Noname manuscript No.
(will be inserted by the editor)

A Computational Analysis of Lower Bounds for Big Bucket Production Planning Problems

Kerem Akartunalı · Andrew J. Miller

the date of receipt and acceptance should be inserted later

Abstract In this paper, we analyze a variety of approaches to obtain lower bounds for multi-level production planning problems with big bucket capacities, i.e., problems in which multiple items compete for the same resources. We give an extensive survey of both known and new methods, and also establish relationships between some of these methods that, to our knowledge, have not been presented before. As will be highlighted, understanding the substructures of difficult problems provide crucial insights on why these problems are hard to solve, and this is addressed by a thorough analysis in the paper. We conclude with computational results on a variety of widely used test sets, and a discussion of future research.

Keywords Production Planning · Lot-Sizing · Integer Programming · Strong Formulations · Lagrangian Relaxation

Mathematics Subject Classification (2000) 90C11

1 Introduction

Production planning problems have drawn considerable interest from both researchers and practitioners since the seminal paper of Wagner and Whitin [48]. These problems search for the production plan with the minimum total cost (fixed charges such as setup costs and linear charges such as inventory holding

The research carried out was supported in part by the National Science Foundation grant No. DMI 0323299.

Dept. of Management Science, University of Strathclyde
40 George Street, Glasgow G1 1QE, United Kingdom
E-mail: kerem.akartunali@strath.ac.uk

IMB, Université de Bordeaux 1 & RealOpt, INRIA Bordeaux Sud-Ouest
351 Cours de la Libération, 33405 Talence, France
E-mail: andrew.miller@math.u-bordeaux1.fr

costs) that satisfies customer demand and follows restrictions of the production environment such as those imposed by capacities. The focus of this paper is on multi-level, multi-item production planning problems with "big bucket" capacities, i.e., each resource is shared by multiple items and different items can be produced in a specific time period. These real-world problems are complicated and computationally challenging to solve, often having complicated BOM (Bills of Materials) structures, where the BOM details which items are required to produce each item. The BOM often has multiple *levels*, where the last level can be thought of as final products, the previous level can be thought of as components required to make final products, and so forth.

Let NT, NI and NK be the number of periods, items, and machine types, respectively. We assume that each machine type operates only on one level, and each level can employ a number of machine types. Note that if a component appears in two or more levels, then it is assumed to be a different item in each different level. The set *endp* indicates all end-items, i.e. items with external demand; the other items are assumed to have only internal demand. (No generality lost, since any item that has both internal and external demand can be modeled as two distinct items that share a setup variable.) Let x_t^i, y_t^i, and s_t^i represent production, setup, and inventory variables for item i in period t, respectively. The setup and inventory cost coefficients are indicated by f_t^i and h_t^i for each period t and item i. Note that production costs might be also included in the problem in a similar fashion to inventory holding costs. The parameter $\delta(i)$ represents the set of immediate successors of item i, and the parameter r^{ij} represents the number of items required of i to produce one unit of j. Note that r^{ij} is defined not only for immediate dependencies, but for *all* dependencies between items i and j. The parameter d_t^i is the demand for end-product i in period t, and $d_{t,t'}^i$ is the total demand between t and t', i.e., $d_{t,t'}^i = \sum_{\bar{t}=t}^{t'} d_{\bar{t}}$. The parameter a_k^i represents the time necessary to produce one unit of i on machine k, and ST_k^i is the setup time for item i on machine k, which has a capacity of C_t^k in period t. Note that each item is processed by a preassigned machine, and we assume that each item is assigned only to one machine (hence, for an item i' that is not processed on a machine k', $a_{k'}^{i'} = 0$ and $ST_{k'}^{i'} = 0$). Let M_t^i be a big number. Then the formulation of the basic model follows:

$$
\begin{aligned}
\min \; & \sum_{t=1}^{NT}\sum_{i=1}^{NI} f_t^i y_t^i + \sum_{t=1}^{NT}\sum_{i=1}^{NI} h_t^i s_t^i && && (1)\\
\text{s.t. } & x_t^i + s_{t-1}^i - s_t^i = d_t^i && t \in [1,NT], i \in endp && (2)\\
& x_t^i + s_{t-1}^i - s_t^i = \sum_{j\in\delta(i)} r^{ij} x_t^j && t \in [1,NT], i \in [1,NI]\backslash endp && (3)\\
& \sum_{i=1}^{NI}(a_k^i x_t^i + ST_k^i y_t^i) \le C_t^k && t \in [1,NT], k \in [1,NK] && (4)\\
& x_t^i \le M_t^i y_t^i && t \in [1,NT], i \in [1,NI] && (5)\\
& y \in \{0,1\}^{NT\times NI} && && (6)\\
& x \ge 0 && && (7)\\
& s \ge 0 && && (8)
\end{aligned}
$$

The constraints (2) and (3) ensure production balance and demand satisfaction for end-items and intermediate items respectively. Note that for the simplicity of the formulation, we assume lead times to be zero (this does not lose generality; if lead times Δ^i for an item i exist, then this can be be introduced in these constraints simply by replacing x_t^i variables with $x_{t-\Delta^i}^i$.). W.l.o.g., we also assume the initial inventories to be zero. The constraints (4) are the big bucket capacity constraints, (5) ensure that the setup variable is set to be 1 if there is positive production, and finally (6), (7), and (8) provide the integrality and nonnegativity requirements. Note that we can define M_t^i as follows, where $k \in [1,NK]$ such that $a_k^i \neq 0$:

$$
\begin{aligned}
M_t^i &= \min\Big(d_{t,NT}^i, \frac{C_t^k - ST_k^i}{a_k^i}\Big) && i \in endp\\
M_t^i &= \min\Big(\sum_{j\in endp} r^{ij} d_{t,NT}^j, \frac{C_t^k - ST_k^i}{a_k^i}\Big) && i \in [1,NI]\backslash endp
\end{aligned}
$$

We next define an echelon reformulation of the problem, see e.g. Pochet and Wolsey [38]. Our motivation for defining this reformulation is that it clearly exhibits the single-item structure that is present for each item, and it therefore enables us to apply results for single-item models to the multi-level model. We first define echelon demand parameters D_t^i and echelon stock variables E_t^i:

$$
\begin{aligned}
D_t^i &= d_t^i + \sum_{j\in\delta(i)} r^{ij} D_t^j && t \in [1,NT], i \in [1,NI] && (9)\\
E_t^i &= s_t^i + \sum_{j\in\delta(i)} r^{ij} E_t^j && t \in [1,NT], i \in [1,NI] && (10)
\end{aligned}
$$

Note that for (9) to be well-defined, we let $d_t^i = 0$ for all $i \in [1,NI]\backslash endp$. Substituting (10) into (2) and (3) for s_t^i, and using the definition (9), we obtain an equation that can replace (2) and (3) in the original formulation:

$$x_t^i + E_{t-1}^i - E_t^i = D_t^i \qquad t \in [1, NT], i \in [1, NI] \tag{11}$$

To satisfy (8), we add the following constraints:

$$E_t^i \geq \sum_{j \in \delta(i)} r^{ij} E_t^j \qquad t \in [1, NT], i \in [1, NI] \tag{12}$$

$$E \geq 0 \tag{13}$$

Finally, to eliminate the inventory variable s, we define echelon inventory holding cost $H_t^j = h_t^j - \sum_{i=1}^{NI} r^{ij} h_t^i$ and replace the objective function (1) with

$$\sum_{t=1}^{NT} \sum_{i=1}^{NI} f_t^i y_t^i + \sum_{t=1}^{NT} \sum_{i=1}^{NI} H_t^i E_t^i \tag{14}$$

We can therefore define the feasible region of the production planning problem as $X = \{(x, y, E) | (4) - (7), (11) - (13)\}$, which will be referred in the remainder of the paper as the "basic formulation". The production planning problem can be defined as $Z = \min\{(14) | (x, y, E) \in X\}$. We could easily include overtime (i.e., extra capacity that can be bought with an additional cost) or backlogging (i.e., satisfying demand later than requested by the customer with a cost for customer dissatisfaction) variables to generalize this basic model, and some of the test problems we consider in Section 4 incorporate them.

For simplicity, we will sometimes use $conv(a)$ to denote $conv((x, y, E)|(a))$, where (a) is a set of constraints. For example, $\{(x, y)|(7) \cap conv((6))\}$ represents $\{(x, y)|(7)\} \cap conv(\{(x, y)|(6)\})$ in our notation.

1.1 Literature Review

Even the capacitated version of the single-item production planning problem is $\mathcal{NP}$-hard (Florian et al. [19] and Bitran and Yanasse [11]) and therefore dynamic programming algorithms are only limited to some special cases, see e.g. Zangwill [51], Florian and Klein [18], Federgruen and Tzur [16]. Therefore, heuristic algorithms have been employed by many researchers with the hope of obtaining good solutions in acceptable computational times. Heuristic frameworks in general use some decomposition ideas, such as Lagrangian-based decomposition (e.g. Trigeiro et al. [46], Tempelmeier and Derstroff [44]), forward scheme and relax-and-fix (e.g. Belvaux and Wolsey [7], Stadtler [43], Federgruen et al. [17], Akartunalı and Miller [1]) and coefficient modification (e.g. Katok et al. [22], Van Vyve and Pochet [47]). The main disadvantages of the heuristic algorithms (unless based on exact methods such as Lagrangian relaxation) are the lack of solution quality guarantee and the lack of useful insights about basic problem structures.

Mathematical programming results on production planning problems have usually focused on special cases such as single-item problems, and they have been limited for problems with big bucket capacities. We will briefly discuss these techniques in two subgroups: 1) Valid inequalities that are added into the original formulation using separation algorithms, and 2) Extended reformulations that solve the problem in a different variable space.

An early polyhedral study that defines problem-specific valid inequalities for production planning problems is the study of Barany et al. [5], which describes fully the convex hull of the single-item uncapacitated problem. Some special cases of single-item problems are investigated in Küçükyavuz and Pochet [24] (uncapacitated, backlogging), Pochet and Wolsey [37] (constant capacities), Loparic et al. [26] (uncapacitated, sales and safety stocks), and Constantino [13] (uncapacitated, start-up costs). Atamtürk and Muñoz [4] provide a recent polyhedral study that investigates the bottleneck cover structure in capacitated single-item problems, and Pochet and Wolsey [36] extend some single-item results to the multi-level case. On the other hand, Miller et al. [32,33] provide rare results on multi-item problems with big bucket capacities, where the authors study single-period relaxations and propose valid inequalities. In a recent study, Levi et al. [25] study a version of the capacitated multi-item problem and they propose an approximation algorithm based on generating flow cover inequalities and randomized rounding.

A compact extended reformulation for production planning is the facility location reformulation of Krarup and Bilde [23], which defines the convex hull of the uncapacitated single-item problem when projected to original variable space. Eppen and Martin [15] study the shortest path reformulation, which is of smaller size compared to facility location reformulation. Rardin and Wolsey [39] investigate the multi-commodity reformulation for fixed-charge network problems. Belvaux and Wolsey [8] and Wolsey [50] are recent studies about reformulations and modeling issues. Anily et al. [3] provide tight reformulations for some special cases of the multi-item problem with joint setups.

Finally, we note that mathematical programming results on production planning problems are not only limited to these two approaches. Lagrangian relaxation has been used by Billington et al. [9] in a Branch&Bound scheme, as well as in the heuristic approach of Thizy and Van Wassenhove [45]. On the other hand, Dantzig-Wolfe decomposition has been in use since the paper of Manne [28], with advancements of Bitran and Matsuo [10] and very recently of Degraeve and Jans [14]. We refer the interested reader to Buschkühl et al. [12] for a thorough and very recent review.

1.2 Motivation and Organization of the Paper

In spite of this research, big bucket production planning problems remain hard to solve. Part of the reason for this is that most previous research focuses on developing and using results for single-item models, which are not sufficient to capture the fundamental sources of complexity of big bucket problems. The primary goals of this paper are to evaluate the strength of the relaxations defined by different mathematical programming techniques and to investigate why big bucket production planning problems are hard to solve in practice. More specifically, we are not primarily interested in extending single-item results to general production planning problems, but we want to discover relationships between different methods for generating lower bounds and the fundamental substructures that often make these methods insufficient to solve these problems well. We will consider all known methods for generating lower bounds of which we are aware, and we will investigate previously untried methods as well.

