%)

Factors		*Extended Formulations*						*Valid Inequalities*			
		(SCF1)	(SCF2)	(MCF1)	(MCF2)	(TF1)	**(TF2)**	(GSEC)	**(S-STAR)**	(U-STAR)	(ALL)
I	5	2.27	2.84	4.97	27.80	30.27	30.30	4.97	30.30	24.52	32.49
	15	0.59	0.63	3.19	28.25	40.47	40.50	3.19	40.50	38.49	43.99
	25	0.17	0.18	2.05	24.99	43.74	43.77	2.05	43.77	41.61	47.24
	35	0.11	0.11	1.55	22.73	45.42	45.45	1.55	45.45	43.51	48.96
ρ	60	0.82	0.95	2.89	26.16	39.79	39.81	2.89	39.81	36.84	42.90
	80	0.65	0.80	2.91	25.29	39.56	39.59	2.91	39.59	36.67	42.86
	100	0.89	1.07	3.03	26.39	40.58	40.60	3.03	40.60	37.59	43.74
θ	50	0.24	0.33	1.03	10.94	24.80	24.84	1.03	24.84	24.05	28.94
	100	1.33	1.55	4.85	40.95	55.15	55.17	4.85	55.17	50.02	57.39
β	0	0.51	0.67	1.46	31.66	47.03	47.03	1.46	47.03	37.94	47.03
	1	1.06	1.21	4.41	20.23	32.92	32.98	4.41	32.98	36.13	39.31
Average		0.78	0.94	2.94	25.94	39.97	40.00	2.94	40.00	37.03	43.17

Table 4: Test results on single-period instances: Ratio of instances with LP gap below certain value

LP Gap	*Extended Formulations*						*Valid Inequalities*			
	(SCF1)	(SCF2)	(MCF1)	(MCF2)	(TF1)	**(TF2)**	(GSEC)	**(S-STAR)**	(U-STAR)	(ALL)
$= 0\%$	6.04	6.04	6.35	8.00	8.56	8.58	6.35	8.58	8.27	9.10
$< 1\%$	7.00	7.06	7.56	10.10	12.06	12.06	7.56	12.06	11.96	13.06
$< 5\%$	15.17	15.21	16.06	21.73	26.02	26.02	16.06	26.02	24.38	26.58
$< 10\%$	24.54	24.54	25.04	29.81	38.23	38.25	25.04	38.25	32.73	38.73

and `(GSEC)`. At the same time, however, we found that there exist no dominance relationships between these inequalities. For some instances, `(S-STAR)` performed better than the others, whereas `(U-STAR)` and `(GSEC)` performed better for other instances.

Additionally, we report the ratio of instances in which the LP gap is below certain values in Table 4. For example, it is reported that `(SCF1)` provides LP bound which is the same as the optimal value for about 6% of instance. For about 15% of instances, it provides LP gap less than 5%, and so on. Similar to the above results, `(TF2)` provides the best results among the extended formulations. Using `(TF2)`, in approximately 8.6% of the instances, one can obtain the same LP bound as the optimal objective value. Moreover, in approximately 40% of the instances, the LP gap is smaller than 10%.

In Table 5, we present the sizes of the extended formulations relative to the smallest formulation `(SCF1)`. As expected, the size of the multi-commodity flow formulations is the largest, and the relative size increases with I. In particular, owing to additional constraints, `(MCF2)` has the largest number of variables and constraints. This results in an increase in computation time, as demonstrated in Table 6.

The number of separated inequalities is reported in Table 7. On average, approximately four

Table 5: Test results on single-period instances: Relative formulation size

I	*Variables*			*Constraints*			
	(SCF1) =(TF1)	(SCF2) =(TF2)	(MCF1) =(MCF2)	(SCF1) =(TF1)	(SCF2) =(TF2)	(MCF1)	(MCF2)
5	1.00	1.00	2.67	1.00	1.47	3.81	4.66
15	1.00	1.00	7.43	1.00	1.74	12.88	14.31
25	1.00	1.00	12.36	1.00	1.83	22.58	24.21
35	1.00	1.00	17.33	1.00	1.87	32.43	34.15
Average	1.00	1.00	9.95	1.00	1.73	17.93	19.33

Table 6: Test results on single-period instances: Computation time

I	*Computation Time (s)*					
	(SCF1)	(SCF2)	(TF1)	(TF2)	(MCF1)	(MCF2)
5	0.002	0.002	0.002	0.003	0.003	0.004
15	0.005	0.006	0.007	0.008	0.030	0.144
25	0.012	0.015	0.026	0.032	0.292	1.579
35	0.028	0.035	0.071	0.073	2.297	11.368
Average	0.012	0.014	0.027	0.029	0.656	3.274

Table 7: Test results on single-period instances: Number of added inequalities

Factors		*Number of added inequalities*			
		(GSEC)	(S-STAR)	(U-STAR)	(ALL)
I	5	0.8	1.9	0.9	3.0
	15	4.0	10.3	3.4	14.8
	25	5.7	22.8	6.7	30.9
	35	6.2	38.0	10.2	49.0
ρ	60	4.1	18.3	5.0	23.8
	80	4.2	18.0	4.9	23.9
	100	4.2	18.5	6.0	25.6
θ	50	4.7	29.6	8.2	38.1
	100	3.6	6.9	2.3	10.7
β	0	4.7	16.9	2.7	22.5
	1	3.6	19.6	7.9	26.4
Average		4.2	18.2	5.3	24.4

times more inequalities are added when using the `S-STAR` inequality than when using `GSEC`. When using all three types, `(ALL)`, the number of added inequalities is smaller than the sum of that when each type is used individually.

Table 8: Test results on multi-period instances: LP strength

Factors		*Extended Formulations*						*Valid Inequalities*			
		(SCF1)	(SCF2)	(MCF1)	(MCF2)	(TF1)	(TF2)	(GSEC)	(S-STAR)	(U-STAR)	(ALL)
I	5	92.98	92.98	93.27	94.12	94.57	94.59	93.27	94.59	96.24	97.00
	15	97.31	97.31	97.43	97.69	98.06	98.07	97.43	98.07	98.35	98.63
	25	97.36	97.36	97.45	97.57	97.77	97.77	97.45	97.77	97.87	98.03
T	5	97.14	97.14	97.32	97.62	97.82	97.82	97.32	97.82	98.30	98.72
	15	96.06	96.06	96.22	96.64	97.01	97.02	96.22	97.02	97.73	98.12
	25	94.45	94.45	94.61	95.12	95.58	95.59	94.61	95.59	96.43	96.82
ρ	60	99.58	99.58	99.70	99.72	99.62	99.62	99.70	99.62	99.80	99.93
	80	97.95	97.95	98.14	98.19	98.08	98.08	98.14	98.08	98.65	98.91
	100	90.11	90.12	90.32	91.47	92.70	92.71	90.32	92.71	94.02	94.83
θ	50	96.21	96.22	96.40	96.77	97.06	97.06	96.40	97.06	97.64	98.04
	100	95.55	95.55	95.70	96.15	96.55	96.55	95.70	96.55	97.34	97.74
β	0	98.53	98.53	98.53	99.25	99.94	99.95	98.53	99.95	99.74	99.95
	1	93.24	93.24	93.57	93.67	93.66	93.67	93.57	93.67	95.24	95.83
Average		95.88	95.88	96.05	96.46	96.80	96.81	96.05	96.81	97.49	97.89

5.3. Experiment results on multi-period instances

For the results of the multi-period instances, we report the relative strength of the formulations relative to the bound which can be obtained when we know the ideal formulation of the single-period substructure, that is, $conv(\mathcal{X})$. This bound is denoted by $z(\texttt{IDEAL})$. To calculate $z(\texttt{IDEAL})$, we define a pattern-based formulation whose LP relaxation is solved using a column generation procedure. Because they are outside the scope of this study, we do not provide detailed descriptions of the pattern-based formulation and column generation procedure (see Appendix B). We denote the strength of a formulation (F) as $\frac{z(\texttt{F})}{z(\texttt{IDEAL})} \times 100$ and present the corresponding results in Table 8.

Similar to the results of the single-period instances, the newly proposed formulations and inequalities are successful in providing tight bounds. On average, they provide only approximately 3.2% less tight bounds relative to (IDEAL). In particular, when $\beta = 0$, they provide almost the same LP bound as that of the ideal formulation. In addition, we report the ratio of instances whose LP strength is above a certain value in Table 9. For example, it is reported that (SCF1) provides LP strength of 100% for about 4.7% of instances. It provides LP strength greater than 99% for more than 50% of instances, and so on. In approximately 40% of the instances, (TF2) and (S-STAR) provide the same LP bound as that of the ideal formulation. In addition, the difference is less than 1% for more than 65% of instances.

In Table 10, we present the sizes of the formulations relative to the smallest one (SCF1). As expected, the multi-commodity flow formulations are the largest. Particularly, when $I = 25$, (MCF2) requires 12.4 times more variables and 23.5 times more constraints than (SCF1) which may be prohibitively large. The average computation times for different problem dimensions are shown in Table 11. This result also demonstrates that multi-commodity formulations are not viable

Table 9: Test results on multi-period instances: Ratio of instances with LP strength above certain value

LP Strength	*Extended Formulations*						*Valid Inequalities*			
	(SCF1)	(SCF2)	(MCF1)	(MCF2)	(TF1)	(TF2)	(GSEC)	(S-STAR)	(U-STAR)	(ALL)
= 100%	4.72	4.72	4.72	21.57	34.26	41.30	4.72	41.30	14.44	41.67
> 99%	51.11	51.11	53.70	55.83	66.20	66.20	53.70	66.20	67.22	74.07
> 95%	74.35	74.35	74.81	80.56	80.28	80.28	74.81	80.28	83.43	84.26
> 90%	88.06	88.06	88.43	88.80	88.89	88.89	88.43	88.89	90.28	91.76

Table 10: Test results on multi-period instances: Relative formulation size

Factors		*Variables*			*Constraints*			
		(SCF1) =(TF1)	(SCF2) =(TF2)	(MCF1) =(MCF2)	(SCF1) =(TF1)	(SCF2) =(TF2)	(MCF1)	(MCF2)
I	5	1.00	1.00	2.67	1.00	1.43	3.59	4.37
	15	1.00	1.00	7.43	1.00	1.71	12.37	13.75
	25	1.00	1.00	12.36	1.00	1.81	21.95	23.53
T	5	1.00	1.00	7.49	1.00	1.65	12.69	13.94
	15	1.00	1.00	7.49	1.00	1.65	12.62	13.87
	25	1.00	1.00	7.49	1.00	1.65	12.61	13.85
Average		1.00	1.00	7.49	1.00	1.65	12.64	13.89

Table 11: Test results on multi-period instances: Computation time

Factors		*Computation Time (s)*					
		(SCF1)	(SCF2)	(TF1)	(TF2)	(MCF1)	(MCF2)
I	5	0.01	0.02	0.02	0.02	0.03	0.05
	15	0.12	0.18	0.16	0.26	1.40	4.25
	25	0.53	0.96	0.92	1.36	15.07	31.89
T	5	0.04	0.06	0.06	0.10	0.62	1.90
	15	0.21	0.36	0.36	0.55	6.10	10.47
	25	0.42	0.75	0.68	0.99	9.79	23.81
Average		0.22	0.39	0.37	0.55	5.50	12.06

options when the problem dimension increases.

Regarding the valid inequalities, as reported in Table 12, the numbers of added inequalities of (S-STAR) and (U-STAR) are much larger than that of GSEC which can be regarded as a trade-off for the tighter bound. We observe that most of the S-STAR and U-STAR inequalities added after earlier iterations do not contribute significantly to the strengthening of the bounds. Nevertheless, owing to the original purpose of the experiments, we did not terminate the separation which led to a number of additional inequalities, most of which were not significant. In this regard, if these inequalities are used as cutting planes in tree-search algorithms, more detailed analysis and further studies on their effects are required.