In Section 2, we provide a comprehensive survey of lower bounding methods presented in previous research, and we discuss previously untested methods as well. Section 3 is devoted to theoretical comparisons of different techniques, which can provide structural insight into multi-level big bucket problems. In Section 4, we present extensive computational comparisons obtained using widely used data sets. We conclude with future directions in Section 5.

2 Valid Inequalities, Reformulations, and Relaxations

In this section we discuss different approaches to obtain lower bounds. These methods vary from defining valid inequalities and reformulations to the use of Lagrangian relaxation.

2.1 Valid Inequalities

The first technique we consider is the use of (ℓ, S) inequalities of Barany et al. [5] defined for single-item problems, and generalized by Pochet and Wolsey [36] to multi-level problems using the echelon reformulation. These can be defined as follows:

$$\sum_{t \in S} x_t^i \leq \sum_{t \in S} D_{t,\ell}^i y_t^i + E_\ell^i \qquad \ell \in [1, NT], i \in [1, NI], S \subseteq [1, \ell] \tag{15}$$

Since these inequalities are valid for the single-item submodels defined by each item, they are valid for the multi-item problem as well. Although there is an exponential number of these inequalities, a simple polynomial separation algorithm exists as shown in Barany et al. [6], see Algorithm 1. As will be discussed later, there exist stronger formulations for the multi-level problem than that provided by using the (ℓ, S) inequalities alone, but (ℓ, S) inequalities have good practical use, especially when considering large problems.

The feasible region associated with this formulation can be defined as $X_{LS} = \{(x, y, E) | (4) - (7), (11) - (13), (15)\}$, and the problem can be defined as $Z_{LS} = \min\{(14) | (x, y, E) \in X_{LS}\}$.

Algorithm 1: (ℓ, S) separation

Input: LP relaxation solution (x^*, y^*, E^*)
Output: Violated (ℓ, S) inequalities
for i=1 **to** NI
  for $\ell = 1$ **to** NT
    Initialize $S \leftarrow \{\,\}$
    for t=1 **to** ℓ
      if ${x^*}^i_t > D^i_{t,\ell} {y^*}^i_t$
        $S \leftarrow S \cup \{t\}$
    if $\sum_{t \in S} {x^*}^i_t > \sum_{t \in S} D^i_{t,\ell} {y^*}^i_t + {E^*}^i_\ell$
      Add the violated (ℓ, S) inequality

2.2 Reformulations

The next technique we consider is the facility location reformulation, originally defined by Krarup and Bilde [23] for the single-item problem. This reformulation divides production according to which period it is intended for. This requires first defining new variables $u^i_{t,t'}$, which indicate the production of item i in period t to satisfy the demand of period t', where $t' \geq t$. The following constraints should be added into the basic formulation to finalize the reformulation:

$$u^i_{t,t'} \leq D^i_{t'} y^i_t \qquad t \in [1, NT], t' \in [t, NT], i \in [1, NI] \tag{16}$$

$$\sum_{t=1}^{t'} u^i_{t,t'} = D^i_{t'} \qquad t' \in [1, NT], i \in [1, NI] \tag{17}$$

$$x^i_{t'} = \sum_{t=t'}^{NT} u^i_{t',t} \qquad t' \in [1, NT], i \in [1, NI] \tag{18}$$

$$u \geq 0 \tag{19}$$

This formulation adds $O(NT^2 NI)$ variables and $O(NT^2 NI)$ constraints to the problem. One advantage of using the new variables $u^i_{t,t'}$ is that we can rewrite the capacity constraint (4) as follows:

$$\sum_{i=1}^{NI} (a^i_k (\sum_{t'=t}^{NT} u^i_{t,t'}) + ST^i_k y^i_t) \leq C^k_t \qquad t \in [1, NT], k \in [1, NK] \tag{20}$$

This, along with constraints (16), can considerably help a state-of-the-art MIP solver generate knapsack cover cuts. Specifically, note that by adding $\sum_{i=1}^{NI} a^i_k D^i_{t,NT} y^i_t$ on both sides and after rearranging the terms, (20) can be rewritten as

$$\sum_{i=1}^{NI} (a^i_k D^i_{t,NT} + ST^i_k) y^i_t \leq C^k_t + \left(\sum_{i=1}^{NI} \sum_{t'=t}^{NT} a^i_k (D^i_{t'} y^i_t - u^i_{t,t'}) \right) \tag{21}$$

For each fixed pair of (t,k), and for any subsets $\mathcal{I} \subseteq \{1,...,NI\}$ and $\mathcal{T} \subseteq \{t,...,NT\}$, we may generate cover cuts for each of the following continuous 0-1 knapsack constraints (which is obtained in the same fashion as (21), but only for the subsets $\mathcal{I}$ and $\mathcal{T}$ and the "continuous variable" is highlighted in parenthesis on the right hand side):

$$\sum_{i\in\mathcal{I}}(a_k^i(\sum_{t'\in\mathcal{T}} D_{t'}^i) + ST_k^i)y_t^i \leq C_t^k + \left(\sum_{i\in\mathcal{I}}\sum_{t'\in\mathcal{T}} a_k^i(D_{t'}^i y_t^i - u_{t,t'}^i)\right) \tag{22}$$

Note that because of (16), the expression in the parenthesis on the right-hand side of (21) or (22) can be considered as a single nonnegative continuous variable. Binary knapsack constraints with a single nonnegative continuous variable were studied by Marchand and Wolsey [29,30] (see also Richard et al. [40,41]). Commercial solvers use the kinds of results they present to efficiently find subsets $\mathcal{I}$ and $\mathcal{T}$ and generate cover cuts that will approximate $conv(X_{KN}^{(t,k)})$, where $X_{KN}^{(t,k)} = \{(y,u)|(6),(16),(19),(20)\}$ is the feasible region of the intersection of these continuous 0-1 knapsack problems for a fixed (t,k) pair. Note that we can also define it as $X_{KN}^{(t,k)} = proj_{y,u}\bar{X}_{KN}^{(t,k)}$ with $\bar{X}_{KN}^{(t,k)} = \{(x,y,E,u)|(6),(16),(19),(20),(18),(11)\}$, just for the convenience of having it in higher dimension. Related to $\bar{X}_{KN}^{(t,k)}$, we will define $\bar{X}_{KN}^{(t,k,\{t(i)\})}$, for which we first choose a $t(i) \in [t,NT]$ for all $i \in [1,NI]$, for a given t. Then, we define

$$u_{t,t_1}^i \leq D_{t_1}^i y_t^i \qquad t_1 \in [t,NT], i \in [1,NI] \tag{23}$$

$$u_{t_1,t_2}^i \leq D_{t_2}^i y_{t_1}^i \qquad t_1 \in [t+1,t(i)], t_2 \in [t_1,t(i)], \; i \in [1,NI] \tag{24}$$

$$x_t^i = \sum_{t_1=t}^{NT} u_{t,t_1}^i \qquad i \in [1,NI] \tag{25}$$

$$E_{t-1}^i = \sum_{t_1=1}^{t-1}\sum_{t_2=t}^{NT} u_{t_1,t_2}^i \qquad i \in [1,NI] \tag{26}$$

$$x_t^i + E_{t-1}^i + \sum_{t_1=t+1}^{t(i)}\sum_{t_2=t_1}^{t(i)} u_{t_1,t_2}^i \geq D_{t,t(i)}^i \qquad i \in [1,NI] \tag{27}$$

Then, $\bar{X}_{KN}^{(t,k,\{t(i)\})} = \{(x,y,E,u)|(6),(19),(20),(23)-(27)\}$. Note that we will use this explicit definition for the purposes of proving a key proposition in the next section.

On a separate note, basic continuous cover inequalities can also be generated as MIR inequalities, which are known to be effective for general mixed integer programs (see e.g. Günlük and Pochet [20]). Of course, our approach will increase the problem size and it might easily become so large that it cannot be solved to optimality in an acceptable time. However, using this approach for the purpose of generating lower bounds can yield insights into the structure of our problems. This idea was initially suggested for single-level, single-machine problems by Van Vyve [1]. To the best of our knowledge, this approach has not been tested for multi-level problems before.

The feasible region associated with the facility location reformulation can be defined as $X_{FL} = \{(x, y, E, u)|(5) - (7), (11) - (13), (16) - (20)\}$, and the associated problem as $Z_{FL} = \min\{(14)|(x, y, E, u) \in X_{FL}\}$. On the other hand, generating all cover cuts approximates $\bigcap_{t=1}^{NT} \bigcap_{k=1}^{NK} conv(X_{KN}^{(t,k)})$, which is an approximation for $conv(\bigcap_{t=1}^{NT} \bigcap_{k=1}^{NK} X_{KN}^{(t,k)})$. This leads us to define the polyhedron:

$$X_{FL}^{KN} = \{(x, y, E, u)|(5), (7), (11) - (13), (17), (18)\} \cap conv(\bigcap_{t=1}^{NT} \bigcap_{k=1}^{NK} X_{KN}^{(t,k)})$$

and the associated problem $Z_{FL}^{KN} = \min\{(14)|(x, y, E, u) \in X_{FL}^{KN}\}$.

Next, we discuss the single-period relaxation of Miller et al. [32,33], called as *PI* (*Preceding Inventory*). To describe the single-period formulation, for a given machine $k \in [1, NK]$ and a given time period $t \in [1, NT]$, we choose a time period $t(i) \geq t$ for each $i \in [1, NI]$. Then we define

$$S^i = E_{t-1}^i + \sum_{\hat{t}=t+1}^{t(i)} D_{\hat{t},t(i)}^i y_{\hat{t}}^i \qquad i \in [1, NI]$$

$$D^i = D_{t,t(i)}^i \qquad i \in [1, NI]$$

Then, the single-period formulation can be written as follows:

$$x_t^i + S^i \geq D^i \qquad i \in [1, NI] \qquad (28)$$

$$x_t^i \leq M_t^i y_t^i \qquad i \in [1, NI] \qquad (29)$$

$$\sum_{i=1}^{NI} (a_k^i x_t^i + ST_k^i y_t^i) \leq C_t^k \qquad (30)$$

$$x_t^i, S^i \geq 0 \qquad i \in [1, NI] \qquad (31)$$

$$y_t^i \in \{0, 1\} \qquad i \in [1, NI] \qquad (32)$$

We can define $X_{PI}^{(t,k,\{t(i)\})} = \{(x, y, S)|(28) - (32)\}$ as the feasible region associated with a set of $t(i)$ values, and $X_{PI}^{(t,k)} = \bigcap_{\{t(i)\}} X_{PI}^{(t,k,\{t(i)\})}$ represents the feasible region for a given (t, k) pair. Note the similarity between this feasible region and $X_{KN}^{(t,k)}$ we discussed earlier. Miller et al. [32,33] define valid inequalities (namely cover and reverse cover inequalities) for *PI*, which are naturally valid for the original problem as well, and these inequalities can be seen as an approximation for $conv(X_{PI}^{(t,k)})$, which is of interest in our context as providing a lower bound for the original problem when all single-period relaxations are considered for a problem.

[1] Personal communication.

Next, we define the shortest path reformulation of Eppen and Martin [15]. In this formulation, which was originally defined for single-item uncapacitated models, the variables $z^i_{t,t'}$ are 1 if production of i in period t satisfies all the demand for i in periods $t, ..., t'$ but not beyond t', and 0 otherwise. Note, as a result of the constraints (5) and zero initial inventories, the relationship between the new and original variables is as follows:

$$x^i_t = \sum_{t'=t}^{NT} D^i_{t,t'} z^i_{t,t'} \qquad t \in [1, NT], i \in [1, NI] \tag{33}$$

For the multi-level capacitated problem, we do not have the same optimality properties that we have for the single-item problem; we therefore let the z variables take fractional values as they represent "the fraction of demand in periods $t, ..., t'$ satisfied by production in period t". Also, using the echelon inventory holding costs H^i_t, we define total inventory costs $c^i_{t,t'} = D^i_{t,t'} \sum_{j=t}^{NT} H^i_j$. The formulation is then as follows:

$$\min \sum_{t=1}^{NT} \sum_{i=1}^{NI} f^i_t y^i_t + \sum_{t=1}^{NT} \sum_{t'=t}^{NT} \sum_{i=1}^{NI} c^i_{t,t'} z^i_{t,t'} \tag{34}$$

$$\text{s.t. } 1 = \sum_{t=1}^{NT} z^i_{1,t} \qquad i \in [1, NI] \tag{35}$$

$$\sum_{t=1}^{t'-1} z^i_{t,t'-1} = \sum_{t=t'}^{NT} z^i_{t',t} \qquad t' \in [2, NT], i \in [1, NI] \tag{36}$$

$$\sum_{t'=t}^{NT} z^i_{t,t'} \leq y^i_t \qquad t \in [1, NT], i \in [1, NI] \tag{37}$$

$$\sum_{i=1}^{NI} (ST^i_k y^i_t + a^i_k \sum_{t'=t}^{NT} D^i_{t,t'} z^i_{t,t'}) \leq C^k_t \qquad t \in [1, NT], k \in [1, NK] \tag{38}$$

$$\sum_{t=1}^{t'} \sum_{\hat{t}=t}^{NT} (D^i_{t,\hat{t}} z^i_{t,\hat{t}} - \sum_{j \in \delta(i)} r^{ij} D^j_{t,\hat{t}} z^j_{t,\hat{t}}) \geq d^i_{1,t'} \qquad t' \in [1, NT], i \in [1, NI] \tag{39}$$

$$z \geq 0 \tag{40}$$

$$y \in \{0, 1\}^{NTxNI} \tag{41}$$

The constraints (35) and (36) are the flow balance constraints, (37) provide the relationship between the linear and binary variables, (38) is the capacity constraint, (39) ensures the relationship between different levels, and finally (40) and (41) provide the nonnegativity and integrality constraints. Note that for our multi-level problem, we derive the constraint (39) as follows: Using (11) and (12), and the assumption of zero initial inventory, we obtain

$$\sum_{t=1}^{t'} (x^i_t - D^i_t) \geq \sum_{t=1}^{t'} \sum_{j \in \delta(i)} r^{ij} (x^j_t - D^j_t) \tag{42}$$

Substituting (33) into (42) and rewriting results in (39). Note that this formulation adds as many variables as the facility location reformulation, but number of constraints is only $O(NT \times NI)$. However, this formulation is not necessarily easier to solve, in part because the new constraints are comparatively dense and the coefficients on the new variables comparatively large.

The feasible region associated with this formulation can be defined as $X_{SP} = \{(y,z)|(35)-(41)\}$, and the problem can be defined as $Z_{SP} = \min\{(34)|(y,z) \in X_{SP}\}$. Part of our motivation for completely substituting the x and E variables out of the formulation is that relaxing the constraints (35), (36), and (39) decomposes the problem into NT distinct subproblems, one for each time period (an analogous observation was first made for single-level problems by Jans and Degraeve [21]). We will discuss this property in more detail later.

Next, we consider the multi-commodity reformulation proposed by Rardin and Wolsey [39]. This approach is originally described for fixed-charge network flow problems. Like the facility location reformulation, it divides production using destination information, but since we have multiple levels, it also includes information about which end-item in the BOM it is produced for. Stock variables are also divided in a similar fashion. Thus, the new variables $w^{i,j}_{t,t'}$ indicate production of item i in period t to satisfy the demand of end-item j in period t', $t' \geq t$, and the new variables $v^{i,j}_{t,t'}$ indicate the inventory of item i held over at the end of period t to satisfy demand of end-item j in period t', $t' > t$. The following constraints should be added to the basic formulation to finalize the reformulation:

$$x^i_{t'} = \sum_{t=t'}^{NT} \sum_{j \in endp} w^{i,j}_{t',t} \qquad t' \in [1,NT], i \in [1,NI] \tag{43}$$

$$w^{i,j}_{t,t'} \leq r^{ij} d^j_{t'} y^i_t \qquad t \in [1,NT], t' \in [t,NT], \; i \in [1,NI], j \in endp \tag{44}$$

$$v^{i,i}_{t-1,t} + w^{i,i}_{t,t} = d^i_t \qquad t \in [1,NT], i \in endp \tag{45}$$

$$v^{i,i}_{t-1,t'} + w^{i,i}_{t,t'} = v^{i,i}_{t,t'} \qquad t \in [1,NT-1], t' \in [t+1,NT], \; i \in endp \tag{46}$$

$$v^{i,q}_{t-1,t} + w^{i,q}_{t,t} = \sum_{j \in \delta(i)} r^{ij} w^{j,q}_{t,t} \qquad t \in [1,NT], i \in [1,NI] \backslash endp, \; q \in endp \tag{47}$$

$$v^{i,q}_{t-1,t'} + w^{i,q}_{t,t'} = v^{i,q}_{t,t'} + \sum_{j \in \delta(i)} r^{ij} w^{j,q}_{t,t'} \qquad t \in [1,NT-1], t' \in [t+1,NT], \; i \in [1,NI] \backslash endp, q \in endp \tag{48}$$

$$w, v \geq 0 \tag{49}$$

The constraints (43) indicate the relation between the new and old variables, (44) provide the relationship between the linear and binary variables,

(45) and (46) are demand flow balance constraints for end items, (47) and (48) are demand flow balance constraints for non-end items, and finally (49) provide the nonnegativity constraints. This reformulation introduces $O(NT^2NI^2)$ additional variables and $O(NT^2NI^2)$ additional constraints. This is the main disadvantage of this reformulation, which can easily become computationally intractable as the problem size grows. However, it is the tightest compact, i.e., polynomial size, reformulation that we know for the problems in question.

The feasible region associated with this formulation can be defined as $X_{MC} = \{(x, y, E, w, v)|(4) - (7), (11) - (13), (43) - (49)\}$, and the problem can be defined as $Z_{MC} = \min\{(14)|\ (x, y, E, w, v) \in X_{MC}\}$.

2.3 Relaxations

Next, we discuss three approaches that employ Lagrangian relaxation to obtain structured subproblems and from those lower bounds for the original problem. The first approach is to relax the capacity constraints (4), and obtain

$$
\begin{aligned}
LR_1(\lambda) = \min & \sum_{t=1}^{NT}\sum_{i=1}^{NI} f_t^i y_t^i + \sum_{t=1}^{NT}\sum_{i=1}^{NI} H^i E_t^i \\
& - \sum_{t=1}^{NT}\sum_{k=1}^{NK} \lambda_t^k \left(C_t^k - (\sum_{i=1}^{NI} a_k^i x_t^i + ST_k^i y_t^i) \right) \qquad (50) \\
& \text{subject to } (x, y, E) \in X_{LR1}
\end{aligned}
$$

where $X_{LR1} = \{(x, y, E)|(5) - (7), (11) - (13)\}$. Thus, the Lagrangian subproblem is a multi-item, multi-level uncapacitated production planning problem. The Lagrangian dual problem is

$$LD_1 = \max_{\lambda \geq 0} LR_1(\lambda) \qquad (51)$$

The next Lagrangian relaxation approach relaxes the constraints linking separate levels, i.e. constraints (12), to obtain

$$
\begin{aligned}
LR_2(\mu) = \min & \sum_{t=1}^{NT}\sum_{i=1}^{NI} f_t^i y_t^i + \sum_{t=1}^{NT}\sum_{i=1}^{NI} H^i E_t^i \\
& - \sum_{t=1}^{NT}\sum_{i=1}^{NI} \mu_t^i \left(E_t^i - \sum_{j \in \delta(i)} r^{ij} E_t^j \right) \qquad (52) \\
& \text{subject to } (x, y, E) \in X_{LR2}
\end{aligned}
$$

where $X_{LR2} = \{(x, y, E)|\ (4) - (7), (11), (13)\}$. The Lagrangian subproblem therefore decomposes into NK disjoint multi-item, big bucket single-machine problems, one for each machine. The Lagrangian dual problem becomes

$$LD_2 = \max_{\mu \geq 0} LR_2(\mu) \qquad (53)$$

Finally, the last Lagrangian approach extends the work of Jans and Degraeve [21] for single-level problems, which itself uses the shortest path reformulation of Eppen Martin [15]. Jans and Degraeve [21] simply relaxed the constraints linking time periods, yielding disjoint single-period subproblems. However, the problem in the multi-level case is that the constraints linking levels also involve multiple periods. Therefore, decomposing the problem into disjoint subproblems for each period is not possible, unless all constraints linking levels are also dualized. We dualize the constraints (35), (36) and (39) in the shortest path reformulation to obtain

$$\begin{aligned} LR_3(\beta,\gamma) =\min\ & \sum_{t=1}^{NT}\sum_{i=1}^{NI} f_t^i y_t^i + \sum_{t=1}^{NT}\sum_{t'=t}^{NT}\sum_{i=1}^{NI} c_{t,t'}^i z_{t,t'}^i - \sum_{i=1}^{NT}\beta_1^i\left(1-\sum_{t=1}^{NT} z_{1,t}^i\right) \\ & -\sum_{i=1}^{NI}\sum_{t'=2}^{NT}\beta_{t'}^i\left(\sum_{t=1}^{t'-1} z_{t,t'-1}^i - \sum_{t=t'}^{NT} z_{t',t}^i\right) \\ & -\sum_{i=1}^{NI}\sum_{t'=1}^{NT}\gamma_{t'}^i\left(\sum_{t=1}^{t'}\sum_{\hat{t}=t}^{NT}(D_{t,\hat{t}}^i z_{t,\hat{t}}^i - \sum_{j\in\delta(i)} r^{ij} D_{t,\hat{t}}^j z_{t,\hat{t}}^j) - d_{1,t'}^i\right) \\ & \text{subject to } (y,z)\in X_{LR3} \end{aligned} \tag{54}$$

where $X_{LR3} = \{(y,z)|\ (37),(38),(40),(41)\}$. It is easy to note that the Lagrangian subproblem decomposes into $NK \times NT$ disjoint capacitated multi-item, single-machine, single-period problems. The Lagrangian dual is

$$LD_3 = \max_{\gamma\geq 0,\beta} LR_3(\beta,\gamma) \tag{55}$$

In the next section we provide theoretical comparisons for the various approaches we have described.

3 Exploring Relationships

Let the superscript LP indicate the LP relaxation of a problem, i.e., the binary variables y relaxed to be continuous with the bounds $0 \leq y \leq 1$. For example, Z_{LS}^{LP} is the problem Z_{LS} with the integrality requirements for y variables relaxed. Similarly, X_{LS}^{LP} is the polyhedron of the LP relaxation of X_{LS}.

Theorem 1 (Akartunalı and Miller [1]) $Z_{LS}^{LP} = Z_{FL}^{LP} = Z_{SP}^{LP}$, *i.e., the* (ℓ, S) *inequalities, the facility location reformulation, and the shortest path reformulation all provide the same lower bound for the original problem.*

For the proof of the theorem, please refer to Akartunalı [2]. The proof uses Lagrangian duality and the fact that all these formulations provide equal lower bounds in the single-item case. See Krarup and Bilde [23], Eppen and Martin [15], and Barany et al. [6] for the convex hull and integrality proofs in the single-item case.

Theorem 2 $Z^{LP}_{MC} \geq Z^{LP}_{FL}$, *i.e., the multi-commodity reformulation provides a lower bound that is at least as strong as that provided by the facility location reformulation. If the problem consists of a single level, then* $Z^{LP}_{MC} = Z^{LP}_{FL}$.

Although this result has been known by at least some researchers since the publication of Rardin and Wolsey [39], it has never been formally stated and proven, to the best of our knowledge. We therefore provide a proof for the sake of completeness.

Proof We will prove this by showing that $proj_{x,y,E}(X^{LP}_{MC}) \subseteq proj_{x,y,E}(X^{LP}_{FL})$ for the multi-level case. Let $(v^*, w^*, x^*, y^*, E^*) \in X^{LP}_{MC}$. First, observe that we can eliminate v^* and rewrite (45)-(48) in terms of w^*, as follows:

$$\sum_{t=1}^{t=t'} {w^*}^{i,j}_{t,t'} = r^{ij} d^j_{t'} \qquad t' \in [1, NT], i \in [1, NI], j \in endp \tag{56}$$

Now, let

$${u^*}^i_{t,t'} = \sum_{j \in endp} {w^*}^{ij}_{t,t'} \tag{57}$$

Obviously $u^* \geq 0$ since $w^* \geq 0$. Since w^* satisfies (43), ${x^*}^i_t = \sum_{t'=t}^{NT} {u^*}^i_{t,t'}$. Similarly, summing (56) over $j \in endp$, we obtain $\sum_{t=1}^{t'} {u^*}^i_{t,t'} = \sum_{j \in endp} r^{ij} d^j_{t'} = D^i_{t'}$, where the second equation follows from the definition of echelon demand (9). Finally, using (44) and (57), we obtain ${u^*}^i_{t,t'} = \sum_{j \in endp} {w^*}^{ij}_{t,t'} \leq (\sum_{j \in endp} r^{ij} d^j_{t'}) {y^*}^i_t = D^i_{t'} {y^*}^i_t$. This shows that $(u^*, x^*, y^*, E^*) \in X^{LP}_{FL}$. Hence, $proj_{x,y,E}(X^{LP}_{MC}) \subseteq proj_{x,y,E}(X^{LP}_{FL})$. □

The second part of the theorem can also be shown using the same technique as in the proof of first theorem, i.e., using Lagrangian duality and the fact that the multi-commodity reformulation and the facility location reformulation provide equivalent lower bounds in the single-item case (see Eppen and Martin [15] and Barany et al. [6]).