Table 12: Test results on multi-period instances: Number of added inequalities

Factors		*Number of added inequalities*			
		(GSEC)	(S-STAR)	(U-STAR)	(ALL)
I	5	4.9	26.9	27.8	66.9
	15	14.4	147.0	152.1	300.6
	25	25.5	296.9	382.3	574.5
T	5	5.6	69.1	77.0	125.4
	15	15.7	154.8	172.2	309.6
	25	23.6	247.0	313.1	507.0
ρ	60	11.3	41.5	47.5	101.8
	80	16.2	67.7	69.9	152.0
	100	17.3	361.6	444.9	688.2
θ	50	18.3	153.3	180.7	309.4
	100	11.6	160.6	194.2	318.7
β	0	0.0	215.3	263.8	376.0
	1	29.9	98.6	111.1	252.0
Average		15.0	156.9	187.4	314.0

6. Conclusion

In this study, we address the lot-sizing and scheduling problem with sequence-dependent setups and its single-period substructure. We present new families of valid inequalities and extended formulations which demonstrate distinct advantages in tightening the LP relaxation bounds, compared with the existing ones. Furthermore, we discuss the theoretical strengths of the proposed inequalities and identify their facet-defining conditions; we also demonstrate that they can be separated in polynomial time.

The results of this study can be utilized in various ways. The proposed valid inequalities can be utilized to devise efficient solution algorithms for LSP-SQ. For instance, along with the known results such as the (l, S)-inequality, one can propose a branch-and-cut algorithm. Specifically, when used as cuts in the branch-and-cut algorithm, detailed algorithmic components such as the number of cuts added in one iteration, the frequency of adding cuts during the tree search, and the order in which cuts are added, affect the algorithm performance. Therefore, considering the algorithmic elements, additional extensive computational experiments are required in future research.

In addition, the proposed time-flow formulation can be adapted to solve real-world problems. Heuristic algorithms based on mathematical programming methodologies such as LP or MIP (matheuristics) have been popularly used to solve LSP-SQ occurs in various industries. The performance of these heuristics is significantly affected by the tightness of the base formulation. At the same time, the formulation with a large number of variables and constraints might be prohibitive because it may have to be solved many times repeatedly. Considering this trade-off between the formulation size and tightness, the proposed time-flow formulation can be utilized to improve the performance of various matheuristics.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (No. 2021R1A2C2005531). The authors would like to thank the Institute for Industrial Systems Innovation of Seoul National University for the administrative support. We also are grateful to the editor and anonymous referees, whose comments and suggestions enabled us to improve the paper significantly.

References

Almada-Lobo, B., Klabjan, D., Antónia Carravilla, M., & Oliveira, J. F. (2007). Single machine multi-product capacitated lot sizing with sequence-dependent setups. *International Journal of Production Research*, *45*, 4873–4894.

Barany, I., Van Roy, T. J., & Wolsey, L. A. (1984). Strong formulations for multi-item capacitated lot sizing. *Management Science*, *30*, 1255–1261.

Bitran, G. R., & Yanasse, H. H. (1982). Computational complexity of the capacitated lot size problem. *Management Science*, *28*, 1174–1186.

Carvalho, D. M., & Nascimento, M. C. (2022). Hybrid matheuristics to solve the integrated lot sizing and scheduling problem on parallel machines with sequence-dependent and non-triangular setup. *European Journal of Operational Research*, *296*, 158–173.

Copil, K., Wörbelauer, M., Meyr, H., & Tempelmeier, H. (2017). Simultaneous lotsizing and scheduling problems: a classification and review of models. *OR Spectrum*, *39*, 1–64.

Desaulniers, G., Desrosiers, J., & Solomon, M. M. (2006). *Column generation* volume 5. Springer Science & Business Media.

Doostmohammadi, M., & Akartunalı, K. (2018). Valid inequalities for two-period relaxations of big-bucket lot-sizing problems: Zero setup case. *European Journal of Operational Research*, *267*, 86–95.

Eppen, G. D., & Martin, R. K. (1987). Solving multi-item capacitated lot-sizing problems using variable redefinition. *Operations Research*, *35*, 832–848.

Fleischmann, B., & Meyr, H. (1997). The general lotsizing and scheduling problem. *Operations-Research-Spektrum*, *19*, 11–21.

Gavish, B., & Graves, S. C. (1978). *The travelling salesman problem and related problems*. Technical Report Massachusetts Institute of Technology, Operations Research Center. Working Paper OR-078-78.

Gouveia, L. (1995). A result on projection for the vehicle routing problem. *European Journal of Operational Research*, *85*, 610–624.

Guimarães, L., Klabjan, D., & Almada-Lobo, B. (2013). Pricing, relaxing and fixing under lot sizing and scheduling. *European Journal of Operational Research*, *230*, 399–411.

Guimarães, L., Klabjan, D., & Almada-Lobo, B. (2014). Modeling lotsizing and scheduling problems with sequence dependent setups. *European Journal of Operational Research*, *239*, 644–662.

Gupta, D., & Magnusson, T. (2005). The capacitated lot-sizing and scheduling problem with sequence-dependent setup costs and setup times. *Computers and Operations Research*, *32*, 727–747.

Hoffman, A. J. (1976). Total unimodularity and combinatorial theorems. *Linear Algebra and its Applications*, *13*, 103–108.

Krarup, J., & Bilde, O. (1977). Plant location, set covering and economic lot size: An $O(mn)$-algorithm for structured problems. In *Numerische Methoden bei Optimierungsaufgaben Band 3*.

Küçükyavuz, S., & Pochet, Y. (2009). Uncapacitated lot sizing with backlogging: The convex hull. *Mathematical Programming*, *118*, 151–175.

Lee, Y., & Lee, K. (2020). Lot-sizing and scheduling in flat-panel display manufacturing process. *Omega*, *93*, 102036.

Lee, Y., & Lee, K. (2021). New integer optimization models and an approximate dynamic programming algorithm for the lot-sizing and scheduling problem with sequence-dependent setups. *European Journal of Operational Research*, *in press*.

Letchford, A. N., Eglese, R. W., & Lysgaard, J. (2002). Multistars, partial multistars and the capacitated vehicle routing problem. *Mathematical Programming*, *94*, 21–40.

Letchford, A. N., & Salazar-González, J. J. (2006). Projection results for vehicle routing. *Mathematical Programming*, *105*, 251–274.

Letchford, A. N., & Salazar-González, J. J. (2015). Stronger multi-commodity flow formulations of the Capacitated Vehicle Routing Problem. *European Journal of Operational Research*, *244*, 730–738.

Leung, J. M., Magnanti, T. L., & Vachani, R. (1989). Facets and algorithms for capacitated lot sizing. *Mathematical programming*, *45*, 331–359.

Manne, A. S. (1958). Programming of economic lot sizes. *Management science*, *4*, 115–135.

Miller, A. J., Nemhauser, G. L., & Savelsbergh, M. W. (2003a). A multi-item production planning model with setup times: algorithms, reformulations, and polyhedral characterizations for a special case. *Mathematical programming*, *95*, 71–90.

Miller, A. J., Nemhauser, G. L., & Savelsbergh, M. W. (2003b). On the polyhedral structure of a multi–item production planning model with setup times. *Mathematical Programming*, *94*, 375–405.

Miller, C. E., Tucker, A. W., & Zemlin, R. A. (1960). Integer programming formulation of traveling salesman problems. *Journal of the ACM*, *7*, 326–329.

Nemhauser, G., & Wolsey, L. (1988). *Integer and Combinatorial Optimization*. John Wiley & Sons, Ltd.

Padberg, M., & Sung, T.-Y. (1991). An analytical comparison of different formulations of the travelling salesman problem. *Mathematical Programming*, *52*, 315–357.

Pochet, Y., & Wolsey, L. A. (2006). *Production planning by mixed integer programming*. Springer Science & Business Media.

Quadt, D., & Kuhn, H. (2008). Capacitated lot-sizing with extensions: a review. *4OR*, *6*, 61–83.

Ríos-Solís, Y. Á., Ibarra-Rojas, O. J., Cabo, M., & Possani, E. (2020). A heuristic based on mathematical programming for a lot-sizing and scheduling problem in mold-injection production. *European Journal of Operational Research*, *284*, 861–873.

Sarin, S. C., Sherali, H. D., & Yao, L. (2011). New formulation for the high multiplicity asymmetric traveling salesman problem with application to the chesapeake problem. *Optimization Letters*, *5*, 259–272.

Toth, P., & Vigo, D. (2002). *The vehicle routing problem*. SIAM.

Van Vyve, M. (2007). Algorithms for single-item lot-sizing problems with constant batch size. *Mathematics of Operations Research*, *32*, 594–613.

Wolsey, L. A. (2020). *Integer programming*. John Wiley & Sons.

Zhu, X., & Wilhelm, W. E. (2006). Scheduling and lot sizing with sequence-dependent setup: A literature review. *IIE Transactions*, *38*, 987–1007.

Appendix A. Proof of Propositions

We note that, for proofs in Appendix A, we only exhibit the values of some variables which are relevant for the sake of simplicity. The variables $\mathbf{x}, \mathbf{y}, \mathbf{z}$ are assumed to zero unless otherwise mentioned. The values of $\mathbf{s}^+$ and $\mathbf{s}^-$ are automatically set with respect to the corresponding $\mathbf{x}$ values and constraints (3b).

Proof of Proposition 2.1. The number of the total variables is I^2+5I. As there are $3I+1$ equality constraints (3b) and (3e) – (3f) which are linearly independent, $dim(conv(\mathcal{X})) \leq I^2 + 2I - 1$. Therefore, it is sufficient to find $I^2 + 2I$ linearly independent points. Because there are I extreme rays of form $s_i^+ = s_i^- = 1, \forall i \in \mathcal{I}$, it is sufficient to find $I^2 + I$ points.

- For each $i \in \mathcal{I}$, $z_{0i} = y_i = z_{i0} = 1$. ($I$ points)
- For each $i \in \mathcal{I}$, $z_{0i} = y_i = z_{i0} = 1$ and $x_i = u_i$. (I points)
- For each $(i,j) \in \mathcal{A}$, $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$, $x_i = \min\{u_i, (K - st_{ij})/2\}$, and $x_j = \min\{u_j, (K - st_{ij})/2\}$. ($I^2 - I$ points)

It is obvious that the above $I^2 + I$ points are linearly independent. □

We use the following Lemma 1 from Nemhauser & Wolsey (1988) to prove other propositions.

Lemma 1 (Nemhauser & Wolsey (1988)). *Given a polyhedron $P = \{x \in \mathbb{R}^n : Ax \leq b\}$, let $(A^=, b^=)$ be the equality set of $P \subseteq \mathbb{R}^n$ and $M^=$ be the corresponding constraint index set. Also, let $F = \{x \in P : \pi x = \pi_0\}$ be a proper face of P. The following two statements are equivalent:*

1. *F is a facet of P.*
2. *If $\lambda x = \lambda_0$ for all $x \in F$, then*

$$(\lambda, \lambda_0) = (u\pi + vA^=, u\pi_0 + vb^=) \text{ for some } u \in \mathbb{R}^1 \text{ and some } v \in \mathbb{R}^{|M^=|}.$$

Proof of Proposition 2.2. Let us consider a hyperplane $\sum_{i\in\mathcal{I}}(\alpha_i x_i + \beta_i^+ s_i^+ + \beta_i^- s_i^- + \gamma_i y_i) + \sum_{(i,j)\in\mathcal{A}_0} \delta_{ij} z_{ij} = \pi_0$ which contains all points $(\mathbf{x}, \mathbf{s}^+, \mathbf{s}^-, \mathbf{y}, \mathbf{z})$ in the face defined by constraint (3c). We show that $(\boldsymbol{\alpha}, \boldsymbol{\beta}^+, \boldsymbol{\beta}^-, \boldsymbol{\gamma}, \boldsymbol{\delta})$ is the sum of a scalar multiple of the coefficients in constraint (3c) and the equality system.

Firstly, without loss of generality, we can let $\gamma_i = 0$ for all $i \in \mathcal{I}$ because y_i can be replaced with $\sum_{j\in\mathcal{I}_0} z_{ij}$. Moreover, as there are I extreme rays of form $s_i^+ = s_i^- = 1$, we can let $\beta_i^+ + \beta_i^- = 0$. Due to constraints (3b), $\beta_i^+ s_i^+ + \beta_i^- s_i^- = \beta_i^+(s_i^+ - s_i^-) = \beta_i^+(d_i - x_i)$. Therefore, we also can let $\beta_i^+ = \beta_i^- = 0$ for all $i \in \mathcal{I}$ without loss of generality. Now, consider the following points which satisfy constraint (3c) at equality:

- For $i \in \mathcal{I}$, let $z_{0i} = y_i = z_{i0} = 1$ and $x_i = K$. Then, $\alpha_i K + \delta_{0i} + \delta_{i0} = \pi_0$. Consequently, $\delta_{0i} = \pi_0 - \alpha_i K - \delta_{i0}$.
- For $(i, j) \in \mathcal{A}$, let $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$. Then, $x_i + x_j = K - st_{ij}$ and the following two cases are possible.