This theorem shows us theoretically that the multi-commodity reformulation is stronger than the formulation defined by adding (ℓ, S) inequalities, the facility location reformulation, and the shortest path reformulation. In the next section, we will computationally address the question of "how much stronger" for a variety of test problems.

So far we have made comparisons of different polyhedral approaches. Also interesting are the relationships between the Lagrangian approaches and these reformulations, as we investigate in the following results.

Theorem 3 $Z^{LP}_{MC} \leq LD_1$.

In words, the lower bound obtained by the Lagrangian that relaxes the capacity constraints is at least as strong as the lower bound obtained by multi-commodity reformulation.

Proof By the theorem related to the strength of the Lagrangian dual (see e.g. Theorem 10.3 of Wolsey [49]),

$$LD_1 = \min\{(14)|(x, y, E) \in (4) \cap conv((5) - (7), (11) - (13))\}$$

On the other hand,

$$\begin{aligned} Z_{MC}^{LP} = \quad & \min\{(14)|(x, y, E, w, v) \in (4) \cap \{((5), (7), (11) - (13), (43) - (49)) \\ & \cap\ conv((6))\}\} \end{aligned}$$

Observe that

$$\begin{aligned} &\{(x, y, E) \in conv((5) - (7), (11) - (13))\} \subseteq \\ &proj_{x,y,E}\{(x, y, E, w, v) \in \{((5), (7), (11) - (13), (43) - (49)) \cap\ conv((6))\}\} \end{aligned}$$

This follows because $conv((5) - (7), (11) - (13))$ has integer extreme points because the polyhedron is the convex hull of an integer feasible region. On the other hand, $\{((5), (7), (11) - (13), (43) - (49)) \cap\ conv((6))\}$ does not necessarily have integral extreme points. Therefore, $Z_{MC}^{LP} \leq LD_1$. □

Theorem 4 $Z_{FL}^{LP} \leq Z_{FL}^{KN} \leq LD_2$.

In words, the lower bound obtained by the Lagrangian that relaxes the level linking constraints is at least as strong as the lower bound obtained by the facility location reformulation strengthened to approximate the knapsack convex hulls.

Proof The first relationship follows from the fact that Z_{FL}^{KN} is obtained by strengthening Z_{FL}^{LP} with additional constraints. For the second relationship, first observe that (using the same theorem as in the previous proof)

$$LD_2 = \min\{(14)|(x, y, E) \in (12) \cap conv((4) - (7), (11), (13))\}$$

Observe also that

$$\begin{aligned} &conv((4) - (7), (11), (13)) \subseteq \\ &proj_{x,y,E}\left\{\{(x, y, E, u)|(5), (7), (11), (13), (17), (18)\} \cap conv(\bigcap_{t=1}^{NT}\bigcap_{k=1}^{NK} X_{KN}^{(t,k)})\right\} \end{aligned}$$

This concludes that Z_{FL}^{KN} is not as strong as LD_2. □

As mentioned before, generating cover cuts from (22) only approximates the knapsack polyhedron and hence Z_{FL}^{KN} is the best possible bound that can be obtained by adding cover cuts to the LP relaxation of the facility location reformulation.

Theorem 5 $Z_{FL}^{KN} = LD_3$.

We will use the following result for the proof of the theorem.

Lemma 6 (Pochet and Wolsey [35]) *All optimal solutions of the single-item uncapacitated problem formulated using the facility location reformulation have the following property:*

$$\frac{u_{t,t'}}{D_{t'}} \geq \frac{u_{t,t'+1}}{D_{t'+1}} \qquad t \in [1, NT], t' \geq t$$

Before starting the proof of Theorem 5, let $S_1 = \bigcap_{t=1}^{NT} \bigcap_{k=1}^{NK} X_{KN}^{(t,k)} = \{(y,u)|(6),\ (16),(19),(20)\}$ and $S_2 = \{(y,z)|(37),(38),(40),(41)\}$. Also let $T_1 = \{(x,y,E,u)|((11)-(13),(18)) \cap conv(S_1)\}$ and $T_2 = \{(x,y,E,z)|((11)-(13),(33)) \cap conv(S_2)\}$. Note that S_1 and S_2 are integer feasible regions whereas T_1 and T_2 are both polyhedra. Then, the proof of Theorem 5 follows.

Proof We will prove this by showing $proj_{x,y,E}(T_1) = proj_{x,y,E}(T_2)$, and by the fact that $LD_3 = \min\{(14)|(x,y,E,z) \in T_2\}$.

First, let $(x^*, y^*, E^*, u^*) \in T_1$ and hence $(x^*, y^*, E^*) \in proj_{x,y,E}(T_1)$. Therefore, $\exists p^j = (x^j, y^j, E^j, u^j) \in S_1$, $j \in [1, J]$, such that $(x^*, y^*, E^*, u^*) = \sum_{j=1}^{J} \lambda_j p^j$ for some $\lambda \geq 0$, $\sum_{j=1}^{J} \lambda_j = 1$.

For all $j \in [1, J]$, let $\{z^i_{tNT}\}^j = \frac{\{u^i_{tNT}\}^j}{D^i_{NT}}$, where $t \in [1, NT]$ and $i \in [1, NI]$. Then, define recursively $\{z^i_{t,t'}\}^j = \frac{\{u^i_{t,t'}\}^j}{D^i_{NT}} - \sum_{\bar{t}=t'+1}^{NT} \{z^i_{t,\bar{t}}\}^j$, for all $t \in [1, NT]$, $t' = NT-1, ..., t$ and $i \in [1, NI]$. Since $\sum_{t'=t}^{NT} D^i_{t,t'} \{z^i_{t,t'}\}^j = \sum_{t'=t}^{NT} \{u^i_{t,t'}\}^j$ and u^j satisfies (20), z^j satisfies (38). Next, note that

$$\sum_{t'=t}^{NT} \{z^i_{t,t'}\}^j = \frac{\{u^i_{t,t}\}^j}{D^i_t} \leq \{y^i_t\}^j$$

where the last inequality is essentially (16). Finally, using Lemma 6, observe that

$$\{z^i_{t,t'}\}^j = \frac{\{u^i_{t,t'}\}^j}{D^i_{t'}} - \frac{\{{u^*}^i_{t,t'+1}\}^j}{D^i_{t'+1}} \geq 0$$

Therefore, $\hat{p}^j = (x^j, y^j, E^j, z^j) \in S_2$, and using the same λ as before, $(x^*, y^*, E^*, z^*) = \sum_{j=1}^{J} \lambda_j \hat{p}^j \in T_2$. Hence, $(x^*, y^*, E^*) \in proj_{x,y,E}(T_2)$. We conclude therefore that $proj_{x,y,E}(T_1) \subseteq proj_{x,y,E}(T_2)$.

Now, let $(x^*, y^*, E^*, z^*) \in T_2$ and hence $(x^*, y^*, E^*) \in proj_{x,y,E}(T_2)$. Therefore, $\exists q^k = (x^k, y^k, E^k, z^k) \in S_2$, $k \in [1, K]$, such that $(x^*, y^*, E^*, z^*) = \sum_{k=1}^{K} \mu_k q^k$ for some $\mu \geq 0$, $\sum_{k=1}^{K} \mu_k = 1$.

For all $k \in [1, K]$, let $\{u^i_{t,t'}\}^k = D^i_{t'} \sum_{\bar{t}=t'}^{NT} \{z^i_{t,\bar{t}}\}^k$, where $t \in [1, NT]$, $t' \in [t, NT]$, and $i \in [1, NI]$. Obviously, u^k satisfies (19) since z^k satisfies (40). Since $\sum_{t'=t}^{NT} \{u^i_{t,t'}\}^k = \sum_{t'=t}^{NT} D^i_{t,t'} \{z^i_{t,t'}\}^k$ and z^k satisfies (38), u^k satisfies (20). Finally, note that

$$\{u^i_{t,t'}\}^k = D^i_{t'} \sum_{\bar{t}=t'}^{NT} \{z^i_{t,\bar{t}}\}^k \leq D^i_{t'} \sum_{\bar{t}=t}^{NT} \{z^i_{t,\bar{t}}\}^k \leq D^i_{t'} \{y^i_t\}^k$$

where the last inequality follows from (37).

Therefore, $\hat{q}^k = (x^k, y^k, E^k, u^k) \in S_1$, and using the same μ as before, $(x^*, y^*, E^*, u^*) = \sum_{k=1}^{K} \mu_k \hat{q}^k \in T_1$. Hence, $(x^*, y^*, E^*) \in proj_{x,y,E}(T_1)$. Therefore, $proj_{x,y,E}(T_2) \subseteq proj_{x,y,E}(T_1)$. This concludes the proof. $\square$

Corollary 7 $LD_3 \leq LD_2$.

The proof for this corollary follows immediately from the Theorems 4 and 5. This result is our main motivation for skipping LD_3 in the computational tests discussed in the next section.

Proposition 8 *For any given* (t, k) *pair and set of* $\{t(i)\}$ *values,*

$$proj_{x,y,E}(conv(X_{PI}^{(t,k,\{t(i)\})})) = proj_{x,y,E}(conv(\bar{X}_{KN}^{(t,k,\{t(i)\})}))$$

This result, combined with Corollary 7, is our main motivation for omitting computationally testing the cover and reverse cover inequalities from Miller et al. [32, 33] in the next section.

Proof First show $proj_{x,y,E}(conv(\bar{X}_{KN}^{(t,k,\{t(i)\})})) \subseteq proj_{x,y,E}(conv(X_{PI}^{(t,k,\{t(i)\})}))$ for a given (t, k) pair and set of $\{t(i)\}$ values. Let $(x^*, y^*, E^*, u^*) \in conv(\bar{X}_{KN}^{(t,k,\{t(i)\})})$. Then, we define $S^{*i} = E^{*i}_{t-1} + \sum_{\hat{t}=t+1}^{t(i)} D^i_{\hat{t},t(i)} y^{*i}_{\hat{t}}$. It is easy to observe that $(x^*, y^*, S^*) \in conv(X_{PI}^{(t,k,\{t(i)\})})$.

Next we prove $proj_{x,y,E}(conv(X_{PI}^{(t,k,\{t(i)\})})) \subseteq proj_{x,y,E}(conv(\bar{X}_{KN}^{(t,k,\{t(i)\})}))$ for any given (t, k) pair and set of $\{t(i)\}$ values. First, let $(x^*, y^*, S^*) \in conv(X_{PI}^{(t,k,\{t(i)\})})$. We define first $u^{*i}_{t_1,t_2} = D^i_{t_2} y^{*i}_{t_1}$ for all $t_1 \in [t+1, t(i)]$ and $t_2 \in [t_1, t(i)]$. Then, we define $E^{*i}_{t-1} = (S^{*i} - \sum_{\hat{t}=t+1}^{t(i)} D^i_{\hat{t},t(i)} y^{*i}_{\hat{t}})^+$. Finally, define $u^{*i}_{t,t'} = (\min\{D^i_{t'} y^{*i}_t, x^{*i}_t - \sum_{\bar{t}=t}^{t'-1} u^{*i}_{t,\bar{t}}\})^+$ for all $t' \in [t, t(i)]$, where they are calculated in the increasing order of t'. Then, we can observe that $(x^*, y^*, E^*, u^*) \in conv(\bar{X}_{KN}^{(t,k,t(i))})$. $\square$

4 Computational Results

4.1 Overview

In order to provide diversified results, we used the following test instances for our computations:

- **TDS instances**: These test problems originate from Tempelmeier and Derstroff [44] and Stadtler [43]. These include overtime variables in addition to the formulation in Section 2. Sets A+ and B+ involve problems with 10 items and 24 periods, and sets C and D involve problems with 40 items and 16 periods. Sets B+ and D include setup times. We chose the hardest instances from each data set for our computations, i.e., for each data set, we picked 10 assembly and 10 general instances with the highest duality gaps according to results of Stadtler [43].