  1. $(x_i, x_j) = (0, K - st_{ij})$, then $\alpha_j(K - st_{ij}) + \delta_{0i} + \delta_{ij} + \delta_{j0} = \pi_0$.
  2. $(x_i, x_j) = (K - st_{ij}, 0)$, then $\alpha_i(K - st_{ij}) + \delta_{0i} + \delta_{ij} + \delta_{j0} = \pi_0$.

  From the above cases, $\alpha_i = \alpha_j = \alpha^*$ and

$$\begin{aligned}\delta_{ij} &= \pi_0 - \delta_{0i} - \delta_{j0} - \alpha^*(K - st_{ij}) \\ &= \pi_0 - (\pi_0 - \alpha^* K - \delta_{i0}) - \delta_{j0} - \alpha^*(K - st_{ij}) \\ &= \delta_{i0} - \delta_{j0} + \alpha^* st_{ij}.\end{aligned}$$

Therefore, $\sum_{i\in\mathcal{I}}(\alpha_i x_i + \beta_i^+ s_i^+ + \beta_i^- s_i^- + \gamma_i y_i) + \sum_{(i,j)\in\mathcal{A}_0} \delta_{ij} z_{ij} = \pi_0$ can be written as follows:

$$(\Leftrightarrow) \quad \sum_{i\in\mathcal{I}} \alpha^* x_i + \sum_{i\in\mathcal{I}} \Big(\delta_{i0} z_{i0} + \delta_{0i} z_{0i}\Big) + \sum_{(i,j)\in\mathcal{A}} \delta_{ij} z_{ij} = \pi_0$$

$$(\Leftrightarrow) \quad \sum_{i\in\mathcal{I}} \alpha^* x_i + \sum_{i\in\mathcal{I}} \Big(\delta_{i0} z_{i0} + (\pi_0 - \alpha^* K - \delta_{i0}) z_{0i}\Big) + \sum_{(i,j)\in\mathcal{A}} (\delta_{i0} - \delta_{j0} + \alpha^* st_{ij}) z_{ij} = \pi_0$$

$$(\Leftrightarrow) \quad \alpha^*\Big(\sum_{i\in\mathcal{I}} (x_i - K z_{0i}) + \sum_{(i,j)\in\mathcal{A}} st_{ij} z_{ij}\Big) + \sum_{i\in\mathcal{I}} \delta_{i0}\Big(z_{i0} - z_{0i} + \sum_{j\in\mathcal{I}} \big(z_{ij} - z_{ji}\big)\Big) + \pi_0(\sum_{i\in\mathcal{I}} z_{0i}) = \pi_0$$

$$(\Leftrightarrow) \quad \alpha^*\Big(\sum_{i\in\mathcal{I}} x_i + \sum_{(i,j)\in\mathcal{A}} st_{ij} z_{ij} - K\Big) + \sum_{i\in\mathcal{I}} \delta_{i0}(\sum_{j\in\mathcal{I}_0} z_{ij} - \sum_{j\in\mathcal{I}_0} z_{ji}) + \pi_0(\sum_{i\in\mathcal{I}} z_{0i} - 1) = 0$$

In the last equation, the first term is a scalar multiplication of the constraint (3c), whereas the second and third terms are scalar multiplications of the equality set. Therefore, from Lemma 1, it is shown that constraint (3c) is a facet-defining inequality. □

Proof of Proposition 2.3. For a given $i \in \mathcal{I}$, consider the following points:

- $z_{0i} = y_i = z_{i0} = 1$ and $x_i = u_i$. (1 point)
- For each $j \in \mathcal{I} \setminus \{i\}$, $z_{0j} = y_j = z_{j0} = 1$ and $x_j = 0$. ($I - 1$ points)
- For each $j \in \mathcal{I} \setminus \{i\}$, $z_{0j} = y_j = z_{j0} = 1$ and $x_j = u_j$. ($I - 1$ points)
- For each $j \in \mathcal{I} \setminus \{i\}$, $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$, $x_i = u_i$, and $x_j = \min\{K - st_{ij} - u_i, u_j\}$. ($I - 1$ points)
- For each $j \in \mathcal{I} \setminus \{i\}$, $z_{0j} = y_j = z_{ji} = y_i = z_{i0} = 1$, $x_i = u_i$, and $x_j = \min\{K - st_{ji} - u_i, u_j\}$. ($I - 1$ points)

- For each $(j,k) \in \mathcal{A}$ such that $j \neq i$ and $k \neq i$, $z_{0j} = y_j = z_{jk} = y_k = z_{k0} = 1$, $x_j = \min\{u_j, (K - st_{jk})/2\}$, and $x_k = \min\{u_k, (K - st_{jk})/2\}$. $((I-1)(I-2)$ points)

There are total I^2+I-1 linearly independent points. Together with I linearly independent extreme lays of form $s_i^+ = s_i^- = 1$ for all $i \in \mathcal{I}$, it is demonstrated that constraint (3d) is a facet defining inequality of $conv(\mathcal{X})$. □

Proof of Proposition 2.4. For a given $i \in \mathcal{I}$, consider the following points:

- $z_{0i} = y_i = z_{i0} = 1$ and $x_i = K$. (1 point)
- For each $j \in \mathcal{I} \setminus \{i\}$, $z_{0j} = y_j = z_{j0} = 1$ and $x_j = 0$. $(I-1$ points)
- For each $j \in \mathcal{I} \setminus \{i\}$, $z_{0j} = y_j = z_{j0} = 1$ and $x_j = u_j$. $(I-1$ points)
- For each $j \in \mathcal{I} \setminus \{i\}$, $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$ and $x_i = K - st_{ij}$. $(I-1$ points)
- For each $j \in \mathcal{I} \setminus \{i\}$, $z_{0j} = y_j = z_{ji} = y_i = z_{i0} = 1$ and $x_i = K - st_{ji}$. $(I-1$ points)
- For each $(j,k) \in \mathcal{A}$ such that $j \neq i$ and $k \neq i$, $z_{0j} = y_j = z_{jk} = y_k = z_{k0} = 1$, $x_j = \min\{u_j, (K - st_{jk})/2\}$ and $x_k = \min\{u_k, (K - st_{jk})/2\}$. $((I-1)(I-2)$ points)