- **LOTSIZELIB instances**: These are the multi-level instances of LOTSIZELIB [27]. These include big bucket capacities, and backlogging is also allowed. The problems vary between 40 item, single end-item problems and 15 item, 3 end-item problems. All problems have 12 periods.
- **Multi-LSB instances**: We have generated 4 sets of test problems based on the problem family described in Simpson and Erenguc [42], each set having 30 instances with low, medium and high variability of demand. We will refer to these sets as SET1, SET2, SET3, and SET4 in the remainder of the paper. The main difference of these instances is that they consider component commonality and hence joint setup variables for each "family of item" (a set of items that are grouped together due to similarity) exist, i.e., setup time of a family of items is exhausted only once as soon as any item from that family is produced. The original BOM structures and holding costs of [42] are preserved, while the setup costs are removed. Moreover, these instances have backlogging variables and hence increase the variety of our test bed. Except for the problems in SET2, which consider a horizon of 24 periods, all the instances have 16 periods. The main difference between SET1, SET2 and SET4 is about resource utilization factors, which are all set over 100% for obtaining hard problem instances. All problems have 78 items and an assembly BOM structure, and all instances allow backlogging to the last period. For more details about these instances, including a full formulation, see Multi-LSB homepage [34].

Note that average duality gaps after default times (see next section for more detail on "default times") for the test sets of TDS and Multi-LSB are provided in the Table 1 for an overview of problem complexity, where the basic formulation is strengthened with all violated (ℓ, S) inequalities generated at the root node of the Branch&Bound tree using Algorithm 1.

Table 1 Average duality gaps for TDS and Multi-LSB instances

A+	B+	C	D	SET1	SET2	SET3	SET4
25.28%	34.21%	35.40%	364.57%	17.40%	13.84%	236.36%	78.87%

The main goal of this section is to computationally test the results we have theoretically proven and to observe how these strength relationships work in practice. This not only provides us with information about how strong the lower bounds actually are but also helps us to understand what prevents us from improving them. All the test instances are run on a PC with an Intel Pentium 4 2.53 GHz processor and 1 GB of RAM. All the formulations are implemented using Xpress Mosel (Xpress-MP 2004C, Mosel version 1.4.1).

In evaluating Lagrangians, we do not exactly solve any of the Lagrangian dual problems and solve an approximation instead, as detailed in the next paragraph. The main reason to avoid calculating Lagrangian duals is the significant computational effort needed, as this exact calculation will require subgradient optimization to choose the optimal Lagrangian multipliers. Subgradient optimization does not have a guarantee for convergence (e.g. noted by [21]) and it might require a very high number of iterations (each iteration being a Lagrangian relaxation problem) to converge to a bound. This is prohibitive in our case, as Lagrangian problems prove not to be easy to solve to optimality in short computational times: As we will also see later in computational results, even for the smaller TDS test sets of A+ and B+, only one instance (namely AK501432) solved to optimality for the 1st Lagrangian relaxation problem in the given 180 seconds time limit. Moreover, ad-hoc testing of subgradient optimization for Lagrangian duals on a few small A+ an B+ instances did not seem to converge efficiently to a bound within CPU times of 2 to 7 hours.

For the approximation to Lagrangian duals, we first consider a strengthened LP formulation, i.e., the echelon formulation with all violated (ℓ, S) inequalities generated at the root node, and then fix the Lagrangian multipliers to the values of the optimal dual variables of the constraints to be relaxed in this formulation. We thus evaluate $LR_1(\lambda^*)$ and $LR_2(\mu^*)$, respectively, for the optimal dual variables λ^* of the capacity constraints and the optimal dual variables μ^* of the level-linking constraints, respectively, in order to approximate LD_1 and LD_2, respectively. These subproblems themselves are MIPs that, in general, are difficult to solve to optimality, as can be seen in computational results. Nevertheless, any lower bound on the optimal solution of the Lagrangian subproblem MIP is also a lower bound on the Lagrangian dual (and hence the original problem), i.e., $LR_1(\lambda^*) \leq LD_1 \leq Z$ and $LR_2(\mu^*) \leq LD_2 \leq Z$. Moreover, in every instance, for both $LR_1(\lambda^*)$ and $LR_2(\mu^*)$, the lower bound obtained computationally for the Lagrangian subproblem MIP is at least as strong as the lower bound provided by the original echelon formulation strengthened with (ℓ, S) inequalities. We note that this is the only theoretical strength we are aware of for using these multipliers. Finally, although this is a limited computational experience and cannot necessarily generalize to other instances, our ad-hoc testing of subgradient optimization indicated that the bounds obtained using λ^* and μ^* can be very competitive.

Similarly, as we discussed before, generating cover cuts on top of the facility location reformulation provides only an approximation of Z_{FL}^{KN}. Hence, the computational comparisons we provide for these relationships are all based on approximations. However, this still gives us the chance to compare empirical results in addition to theoretically proven relationships.

4.2 Results

The detailed results for TDS instances can be found in the "Online Supplement". Note that we obtain the root node solution of the Branch&Bound tree for (ℓ, S) inequalities, all generated through Algorithm 1, and for the multicommodity reformulation (MC), without the effect of any solver cuts. For the facility location reformulation (FL), all the cover cuts generated by the solver are added at the root node and this strengthened formulation is used as FL lower bound. For comparison purposes, we also use the lower bound obtained by the heuristic in our companion paper (Akartunalı and Miller [1]), where the lower bound is based on the first iteration of a relax-and-fix framework, i.e., a partial LP relaxation of the original problem. For the Lagrangian relaxations that relax the capacity and level-linking constraints, we use the dual optimal values of the constraints from the strong LP relaxation as multipliers, and we set default times of 180 seconds for A+ and B+ instances, and 500 seconds for C and D instances. Note that if a Lagrangian relaxation subproblem ($LR_1(\lambda^*)$ or $LR_2(\mu^*)$, referred as LR1 and LR2 in the discussion, resp.) is not solved to optimality in this preassigned time, the lower and upper bounds (denoted by the functions $LB(.)$ and $UB(.)$, resp.) of this Lagrangian subproblem provide us the range where the actual lower bound of this Lagrangian relaxation lies, since $LB(LR_1(\lambda^*)) \leq LR_1(\lambda^*) \leq UB(LR_1(\lambda^*))$ and $LB(LR_2(\mu^*)) \leq LR_2(\mu^*) \leq UB(LR_2(\mu^*))$ obviously hold, while $LR_1(\lambda^*) \leq LD_1 \leq Z$ and $LR_2(\mu^*) \leq LD_2 \leq Z$. Therefore, we use the lower and upper bounds of Lagrangian subproblems in our discussions. One important remark here is that these upper bounds $UB(LR_1(\lambda^*))$ and $UB(LR_2(\mu^*))$ do not provide any information on the original problem Z. Finally, note that due to Theorem 1 we omit the shortest path reformulation in our tests.

We review the results in pairwise comparisons, which are summarized in Table 2 (for detailed results, refer to Tables 1-4 of "Online Supplement"). One interesting computational comparison is the relationship we have proven in Theorem 2. As we can see from the detailed results, MC improves the (ℓ, S) bound slightly, in general less than %1. The average improvements from the (ℓ, S) inequalities bound to the MC bound, calculated as (MC bound - ℓ, S bound)/(ℓ, S bound) for each test instance, are provided in the column "MC vs. ℓ, S", and these values are around 0.20%. Considering the enormous size of the MC reformulation, these improvements are simply not worth the computational effort. The Lagrangian relaxation LR1 that relaxes the capacity constraints (i.e., $LR_1(\lambda^*)$) provides in general another slight improvement over the lower bounds of the MC reformulation, as can be seen in the second column of the same table (Column LB under "LR1 vs. MC"), which is calculated in a similar fashion, i.e., ($LB(LR_1(\lambda^*))$ - MC bound)/(MC bound). Note that we also provide averages calculated in the same way using the LR1's upper bounds (Column UB under "LR1 vs. MC"), i.e. ($UB(LR_1(\lambda^*))$ - MC bound)/(MC bound). An interesting observation regarding the problems in set D, where all $LR_1(\lambda^*)$ problems are solved to optimality, is that although $LR_1(\lambda^*)$ provided improvements over the MC bounds for instances outwith set D, the same effect was not observed in set D instances. This is due to the fact that $LR_1(\lambda^*)$ is only an approximation of LD_1, and therefore it does not necessarily provide a theoretically stronger bound than MC bound. However, as these results indicate, $LR_1(\lambda^*)$ and MC bounds are in general very close to each other in our computational results.

On the other hand, as the "FL vs. ℓ, S" column of Table 2 indicates, the facility location reformulation with cover cuts added (FL) improves in gen-

Table 2 Pairwise comparisons of lower bounds and LR gaps for TDS instances

Test Set	MC vs. ℓ, S	LR1 vs. MC LB	LR1 vs. MC UB	FL vs. ℓ, S	LR2 vs. FL LB	LR2 vs. FL UB	LR Gaps LR1	LR Gaps LR2
A+	0.29%	0.80%	2.99%	1.81%	-0.05%	7.44%	2.09%	6.87%
B+	0.28%	0.59%	3.06%	1.37%	-0.35%	6.23%	2.38%	6.18%
C	0.14%	0.20%	1.67%	0.86%	-0.32%	6.25%	1.44%	6.14%
D	0.21%	-0.06%	-0.06%	0.45%	-0.43%	19.88%	0%	15.85%

eral the (ℓ, S) bound more significantly compared to previous methods. These average percentages are calculated by (FL bound - ℓ, S bound)/(ℓ, S bound). Similar to our previous comparisons, we also provide the average improvements of the Lagrangian relaxation LR2 that relaxes level-linking constraints (i.e., $LR_2(\mu^*)$) over the FL bound in the column "LR2 vs. FL", calculated by ($LB(LR_2(\mu^*))$ - FL bound)/(FL bound). Although one would expect the LR2, the approximation of LD_2, to improve the FL lower bounds, at first sight this does not seem to be the case for many problem instances, particularly due to negative averages in the LB column of Table 2. However, as can be seen from the UB column of the table, which indicates ($UB(LR_2(\mu^*))$ - FL bound)/(FL bound), these Lagrangian problems are far from optimality, particularly the bigger instances of test sets C and D, and the challenge here is that these problems need much more time than the assigned default times (or any reasonable amount of time) for optimality or even for an acceptable gap. For testing whether this is the case here, we experimented with a few randomly selected instances from sets A+ and B+ that did not achieve the FL bounds earlier and ran them either until the lower bound was at least as strong as the FL bound or to optimality. For the instances we took for this ad-hoc test, we ended up with bounds that reached at least FL bounds, though we would not be able to generalize this as this was simply for a small subset of the test problems, due to high computational effort. Furthermore, this experiment failed due to memory problems for the few instances from sets C and D and hence could not be completed.

Finally, the last two columns of Table 2 should also be addressed briefly. These columns indicate the duality gaps for the two Lagrangian relaxation problems, which can be defined as:

$$[UB(LR_1(\lambda^*)) - LB(LR_1(\lambda^*))]/LB(LR_1(\lambda^*))$$

$$[UB(LR_2(\mu^*)) - LB(LR_2(\mu^*))]/LB(LR_2(\mu^*))$$

Note that these gaps are not related to the original problem and only indicate the problem complexity of these Lagrangian subproblems. As we mentioned before, the LR1 problem is in general comparatively easier to solve than the LR2 problem. We had a total of 11 instances where the LR1 could solve optimally in the assigned default times, compared to none for the LR2.

Next, we present results for LOTSIZELIB instances in Table 3, where all values are shown explicitly, including the optimal solutions (OPT) in the last column. The table also has a "Heur" column for comparison purposes, which is the lower bound obtained by the heuristic in our companion paper (Akartunalı and Miller [1]), calculated from the first iteration of a relax-and-fix framework. MC provides significant improvement over the (ℓ, S) bound for some of these instances, whereas FL provides negligible improvement over MC. The LR1 is comparatively more efficient on these instances than the LR2. Note that LR1 and LR2 do not necessarily improve MC and FL bounds respectively, similarly to the results for some TDS instances, since these are approximations for LD_1 and LD_2. Also, note that all LR2 problems solved optimally for most of the instances, whereas LR1 problems did not finish in quite a few instances after the default time of 180 seconds. This indicates that these instances have the bottleneck not in capacity constraints but in the multi-level structure. This seems to be due in part to the fact that there is a single machine, and the capacity in these problems is comparatively loose.