There are total $I^2 + I - 1$ linearly independent points. Together with I linearly independent extreme lays of form $s_i^+ = s_i^- = 1$ for all $i \in \mathcal{I}$, it is demonstrated that constraint (4) defines a facet of $conv(\mathcal{X}_0)$. □

Proof of Proposition 3.2. Let us consider a hyperplane $\sum_{i\in\mathcal{I}} \alpha_i x_i + \sum_{(i,j)\in\mathcal{A}_0} \delta_{ij} z_{ij} = \pi_0$ which contains all points $(\mathbf{x}, \mathbf{s}^+, \mathbf{s}^-, \mathbf{y}, \mathbf{z})$ in the face defined by inequality (5), given $S \subseteq \mathcal{I}$. As shown in the proof of Proposition 2.2, we can assume that the coefficients of $\mathbf{s}^+$, $\mathbf{s}^-$, and $\mathbf{y}$ is zero. Consider the following points which satisfy inequality (5) at equality, given S:

- For each $i \in S$, $z_{0i} = y_i = z_{i0} = 1$. Then, $x_i = K$ and $\alpha_i K + \delta_{0i} + \delta_{i0} = \pi_0$, that is, $\delta_{i0} = \pi_0 - \delta_{0i} - \alpha_i K$ for all $i \in S$.
- For each $i \in \mathcal{I} \setminus S$, $z_{0i} = y_i = z_{i0} = 1$. Then, x_i can have any value between 0 and K which indicates $\alpha_i = 0$ and $\delta_{i0} = \pi_0 - \delta_{0i}$ for all $i \in \mathcal{I} \setminus S$.
- For each $(i,j) \in E(S)$, $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$. In this case, x_i and x_j can have any values satisfying $x_i + x_j = K - st_{ij}$. There exist many combinations satisfying $x_i + x_j = K - st_{ij}$. Therefore, $\alpha_i = \alpha_j = \alpha^*$ and $\delta_{ij} = \pi_0 - \alpha^*(K - st_{ij}) - \delta_{0i} - \delta_{j0} = \alpha^* st_{ij} - \delta_{0i} + \delta_{0j}$ for all $(i,j) \in E(S)$.
- For each $(i,j) \in E(S : \mathcal{I} \setminus S)$, $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$ and $x_i = K - st_{ij}$. Therefore, $\alpha^*(K - st_{ij}) + \delta_{0i} + \delta_{ij} + \delta_{j0} = \pi_0$, that is, $\delta_{ij} = \alpha^*(st_{ij} - K) - \delta_{0i} + \delta_{0j}$ for all $(i,j) \in E(S : \mathcal{I} \setminus S)$.
- For each $(i,j) \in E(\mathcal{I} \setminus S : S)$, $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$ and $x_j = K - st_{ij}$. Therefore, $\alpha^*(K - st_{ij}) + \delta_{0i} + \delta_{ij} + \delta_{j0} = \pi_0$, that is, $\delta_{ij} = \alpha^* st_{ij} - \delta_{0i} + \delta_{0j}$ for all $(i,j) \in E(\mathcal{I} \setminus S : S)$.

- For each $(i,j) \in E(\mathcal{I} \setminus S)$, $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$. In this case, $\delta_{ij} = \pi_0 - \delta_{0i} - \delta_{j0} = -\delta_{0i} + \delta_{0j}$ for all $(i,j) \in E(\mathcal{I} \setminus S : S)$.

The term $\sum_{(i,j)\in\mathcal{A}_0} \delta_{ij} z_{ij}$ can be decomposed as

$$\underbrace{\sum_{i\in\mathcal{I}}(\delta_{i0}z_{i0} + \delta_{0i}z_{0i})}_{(i)} + \underbrace{\sum_{(i,j)\in E(S)} \delta_{ij}z_{ij}}_{(ii)} + \underbrace{\sum_{(i,j)\in E(S:\mathcal{I}\setminus S)} \delta_{ij}z_{ij}}_{(iii)} + \underbrace{\sum_{(i,j)\in E(\mathcal{I}\setminus S:S)} \delta_{ij}z_{ij}}_{(iv)} + \underbrace{\sum_{(i,j)\in E(\mathcal{I}\setminus S)} \delta_{ij}z_{ij}}_{(v)}.$$

The decomposed terms (i) to (v) can be stated as follows:

$$\begin{aligned}(i) =& \sum_{i\in S}(\delta_{i0}z_{i0} + \delta_{0i}z_{0i}) + \sum_{i\in\mathcal{I}\setminus S}(\delta_{i0}z_{i0} + \delta_{0i}z_{0i}) \\ =& \sum_{i\in S}\big((\pi_0 - \delta_{0i} - \alpha^* K)z_{i0} + \delta_{0i}z_{0i}\big) + \sum_{i\in\mathcal{I}\setminus S}\big((\pi_0 - \delta_{0i})z_{i0} + \delta_{0i}z_{0i}\big) \\ =& \pi_0 \sum_{i\in\mathcal{I}} z_{i0} + \sum_{i\in\mathcal{I}} \delta_{0i}(z_{0i} - z_{i0}) - \alpha^* K \sum_{i\in S} z_{i0}\end{aligned}$$

$$(ii) = \sum_{(i,j)\in E(S)} \big(\alpha^* st_{ij} - \delta_{0i} + \delta_{0j}\big)z_{ij} = \alpha^* \sum_{(i,j)\in E(S)} st_{ij}z_{ij} + \sum_{i\in S}\delta_{0i}\Big(\sum_{j\in S} z_{ji} - \sum_{j\in S} z_{ij}\Big)$$

$$\begin{aligned}(iii) =& \sum_{(i,j)\in E(S:\mathcal{I}\setminus S)} \big(\alpha^* st_{ij} - \alpha^* K - \delta_{0i} + \delta_{0j}\big)z_{ij} \\ =& \alpha^* \sum_{(i,j)\in E(S:\mathcal{I}\setminus S)} (st_{ij} - K)z_{ij} + \sum_{j\in\mathcal{I}\setminus S}\delta_{0j}\Big(\sum_{i\in S} z_{ij}\Big) - \sum_{i\in S}\delta_{0i}\Big(\sum_{j\in\mathcal{I}\setminus S} z_{ij}\Big)\end{aligned}$$

$$\begin{aligned}(iv) =& \sum_{(i,j)\in E(\mathcal{I}\setminus S:S)} \big(\alpha^* st_{ij} - \delta_{0i} + \delta_{0j}\big)z_{ij} \\ =& \alpha^* \sum_{(i,j)\in E(\mathcal{I}\setminus S:S)} st_{ij}z_{ij} + \sum_{j\in S}\delta_{0j}\Big(\sum_{i\in\mathcal{I}\setminus S} z_{ij}\Big) - \sum_{i\in\mathcal{I}\setminus S}\delta_{0i}\Big(\sum_{j\in S} z_{ij}\Big)\end{aligned}$$

$$(v) = \sum_{(i,j)\in E(\mathcal{I}\setminus S)} (-\delta_{0i} + \delta_{0j})z_{ij} = \sum_{i\in\mathcal{I}\setminus S}\delta_{0i}\Big(\sum_{j\in\mathcal{I}\setminus S} z_{ji} - \sum_{j\in\mathcal{I}\setminus S} z_{ij}\Big)$$

Therefore, $\sum_{i\in\mathcal{I}} \alpha_i x_i + \sum_{(i,j)\in\mathcal{A}_0} \delta_{ij}z_{ij} = \pi_0$ is reduced to

$$\alpha^*\Big(\sum_{i\in S} x_i + \sum_{(i,j)\in\delta(S)} st_{ij}z_{ij} + \sum_{(i,j)\in E(S)} st_{ij}z_{ij} - K\sum_{(i,j)\in\delta^+(S)} z_{ij}\Big) +$$

$$\pi_0\Big(\sum_{i\in\mathcal{I}} z_{i0} - 1\Big) + \sum_{i\in\mathcal{I}}\delta_{0i}\Big(\sum_{j\in\mathcal{I}_0} z_{ji} - \sum_{j\in\mathcal{I}_0} z_{ij}\Big) = 0.$$

Because the second and third terms are scalar multiplications of the equality set and the first term is that of the inequality (5), it defines a facet of $conv(\mathcal{X}_0)$. □

Proof of Proposition 3.4. Inequality (8) can be rewritten as

$$\sum_{i\in S} x_i \le \sum_{i\in S} u_i y_i - \sum_{(i,j)\in E(S)} [u_i + st_{ij} + u_j - K]^+ z_{ij},$$

where $[a]^+ = \max\{0, a\}$. To show that this inequality is valid for $\mathcal{X}$, let us given a feasible solution $(\bar{\mathbf{x}}, \bar{\mathbf{s}}^+, \bar{\mathbf{s}}^-, \bar{\mathbf{y}}, \bar{\mathbf{z}})$. Then, $\bar{\mathbf{z}}$ forms a cycle $C = \{(i,j) : \bar{z}_{ij} = 1,\ (i,j) \in \mathcal{A}_0\}$ which includes node 0. The set of nodes included in C is denoted as $V(C)$.

If $u_i + st_{ij} + u_j \le K$ for all $(i,j) \in E(S) \cap C$, then the inequality trivially holds because it is reduced to an aggregated upper bound constraint for $i \in S$. Therefore, let $R \subseteq E(S) \cap C$ such that $u_i + st_{ij} + u_j > K$ for $(i,j) \in R$, and let $R \neq \emptyset$. In addition, assume that R forms a path, that is, $R = \{(i_1, i_2), (i_2, i_3), \cdots, (i_{l-2}, i_{l-1}), (i_{l-1}, i_l)\}$ (If not, R can be partitioned into several mutually exclusive paths. Applying the following logic to each of them separately is straightforward and the proof still holds.). Then, from the following relations, it is demonstrated that the `U-STAR` inequality is valid:

$$\sum_{i\in S} u_i\bar{y}_i - \sum_{(i,j)\in E(S)} [u_i + st_{ij} + u_j - K]^+ \bar{z}_{ij} = \sum_{i\in S\cap V(C)} u_i - \sum_{(i,j)\in R} (u_i + st_{ij} + u_j - K)$$

$$= \sum_{i\in S\cap V(C)} u_i - \sum_{k=1}^{l-1}(u_{i_k} + u_{i_{k+1}} + st_{i_k i_{k+1}} - K) \ge (l-1)K - (u_{i_2} + \cdots u_{i_{l-1}}) - \sum_{k=1}^{l-1} st_{i_k i_{k+1}}$$

$$= K + (K - u_{i_2}) + \cdots + (K - u_{i_{l-1}}) - \sum_{k=1}^{l-1} st_{i_k i_{k+1}} \ge K - \sum_{k=1}^{l-1} st_{i_k i_{k+1}} \ge \sum_{i\in S} \bar{x}_i.$$

□

Proof of Proposition 3.5. For a given $S \subseteq \mathcal{I}$, consider the following points satisfying constraint (8) at equality:

- For each $i \in S$, $z_{0i} = y_i = z_{i0} = 1$ and $x_i = u_i$. ($|S|$ points)
- For each $i \in \mathcal{I} \setminus S$, $z_{0i} = y_i = z_{i0} = 1$ and $x_i = 0$. ($I - |S|$ points)
- For each $i \in \mathcal{I} \setminus S$, $z_{0i} = y_i = z_{i0} = 1$ and $x_i = u_i$. ($I - |S|$ points)

- For each $(i,j) \in E(\mathcal{I} \setminus S)$, $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$ and $x_i = x_j = 0$. $((I - |S|)(I - |S| - 1)$ points)
- For each $(i,j) \in E(S : \mathcal{I} \setminus S)$, $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$ and $x_i = u_i$. $(|S|(I - |S|)$ points)
- For each $(i,j) \in E(\mathcal{I} \setminus S : S)$, $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$ and $x_j = u_j$. $(|S|(I - |S|)$ points)
- For each $(i,j) \in E(S)$, let $z_{0i} = y_i = z_{ij} = y_j = z_{j0} = 1$. In this case, $x_i + x_j$ should be equal to $K - st_{ij}$. From the second condition of the proposition, $K - st_{ij} < u_i + u_j$. Therefore, one can let $(x_i, x_j) = (u_i, K - st_{ij} - u_i)$. $(|S|(|S| - 1)$ points)
- Choose one element from S, say i^*. Then, for each $j \in S \setminus \{i^*\}$, let $z_{0i^*} = y_{i^*} = z_{i^*j} = y_j = z_{j0} = 1$ and $(x_{i^*}, x_j) = (K - st_{i^*j} - u_j, u_j)$. $(|S| - 1$ points)