Table 3 LOTSIZELIB results

	Lower Bounds				**LR1 (Cap)**		**LR2 (Lev)**		
	ℓ, S	MC	FL	Heur [1]	LB	UB	LB	UB	OPT
B	3,888	3,890	3,892	3,915	3,888	3,888	3,888	3,888	3,965
C	1,904	1,993	1,998	2,067	1,904	1,904	1,904	1,905	2,083
D	4,534	4,794	4,795	4,714	4,766	6,095	4,534	4,535	6,482
E	2,341	2,361	2,361	2,416	2,462	3,136	2,341	2,341	2,801
F	2,075	2,098	2,111	2,099	2,237	2,459	2,079	2,079	2,429

The detailed results on Multi-LSB instances can be seen in the Tables 5-10 of "Online Supplement", and the pairwise comparisons are summarized in Table 4, which is organized in the same fashion as Table 2. The default times for the first two sets are 180 seconds, and for the last two sets 500 seconds. First of all, note that MC improves the (ℓ, S) bound poorly in most of the instances. Also note that the LR1 is solved to optimality for all these test problems, and as the table indicates, this approximation of LD_1 does not often provide an improvement over MC. This might be due in part to poor multipliers generated from the (ℓ, S) formulation (also recall that these instances have backlogging variables).

On the other hand, FL improves in general the (ℓ, S) bound more significantly than MC, although the improvements are still minuscule. Note that LR2 does not solve to optimality for many test instances, particularly for the hard problems. Similar to the LR1, the LR2 does not provide necessarily an improvement over FL bound, possibly due to poor multipliers. Compared to previous test problems, Multi-LSB instances are parallel to TDS problems, where the bottleneck lies in the capacities rather than the multi-level structure of these problems.

Table 4 Pairwise comparisons of lower bounds and LR gaps for Multi-LSB instances

Test Set	MC vs. ℓ, S	LR1 vs. MC LB	LR1 vs. MC UB	FL vs. ℓ, S	LR2 vs. FL LB	LR2 vs. FL UB	LR Gaps LR1	LR Gaps LR2
SET1	0.02%	-0.02%	-0.02%	0.85%	-0.29%	-0.28%	0.00%	0.01%
SET2	0.06%	-0.06%	-0.06%	0.28%	-0.11%	-0.05%	0.00%	0.06%
SET3	6.28%	-4.27%	-4.27%	6.11%	-5.14%	24.83%	0.00%	21.92%
SET4	1.23%	-1.14%	-1.14%	3.40%	-0.99%	4.34%	0.00%	4.76%

4.3 Summary

One of our main goals of this paper was to understand the structure of production planning problems and the underlying difficulties that make these problems very hard. In general, the Lagrangian relaxations we tested are helpful for this. First of all, recall that in general the Lagrangian relaxation that relaxes capacity constraints, i.e., $LR1(\lambda^*)$, provides only slight improvement over the (ℓ, S) bound. Also recall that $LR1(\lambda^*)$ values provide a lower bound to LD_1. The $LR1(\lambda^*)$ bound can be seen as an approximation to the convex hull of the uncapacitated problem polyhedron, and our computational results indicate that removing capacities makes the problem much easier. This can also be observed by recalling that the final gaps after the default times were quite small for this Lagrangian relaxation in general.

On the other hand, the facility location reformulation with cover cuts and the Lagrangian relaxation that relaxes the level-linking constraints (although only an approximation to the Lagrangian dual) seem to improve the lower bounds much more significantly. Recall that the cover cuts approximate the intersection of all knapsack sets included in the problem, and $LR2(\mu^*)$ provides an approximation to the convex hull of the single-level capacitated polyhedrons within the overall multi-level problem. Having higher duality gaps compared to the LR1, this Lagrangian relaxation problem is in general much harder to solve, indicating that the level-linking constraints are not the bottleneck of these problems. A similar comparison is achieved by Jans and Degraeve [21] for single-level problems, where their Lagrangian relaxation relaxing only period-linking constraints is a harder problem than the one that relaxes capacities. Recall that we did not report computational results on LD_3, due to the result presented in Corollary 7.

5 Conclusion

In this paper, we have provided an extensive survey of different methodologies for obtaining lower bounds for big bucket production planning problems, and presented both theoretical and computational comparisons of them.

In summary, it seems that the multi-level structure by itself makes some of our problems challenging to solve. However, for most instances, and in particular for the most challenging, the single-level, capacitated substructures are clearly a much greater contributor to problem difficulty. It is this substructure for which the tools currently at our disposal are evidently not sufficient.

These observations indicate that the main bottleneck with these problems lies in the fact that there is no efficient polyhedral approximation of the multi-item, multi-period, single-level, single-machine capacitated problems. It seems that if we could solve these problems well or even adequately, our ability to solve multi-level bug bucket problems would increase dramatically. While initial efforts to find strong formulations for these problems have been made (e.g. see Miller et al. [32]), this is a fundamental area in which it is crucial for the research community to improve the current state of the art. We will attempt to make contributions in this direction in future research.

Acknowledgements We would like to thank two anonymous referees for their constructive comments that helped us to improve the presentation of the paper.

References

1. K. Akartunalı and A.J. Miller. A heuristic approach for big bucket multi-level production planning problems. *European Journal of Operational Research*, 193:396–411, 2009.
2. K. Akartunalı. Computational methods for big bucket production planning problems: Feasible solutions and strong formulations. PhD thesis, *Industrial and Systems Engineering Department, University of Wisconsin-Madison*, 2007.
3. S. Anily, M. Tzur and L.A. Wolsey. Multi-item lot-sizing with joint set-up costs. *Mathematical Programming*, 119(1):79-94, 2009.
4. A. Atamtürk and J.C. Muñoz. A study of the lot-sizing polytope. *Mathematical Programming*, 98:443–465, 2004.
5. I. Barany, T.J. Van Roy, and L.A. Wolsey. Strong formulations for multi-item capacitated lot-sizing. *Management Science*, 30(10):1255–1261, 1984.
6. I. Barany, T.J. Van Roy, and L.A. Wolsey. Uncapacitated lot sizing: The convex hull of solutions. *Mathematical Programming Study*, 22:32–43, 1984.
7. G. Belvaux and L.A. Wolsey. bc-prod: A specialized branch-and-cut system for lot-sizing problems. *Management Science*, 46(5):724–738, 2000.
8. G. Belvaux and L.A. Wolsey. Modelling practical lot-sizing problems as mixed-integer programs. *Management Science*, 47(7):993–1007, 2001.
9. P.J. Billington, J.O. McClain and L.J. Thomas. Heuristics for multilevel lot-sizingwith a bottleneck. *Management Science*, 32:989-1006, 1986.
10. G.R. Bitran and H. Matsuo. The multi-item capacitated lot size problem: error bounds of Mannes formulations. *Management Science*, 32:350-359, 1986.
11. G.R. Bitran and H.H. Yanasse. Computational complexity of the capacitated lot size problem. *Management Science*, 28(10):1174–1186, 1982.
12. L. Buschkühl, F. Sahling, S. Helber and H. Tempelmeier. Dynamic capacitated lot-sizing problems: a classification and review of solution approaches. *OR Spectrum*, 32(2):231–261, 2010.
13. M. Constantino. A cutting plane approach to capacitated lot-sizing with start-up costs. *Mathematical Programming*, 75:353–376, 1996.
14. Z. Degraeve and R. Jans. A new Dantzig-Wolfe reformulation and branch-and-price algorithm for the capacitated lot-sizing problem with setup times. *Operations Research*, 55(5):909-920, 2007.
15. G.D. Eppen and R.K. Martin. Solving multi-item capacitated lot-sizing problems using variable redefinition. *Operations Research*, 35(6):832–848, 1987.
16. A. Federgruen and M. Tzur. A simple forward algorithm to solve general dynamic lot sizing models with n periods in O($nlogn$) or O(n) time. *Management Science*, 37(8):909–925, 1991.

17. A. Federgruen, J. Meissner and M. Tzur. Progressive interval heuristics for multi-item capacitated lot sizing problem. *Operations Research*, 55(3):490–502, 2007.
18. M. Florian and M. Klein. Deterministic production planning with concave costs and capacity constraints. *Management Science*, 18(1):12–20, 1971.
19. M. Florian, J.K. Lenstra, and H.G. Rinnooy Kan. Deterministic production planning: Algorithms and complexity. *Management Science*, 26(7):669–679, 1980.
20. O. Günlük and Y. Pochet. Mixing mixed integer inequalities. *Mathematical Programming*, 90:429–457, 2001.
21. R. Jans and Z. Degraeve. Improved lower bounds for the capacitated lot sizing problem with setup times. *Operations Research Letters*, 32:185–195, 2004.
22. E. Katok, H.S. Lewis, and T.P. Harrison. Lot sizing in general assembly systems with setup costs, setup times, and multiple constrained resources. *Management Science*, 44(6):859–877, 1998.
23. J. Krarup and O. Bilde. *Plant location, set covering and economic lotsizes: an O(mn) algorithm for structured problems*, pages 155–180. Optimierung bei Graphentheoretischen und Ganzzahligen Probleme. Birkhauser Verlag, 1977.
24. S. Küçükyavuz and Y. Pochet. Uncapacitated Lot-Sizing with Backlogging: The Convex Hull. *Mathematical Programming*, 118(1):151-175, 2009.
25. R. Levi, A. Lodi and M. Sviridenko. Approximation algorithms for the capacitated multi-item lot-sizing problem via flow-cover inequalities. *Mathematics of Operations Research*, 33(2):461–474, 2008.
26. M. Loparic, Y. Pochet, and L.A. Wolsey. The uncapacitated lot-sizing problem with sales and safety stocks. *Mathematical Programming*, 89:487–504, 2001.
27. LOTSIZELIB. Lot-sizing problems: A library of models and matrices. *http://www.core.ucl.ac.be/wolsey/lotsizel.htm*, 1999.
28. A.S. Manne. Programming of economic lot sizes. *Management Science*, 4(2):115135, 1958.
29. H. Marchand and L.A. Wolsey. The 0-1 knapsack problem with a single continuous variable. *Mathematical Programming*, 85(1):15–33, 1999.
30. H. Marchand and L.A. Wolsey. Aggregation and mixed integer rounding to solve mips. *Operations Research*, 49(3):363–371, 2001.
31. A.J. Miller. Polyhedral Approaches to Capacitated Lot-Sizing Problems. PhD thesis, *Industrial and Systems Engineering Department, Georgia Institute of Technology*, 1999.
32. A.J. Miller, G.L. Nemhauser, and M.W.P. Savelsbergh. Solving the multi-item capacitated lot-sizing problem with setup times by branch-and-cut. CORE Discussion Paper 2000/39, *CORE, UCL, Belgium*, 2000.
33. A.J. Miller, G.L. Nemhauser, and M.W.P. Savelsbergh. On the polyhedral structure of a multi-item production planning model with setup times. *Mathematical Programming*, 94:375–405, 2003.
34. Multi-LSB. Multi-item lot-sizing problems with backlogging: A library of test instances. Available at *http://personal.strath.ac.uk/kerem.akartunali/research/multi-lsb/*, 2010.
35. Y. Pochet and L.A. Wolsey. Lot-size models with backlogging: Strong reformulations and cutting planes. *Mathematical Programming*, 40:317–335, 1988.
36. Y. Pochet and L.A. Wolsey. Solving multi-item lot-sizing problems using strong cutting planes. *Management Science*, 37(1):53–67, 1991.
37. Y. Pochet and L.A. Wolsey. Polyhedra for lot-sizing with Wagner-Whitin costs. *Mathematical Programming*, 67:297–323, 1994.
38. Y. Pochet and L.A. Wolsey. *Production Planning by Mixed Integer Programming.* Springer, 2006.
39. R.L. Rardin and L.A. Wolsey. Valid inequalities and projecting the multicommodity extended formulation for uncapacitated fixed charge network flow problems. *European Journal of Operational Research*, 71:95–109, 1993.
40. J.-P.P. Richard, I.R. de Farias, and G.L. Nemhauser. Lifted inequalities for 0-1 mixed integer programming: Basic theory and algorithms. *Mathematical Programming*, 98(1-3):89–113, 2003.
41. J.-P.P. Richard, I.R. de Farias, and G.L. Nemhauser. Lifted inequalities for 0-1 mixed integer programming: Superlinear lifting. *Mathematical Programming*, 98(1-3):115–143, 2003.