It is not hard to see that the above $I^2 + I - 1$ points are affinely independent. □

Appendix B. Pattern-based Formulation

We present a pattern-based formulation which is obtained by applying Dantzig-Wolfe decomposition for the original LSP-SQ. In particular, we apply a period-wise decomposition, that is, each pattern corresponds to a production plan of a single period. Let $\mathcal{P}_t$ be the set of possible production schedules in period t. For each pattern $p \in \mathcal{P}_t$, the following associated parameters are defined:

- $\bar{x}^p_{it}$: Production amount of item i in pattern $p \in \mathcal{P}_t$.
- $\bar{y}^p_{it} = 1$ if item i is produced in pattern $p \in \mathcal{P}_t$.
- $\bar{z}^p_{ijt} = 1$ if setup from item i to j occurs in pattern $p \in \mathcal{P}_t$.
- C_{tp}: Sum of the production and setup costs of pattern $p \in \mathcal{P}_t$, corresponding to $(\bar{\mathbf{x}}^p, \bar{\mathbf{y}}^p, \bar{\mathbf{z}}^p)$.

Let the pattern variable $\lambda_{tp} = 1$ if the pattern $p \in \mathcal{P}_t$ is selected in period t. The master problem `(MP)` is represented as follows:

$$\texttt{(MP)} \quad \text{minimize} \quad \sum_{t \in \mathcal{T}} \sum_{p \in \mathcal{P}_t} C_{tp} \lambda_{tp} + \sum_{i \in \mathcal{I}} \sum_{t \in \mathcal{T}} (hc_{it} s_{it} + bc_{it} b_{it}) \tag{B.1a}$$

$$\text{subject to} \quad s_{it-1} - b_{it-1} + \sum_{p \in \mathcal{P}_t} \bar{x}^p_{it} \lambda_{tp} = d_{it} + s_{it} - b_{it} \qquad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{B.1b}$$

$$\sum_{p \in \mathcal{P}_t} \bar{z}^p_{0it} \lambda_{tp} = \sum_{p \in \mathcal{P}_{t+1}} \bar{z}^p_{i0t+1} \lambda_{t+1p} \qquad \forall i \in \mathcal{I}, t \in \mathcal{T} \setminus \{T\} \tag{B.1c}$$

$$\sum_{p \in \mathcal{P}_t} \lambda_{tp} = 1 \qquad \forall t \in \mathcal{T} \tag{B.1d}$$

$$s_{it},\ b_{it} \geq 0 \qquad \forall i \in \mathcal{I},\ t \in \mathcal{T} \tag{B.1e}$$

$$\lambda_{tp} \in \{0,1\} \qquad \forall t \in \mathcal{T}, p \in \mathcal{P}_t \tag{B.1f}$$

Demand constraints (1b) and setup carryover constraints (1e) of the original LSP-SQ (1) are kept in (MP) as linking constraints (B.1b)–(B.1c). Other constraints are presented in subproblems. It is known that, as the pattern incorporates all the decisions regarding a single period, the LP relaxation bound of (MP) is equivalent to that which is obtained when the convex hull of the single-period solution set $\mathcal{X}$ is known (Wolsey, 2020).

The LP relaxation of (MP) is solved by a column generation procedure which recursively adds profitable columns by solving pricing subproblems until no one is found (Desaulniers et al., 2006). Let μ_{it}, π_{it}, and σ_t be the dual variables of constraints (B.1b)–(B.1d), respectively. Then, the pricing subproblem for period t which tries to find profitable patterns by minimizing the reduced cost can be constructed as follows:

$$(\texttt{SP}_t) \quad \min \quad \sum_{i \in \mathcal{I}} \Big((pc_{it} - \mu_{it}) x_{it} - \pi_{it} z_{0it} + \pi_{it-1} z_{i0t} \Big) + \sum_{(i,j) \in \mathcal{A}} sc_{ijt} z_{ijt} \tag{B.2a}$$

$$\text{s.t.} \quad \sum_{i \in \mathcal{I}} x_{it} + \sum_{(i,j) \in \mathcal{A}} st_{ijt} z_{ijt} \leq K_t \tag{B.2b}$$

$$x_{it} \leq u_{it} y_{it} \qquad \forall i \in \mathcal{I} \tag{B.2c}$$

$$\sum_{i \in \mathcal{I}} z_{0it} = 1 \tag{B.2d}$$

$$\sum_{j \in \mathcal{I}_0} z_{jit} = \sum_{j \in \mathcal{I}_0} z_{ijt} = y_{it} \qquad \forall i \in \mathcal{I} \tag{B.2e}$$

$$\sum_{i \in \mathcal{I}} f_{0it} = \sum_{i \in \mathcal{I}} y_{it} \tag{B.2f}$$

$$\sum_{j \in \mathcal{I}_0 \setminus \{i\}} f_{jit} - \sum_{j \in \mathcal{I}_0 \setminus \{i\}} f_{ijt} = y_{it} \qquad \forall i \in \mathcal{I} \tag{B.2g}$$

$$f_{ijt} \leq I z_{ijt} \qquad \forall (i,j) \in \mathcal{A}_0 \tag{B.2h}$$

$$x_{it} \geq 0,\ y_{it} \in \{0,1\} \qquad \forall i \in \mathcal{I} \tag{B.2i}$$

$$f_{ijt} \geq 0,\ z_{ijt} \in \{0,1\} \qquad \forall (i,j) \in \mathcal{A}_0 \tag{B.2j}$$

We use single-commodity flow formulation (B.2f)–(B.2h) to ensure the validity of cycles. If the optimal objective value of ($\texttt{SP}_t$) is smaller than σ_t, that is, if a pattern with the negative reduced cost is found, the pattern corresponding to the optimal solution of ($\texttt{SP}_t$) is added to (MP). This procedure is repeated until no pattern is generated by ($\texttt{SP}_t$) for all $t \in \mathcal{T}$.