42. N.C. Simpson and S.S. Erenguc. Modeling multiple stage manufacturing systems with generalized costs and capacity issues. *Naval Research Logistics*, 52:560–570, 2005.
43. H. Stadtler. Multilevel lot sizing with setup times and multiple constrained resources: Internally rolling schedules with lot-sizing windows. *Operations Research*, 51:487–502, 2003.
44. H. Tempelmeier and M. Derstroff. A lagrangean-based heuristic for dynamic multilevel multiitem constrained lotsizing with setup times. *Management Science*, 42(5):738–757, 1996.
45. J.M. Thizy and L.N. Van Wassenhove. Lagrangean relaxation for the multi-item capacitated lot-sizing problem: A heuristic implementation. *IIE Transactions*, 17(4):308-313, 1985.
46. W.W. Trigeiro, L.J. Thomas, and J.O. McClain. Capacitated lot sizing with setup times. *Management Science*, 35:353–366, 1989.
47. M. Van Vyve and Y. Pochet. A general heuristic for production planning problems. *INFORMS Journal of Computing*, 16(3):316–327, 2004.
48. H.M. Wagner and T.M. Whitin. Dynamic version of the economic lot size model. *Management Science*, 5:89–96, 1958.
49. L.A. Wolsey. *Integer Programming*. Wiley-Interscience, 1998.
50. L.A. Wolsey. Solving multi-item lot-sizing problems with an MIP solver using classification and reformulation. *Management Science*, 48(12):1587–1602, 2002.
51. W.I. Zangwill. A backlogging model and a multi-echelon model of a dynamic economic lot size production system-a network approach. *Management Science*, 15(9):506–527, 1969.

6 APPENDIX: Detailed Results (Online Supplement)

This section is prepared to present all detailed computational results that are too overwhelming for and therefore only summarized in the paper titled "A Computational Analysis of Lower Bounds for Big Bucket Production Planning Problems". It is aimed that this level of detail can help other researchers to get better insight, as well as have benchmark values when needed.

All the tables are structured the same way, as following: The first column indicates the specific name of the instance. The next four columns present the lower bound values obtained, in the order of (ℓ, S) inequalities (root node solution of the Branch&Bound tree), multi-commodity (MC) reformulation (without the effect of any solver cuts), facility location (FL) reformulation (with all cover cuts generated by the solver), and lower bound obtained by our heuristic (see reference [1] in the paper). Then, the next two columns provide the lower and upper bounds of the 1st Lagrangian problem (relaxing capacity), followed by the two columns presenting the lower and upper bounds of the 2nd Lagrangian problem (relaxing level). Finally, the last column indicates the best solution we are aware of. Note that for the Lagrangian relaxations, we use the dual optimal values of the constraints from the strong LP relaxation as multipliers, and we set default times of 180 seconds for A+, B+, SET1 and SET2 instances, and 500 seconds for C, D, SET3, SET4 instances. Note that if the Lagrangian relaxation subproblem is not solved to optimality in this preassigned time, the lower and upper bounds of this Lagrangian subproblem provide us the range where the actual lower bound of the Lagrangian relaxation lies.

Table 5 TDS Instances Detailed Computational Results, set A+

	Lower Bounds				**LR1 (Cap)**		**LR2 (Lev)**		Best
	ℓ, S	MC	FL	Heuristic	LB	UB	LB	UB	Soln
AG501130	116,183	116,600	118,340	119,146	117,808	123,203	120,764	127,683	153,418
AG501131	107,829	108,106	108,987	109,714	109,298	115,656	108,822	117,533	145,225
AG501132	118,677	118,957	119,986	121,740	120,163	123,663	120,454	128,249	154,191
AG501141	133,424	134,008	135,519	134,421	135,078	141,548	136,547	147,696	171,895
AG501142	145,508	145,873	147,646	148,911	146,527	151,197	149,002	156,488	192,582
AG502130	122,353	123,904	125,925	128,101	125,087	125,472	127,119	134,118	167,927
AG502131	109,085	109,501	110,500	111,001	111,043	116,443	109,959	121,005	145,322
AG502141	134,971	135,527	136,973	136,353	136,792	141,900	139,060	146,767	173,640
AG502232	97,032	97,488	97,890	97,632	98,529	101,859	98,206	102,415	121,108
AG502531	102,340	103,252	102,817	103,506	103,216	105,542	103,211	109,727	129,080
AK501131	96,968	96,983	99,966	99,020	97,892	98,030	97,811	112,060	123,366
AK501132	101,699	101,781	103,276	103,077	102,289	102,887	102,847	109,206	123,473
AK501141	134,805	134,943	139,399	136,428	135,487	136,315	137,303	163,011	170,897
AK501142	134,880	135,006	138,151	135,875	135,122	137,204	137,867	151,661	161,262
AK501432	92,533	92,605	92,968	93,546	94,679	94,679	93,270	93,645	109,249
AK502130	102,222	102,245	106,358	103,949	103,054	103,460	104,351	117,191	127,889
AK502131	93,369	93,423	95,912	94,969	93,778	94,145	94,338	101,804	115,819
AK502132	96,312	96,396	98,423	97,233	96,933	97,092	97,644	104,528	118,319
AK502142	127,792	127,977	129,654	129,034	128,226	130,758	129,863	138,752	146,616
AK502432	88,980	89,088	89,550	89,609	90,193	91,779	89,995	91,225	105,415

Table 6 TDS Instances Detailed Computational Results, set B+

	Lower Bounds				**LR1 (Cap)**		**LR2 (Lev)**		Best
	ℓ, S	MC	FL	Heuristic	LB	UB	LB	UB	Soln
BG511132	108,772	109,045	109,875	110,466	110,136	114,629	109,545	116,781	137,637
BG511142	133,158	133,652	134,424	133,880	134,500	137,991	134,648	146,913	159,769
BG512131	104,054	104,483	105,158	105,804	105,469	110,855	104,580	112,766	138,752
BG512132	114,786	115,314	115,894	116,135	115,931	119,395	115,156	125,132	151,770
BG512142	142,917	143,659	144,840	143,848	144,161	148,340	145,305	158,261	199,051
BG521132	108,324	108,559	109,338	110,024	109,805	113,609	109,109	115,077	138,133
BG521142	131,363	131,908	132,996	132,604	132,905	137,629	133,224	141,350	156,694
BG522130	113,540	114,876	116,472	121,578	115,240	119,850	115,961	123,968	154,581
BG522132	113,382	113,838	114,305	115,158	114,551	119,158	114,262	121,255	147,894
BG522142	137,126	137,782	138,608	138,077	138,405	142,417	138,851	144,180	186,268
BK511131	92,602	92,640	93,964	94,411	93,107	94,310	93,304	99,779	120,303
BK511132	95,323	95,355	97,283	95,938	95,942	96,844	96,310	103,668	115,416
BK511141	125,307	125,494	126,753	126,769	125,679	127,256	126,534	135,597	162,629
BK512131	90,733	90,787	92,253	92,058	91,391	92,036	91,568	96,009	113,536
BK512132	90,814	90,858	92,896	91,346	91,738	92,208	91,870	98,554	112,809
BK521131	92,350	92,382	93,469	94,164	92,881	94,004	92,884	97,318	118,217
BK521132	94,257	94,317	96,197	94,957	94,932	95,914	95,110	101,441	117,423
BK521142	124,988	125,257	126,384	125,480	125,333	128,448	126,548	134,871	153,805
BK522131	90,532	90,588	91,731	91,742	91,131	91,802	91,291	96,184	111,339
BK522142	119,559	119,739	120,794	119,625	120,047	124,160	120,956	127,283	148,471

Table 7 TDS Instances Detailed Computational Results, set C

	Lower Bounds				**LR1 (Cap)**		**LR2 (Lev)**		Best
	ℓ, S	MC	FL	Heuristic	LB	UB	LB	UB	Soln
CG501120	1,011,260	1,012,042	1,025,118	1,027,177	1,012,992	1,022,396	1,017,258	1,109,345	1,252,308
CG501131	472,421	472,711	475,464	478,437	473,125	476,392	472,947	513,188	614,303
CG501141	627,035	627,631	630,113	628,114	628,641	631,308	627,980	678,899	777,831
CG501121	945,696	946,442	953,112	959,756	948,052	953,730	946,612	1,045,688	1,247,493
CG502221	724,648	725,517	725,827	728,105	726,515	743,421	724,779	765,713	889,548
CG501132	561,827	562,158	566,137	606,568	562,887	567,636	567,379	597,061	842,734
CG501222	697,129	698,410	699,934	699,021	699,024	718,231	697,860	723,508	858,289
CG501142	754,238	757,449	761,826	824,887	757,128	758,835	764,794	802,021	1,146,638
CG501122	1,161,383	1,162,216	1,171,502	1,281,687	1,165,839	1,178,726	1,174,289	1,243,710	1,787,833
CG502222	704,096	705,161	707,153	708,597	706,766	725,192	704,971	753,284	873,858
CK501120	141,900	142,034	143,869	143,260	142,581	145,659	143,212	156,264	176,187
CK501221	101,028	101,108	101,570	101,105	101,299	103,024	101,114	106,030	123,066
CK501121	131,993	132,185	133,494	132,840	132,708	137,522	132,496	147,865	169,804
CK502221	101,478	101,740	102,242	101,899	101,968	103,730	101,623	107,423	122,596
CK501222	97,937	98,050	98,858	98,096	98,313	100,271	98,267	102,163	122,485
CK501422	101,864	102,007	102,660	102,150	102,135	102,981	103,846	107,102	124,315
CK502222	98,052	98,236	98,898	98,282	98,450	100,835	98,333	104,359	119,965
CK501122	153,861	154,358	156,048	155,485	154,841	155,914	155,016	165,574	206,646
CK501132	75,257	75,301	76,198	75,782	75,648	76,311	75,780	80,388	98,248
CK501142	90,218	90,347	91,277	90,673	90,477	91,215	90,701	96,230	115,918

Table 8 TDS Instances Detailed Computational Results, set D

	Lower Bounds				**LR1 (Cap)**		**LR2 (Lev)**		Best
	ℓ, S	MC	FL	Heuristic	LB	UB	LB	UB	Soln
DG512141	609,464	610,630	611,291	615,992	610,613	610,613	609,599	659,071	736,181
DG512131	465,272	466,156	466,203	469,460	466,333	466,333	465,372	495,481	581,932
DG012132	554,595	556,651	559,610	555,689	556,441	556,441	554,922	781,344	3,160,347
DG012142	756,588	758,120	763,304	756,588	757,387	757,387	756,898	1,001,177	3,121,762
DG012532	554,167	555,261	556,877	555,032	555,045	555,045	554,167	775,666	1,194,004
DG012542	756,062	756,956	759,793	756,062	756,563	756,563	756,159	982,363	1,413,476
DG512132	512,330	513,440	514,386	514,682	512,722	512,722	512,376	554,333	2,909,628
DG512142	678,733	679,821	681,450	682,205	679,062	679,062	678,777	854,902	3,583,354
DG512532	509,567	511,041	510,510	512,147	510,670	510,670	509,587	542,328	584,491
DG512542	674,241	675,180	675,969	677,189	674,734	674,734	674,241	715,533	767,428

Table 9 Multi-LSB Detailed Computational Results, SET1 instances 1-20

	Lower Bounds				**LR1 (Cap)**		**LR2 (Lev)**		Best
	ℓ, S	MC	FL	Heuristic	LB	UB	LB	UB	Soln
SET1_01	17,888	17,888	18,173	18,840	17,888	17,888	17,888	17,972	22,781
SET1_02	23,534	23,534	23,656	24,134	23,534	23,534	23,534	23,534	28,624
SET1_03	21,227	21,227	21,346	21,676	21,227	21,227	21,227	21,227	26,349
SET1_04	22,232	22,232	22,334	23,175	22,232	22,232	22,232	22,232	26,337
SET1_05	21,446	21,446	21,540	21,994	21,446	21,446	21,446	21,446	25,621
SET1_06	22,974	22,974	23,072	23,636	22,974	22,974	22,974	22,974	26,741
SET1_07	20,360	20,360	20,386	21,125	20,360	20,360	20,360	20,360	24,693
SET1_08	25,582	25,582	25,616	26,249	25,582	25,582	25,582	25,582	29,810
SET1_09	16,321	16,321	16,442	17,013	16,321	16,321	16,321	16,338	21,146
SET1_10	17,998	17,998	18,151	18,945	17,998	17,998	17,998	18,011	22,863
SET1_11	11,080	11,080	11,237	11,407	11,080	11,080	11,164	11,169	12,956
SET1_12	24,721	24,721	24,762	25,238	24,721	24,721	24,721	24,725	26,985
SET1_13	20,782	20,788	20,830	21,195	20,782	20,782	20,782	20,786	23,129
SET1_14	22,264	22,268	22,331	22,745	22,264	22,264	22,264	22,264	25,720
SET1_15	12,401	12,404	12,805	12,575	12,401	12,401	12,564	12,564	14,121
SET1_16	15,122	15,122	15,356	15,387	15,122	15,122	15,543	15,543	17,542
SET1_17	20,468	20,475	20,498	20,864	20,468	20,468	20,468	20,468	23,404
SET1_18	11,075	11,077	11,366	11,456	11,075	11,075	11,462	11,462	12,300
SET1_19	13,276	13,276	13,528	13,342	13,276	13,276	13,388	13,388	17,448
SET1_20	14,101	14,101	14,177	14,612	14,101	14,101	14,101	14,113	17,167

Table 10 Multi-LSB Detailed Computational Results, SET1 instances 21-30 and SET2 instances 1-10

	Lower Bounds				**LR1 (Cap)**		**LR2 (Lev)**		Best
	ℓ, S	MC	FL	Heuristic	LB	UB	LB	UB	Soln
SET1_21	10,159	10,166	10,429	10,392	10,159	10,159	10,325	10,325	12,421
SET1_22	38,040	38,056	38,166	38,040	38,040	38,040	38,040	38,077	40,158
SET1_23	29,331	29,343	29,376	29,355	29,331	29,331	29,331	29,331	30,606
SET1_24	28,858	28,858	29,074	29,250	28,858	28,858	28,886	28,886	32,174
SET1_25	51,371	51,371	51,403	51,371	51,371	51,371	51,371	51,371	53,009
SET1_26	39,379	39,379	39,463	39,488	39,379	39,379	39,402	39,402	41,442
SET1_27	40,838	40,838	40,838	40,918	40,838	40,838	40,838	40,838	43,320
SET1_28	39,846	39,864	39,894	40,144	39,846	39,846	39,857	39,857	40,993
SET1_29	23,155	23,165	23,275	23,232	23,155	23,155	23,182	23,182	25,606
SET1_30	68,989	68,989	69,074	68,989	68,989	68,989	68,989	68,989	70,868
SET2_01	46,116	46,116	46,207	46,591	46,116	46,116	46,116	46,116	55,039
SET2_02	47,780	47,780	47,861	48,159	47,780	47,780	47,780	47,780	57,825
SET2_03	40,551	40,551	40,610	40,814	40,551	40,551	40,551	40,551	49,147
SET2_04	36,347	36,347	36,564	36,808	36,347	36,347	36,347	36,430	44,656
SET2_05	45,395	45,395	45,508	45,784	45,395	45,395	45,395	45,395	55,650
SET2_06	45,902	45,902	45,939	45,902	45,902	45,902	45,902	45,902	54,361
SET2_07	52,825	52,825	52,939	53,108	52,825	52,825	52,825	52,825	61,140
SET2_08	48,033	48,033	48,280	48,632	48,033	48,033	48,084	48,084	56,444
SET2_09	37,553	37,553	37,661	37,943	37,553	37,553	37,553	37,553	44,523
SET2_10	38,751	38,751	38,898	39,181	38,751	38,751	38,751	38,751	49,481

Table 11 Multi-LSB Detailed Computational Results, SET2 instances 11-30

	Lower Bounds				**LR1 (Cap)**		**LR2 (Lev)**		Best
	ℓ, S	MC	FL	Heuristic	LB	UB	LB	UB	Soln
SET2_11	65,210	65,211	65,213	65,648	65,210	65,210	65,210	65,210	69,177
SET2_12	62,792	62,792	62,979	62,792	62,792	62,792	62,803	62,803	66,914
SET2_13	34,778	34,778	34,882	34,987	34,778	34,778	34,885	34,885	40,114
SET2_14	62,907	62,907	62,993	62,907	62,907	62,907	62,907	62,916	67,201
SET2_15	59,079	59,079	59,125	59,079	59,079	59,079	59,079	59,079	61,616
SET2_16	75,682	75,682	75,698	75,682	75,682	75,682	75,682	75,682	79,576
SET2_17	36,809	36,818	36,918	36,925	36,809	36,809	36,826	36,935	41,484
SET2_18	77,873	77,874	77,935	78,087	77,873	77,873	77,873	77,873	83,200
SET2_19	54,981	54,981	55,120	55,484	54,981	54,981	55,026	55,026	59,010
SET2_20	119,568	119,568	119,588	119,568	119,568	119,568	119,568	119,568	122,974
SET2_21	22,281	22,315	22,557	22,281	22,281	22,281	22,643	22,643	24,459
SET2_22	51,279	51,279	51,439	51,279	51,279	51,279	51,414	51,414	53,690
SET2_23	29,793	30,067	30,210	29,793	29,793	29,793	29,814	29,815	33,969
SET2_24	65,891	65,891	65,984	65,891	65,891	65,891	65,891	65,891	68,727
SET2_25	75,627	75,628	75,745	75,627	75,627	75,627	75,705	75,705	78,266
SET2_26	60,952	61,002	61,173	60,977	60,952	60,952	60,988	60,988	63,558
SET2_27	53,016	53,016	53,052	53,016	53,016	53,016	53,016	53,441	54,797
SET2_28	44,545	44,552	44,705	44,549	44,545	44,545	44,923	44,923	46,733
SET2_29	93,631	93,638	93,659	93,631	93,631	93,631	93,632	93,632	96,281
SET2_30	68,324	68,333	68,573	68,573	68,324	68,324	68,324	68,324	71,919

Table 12 Multi-LSB Detailed Computational Results, SET3 instances 1-20

	Lower Bounds				**LR1 (Cap)**		**LR2 (Lev)**		Best
	ℓ, S	MC	FL	Heuristic	LB	UB	LB	UB	Soln
SET3_01	65,668	71,594	71,584	71,533	66,984	66,984	65,761	112,652	209,129
SET3_02	82,342	89,855	89,887	89,980	84,865	84,865	82,704	105,740	243,511
SET3_03	74,209	82,398	82,440	81,340	77,086	77,086	74,611	99,483	235,198
SET3_04	78,282	85,258	85,229	86,280	80,716	80,716	78,436	108,664	240,339
SET3_05	76,607	83,692	83,667	84,430	78,931	78,931	76,884	102,852	227,758
SET3_06	79,093	88,689	88,737	85,674	82,910	82,910	79,625	112,534	235,642
SET3_07	72,979	79,067	79,181	79,668	75,365	75,365	73,098	105,466	237,218
SET3_08	88,610	94,504	94,481	98,469	92,108	92,108	89,213	129,505	251,628
SET3_09	64,180	67,768	67,760	73,019	64,336	64,336	64,180	85,114	216,025
SET3_10	66,878	74,333	74,324	73,902	67,928	67,928	66,912	92,540	229,242
SET3_11	42,946	46,063	45,997	47,273	43,902	43,902	43,012	69,501	152,962
SET3_12	86,047	95,953	95,980	97,672	90,412	90,412	87,641	112,402	217,497
SET3_13	74,643	81,477	81,348	83,699	75,379	75,379	74,987	102,771	224,670
SET3_14	85,209	91,252	91,435	94,426	86,813	86,813	85,493	102,438	225,657
SET3_15	40,715	43,551	43,343	45,265	40,843	40,843	40,750	74,085	167,494
SET3_16	46,548	50,868	50,784	51,811	48,528	48,528	48,360	62,509	162,616
SET3_17	71,555	78,132	77,988	82,199	72,458	72,458	71,837	95,764	212,399
SET3_18	39,533	40,406	40,259	46,743	39,658	39,658	39,616	57,199	112,468
SET3_19	47,495	50,636	50,497	53,815	48,266	48,266	47,636	84,711	154,981
SET3_20	58,189	60,240	60,125	62,614	58,529	58,529	59,753	95,852	191,639

Table 13 Multi-LSB Detailed Computational Results, SET3 instances 21-30 and SET4 instances 1-10

	Lower Bounds				LR1 (Cap)		LR2 (Lev)		Best
	ℓ, S	MC	FL	Heuristic	LB	UB	LB	UB	Soln
SET3_21	44,182	45,435	45,383	53,138	44,359	44,359	44,182	60,262	150,758
SET3_22	130,235	138,607	138,279	136,582	133,995	133,995	130,930	142,716	292,199
SET3_23	96,810	102,993	102,912	107,981	99,719	99,719	96,939	115,205	240,643
SET3_24	105,300	110,117	109,994	115,086	105,327	105,327	105,300	136,353	292,996
SET3_25	203,044	210,031	209,928	210,037	204,955	204,955	203,044	212,110	349,975
SET3_26	145,184	152,864	152,545	160,639	146,938	146,938	145,198	155,347	323,870
SET3_27	145,420	154,121	153,805	154,499	148,698	148,698	145,674	169,988	343,486
SET3_28	145,227	153,083	153,327	152,942	147,940	147,940	145,927	162,729	254,008
SET3_29	79,813	87,043	86,551	84,552	81,494	81,494	80,206	96,912	207,127
SET3_30	274,018	283,252	282,958	275,167	276,810	276,810	274,018	284,338	431,136
SET4_01	16,353	16,532	18,093	21,961	16,353	16,353	16,951	23,694	58,720
SET4_02	31,541	32,773	34,074	41,393	31,541	31,541	31,726	33,919	82,496
SET4_03	24,864	25,616	27,464	33,058	24,864	24,864	24,864	28,061	73,740
SET4_04	27,786	28,837	30,023	36,512	27,786	27,786	27,928	31,426	73,651
SET4_05	25,450	26,353	27,335	35,022	25,450	25,450	25,450	29,755	67,874
SET4_06	30,632	31,495	32,990	40,513	30,632	30,632	31,054	35,402	79,781
SET4_07	22,650	23,189	24,599	31,952	22,650	22,650	23,884	30,365	65,736
SET4_08	40,532	42,512	43,131	48,381	40,532	40,532	40,538	41,812	88,388
SET4_09	13,490	13,557	14,687	21,182	13,490	13,490	14,650	19,585	57,070
SET4_10	15,542	15,553	16,857	25,595	15,542	15,542	16,041	26,902	59,319

Table 14 Multi-LSB Detailed Computational Results, SET4 instances 11-30

	Lower Bounds				LR1 (Cap)		LR2 (Lev)		Best
	ℓ, S	MC	FL	Heuristic	LB	UB	LB	UB	Soln
SET4_11	12,802	12,996	13,825	17,303	12,802	12,802	13,675	15,205	28,989
SET4_12	43,341	44,527	45,100	50,868	43,341	43,341	44,523	46,502	78,062
SET4_13	28,152	28,736	30,049	34,945	28,152	28,152	28,152	33,352	53,833
SET4_14	56,174	57,052	57,302	64,255	56,174	56,174	56,406	57,049	82,406
SET4_15	14,628	14,715	15,304	15,863	14,628	14,628	15,244	16,260	26,980
SET4_16	17,171	17,529	17,990	22,405	17,172	17,172	17,662	19,874	35,280
SET4_17	29,001	29,886	30,581	36,480	29,225	29,225	29,237	31,729	54,515
SET4_18	19,184	19,213	19,309	22,584	19,185	19,185	19,705	19,997	26,279
SET4_19	10,724	10,769	11,780	14,950	10,724	10,724	12,581	15,411	31,974
SET4_20	18,718	18,858	19,702	23,969	18,731	18,731	19,420	21,014	39,983
SET4_21	15,812	16,243	16,819	18,259	15,812	15,812	16,386	17,720	25,899
SET4_22	91,715	93,010	93,185	93,869	91,733	91,733	92,228	92,310	120,166
SET4_23	55,058	55,601	56,077	57,298	55,151	55,151	55,562	56,132	76,857
SET4_24	58,919	59,231	59,512	63,700	58,919	58,919	59,213	60,947	85,119
SET4_25	171,987	172,779	172,904	173,663	171,987	171,987	171,987	171,988	201,717
SET4_26	110,570	111,393	111,703	117,746	110,570	110,570	110,570	110,577	142,090
SET4_27	101,114	102,197	102,182	103,873	101,471	101,471	101,267	101,340	139,874
SET4_28	112,892	113,353	114,022	113,987	112,892	112,892	112,987	112,987	126,027
SET4_29	51,149	51,394	51,776	56,304	51,149	51,149	51,253	51,253	68,320
SET4_30	241,678	243,702	243,998	242,481	241,801	241,801	241,678	241,693	267,976