CORE DISCUSSION PAPER
2001/8

Discrete Lot–Sizing and Convex Integer Programming

Andrew J. Miller [1]
Laurence A. Wolsey [2]

February 2001

Abstract

We study the polyhedral structure of variants of the discrete lot–sizing problem viewed as special cases of convex integer programs. Our approach in studying convex integer programs is to develop results for simple mixed integer sets that can be used to model integer convex objective functions. These results allow us to define integral linear programming formulations for the discrete lot–sizing problem in which backlogging and/or safety stocks are present, and to give extended formulations for other cases. Our results help significantly to solve test cases arising from an industrial application motivating this research.

[1] CORE, Université Catholique de Louvain, Belgium
[2] CORE and INMA, Université Catholique de Louvain, Belgium

This paper presents research results of the Belgian Program on Interuniversity Poles of Attraction initiated by the Belgian State, Prime Minister's Office, Science Policy Programming. The scientific responsibility is assumed by the authors.
This research was also supported by the European Commission GROWTH Programme, Research Project LISCOS, Large Scale Integrated Supply Chain Optimization Software Based on Branch–and–Cut and Constraint Programming Methods, Contract No. GRDI–1999–10056.

1 Introduction

This article is motivated by a production planning problem from a well-known multinational company encountered recently. The problem in question is a multi–item discrete lot–sizing problem with backlogging and with additional complicating constraints, so finding tight formulations for different variants of the *discrete lot–sizing problem* is a principal topic of this paper. The second topic arose from the observation that the different variants can also be viewed as *convex integer programs* ($CIPs$) with simple piecewise affine objective functions and simple linear constraints. This second viewpoint leads us to study the convex hulls of some simple mixed integer programs.

Below we outline the contents of the paper, and discuss some of the relevant literature. In Section 2 we first motivate our results by presenting formulations of the discrete lot–sizing problem variants that we study and indicating how the variants can be formulated as $CIPs$.

In Section 3 we look at convex integer programs, and in particular two functions: the function $g(z) = \max\{0, \max_{k=1,\dots,K}(a_k z - c_k)\}$ with z integer arising in separable objective functions, and the function $g(z) = \max\{0, \max_{k=1,\dots,K}(b_k - z_k)\}$ with z a K-dimensional integer vector. We develop polynomial algorithms to solve unconstrained instances of CIP with these functions, and we also discuss under what conditions the approach can be extended to solve constrained instances of CIP with these objective functions.

In Chapter 13 of Ahuja et al. [1993], it is shown that it is possible to solve $CIPs$ of the form $\min\{g(z) : Az \geq b, z \in \mathbb{Z}^n_+\}$, when $g(z) = \sum_j g_j(z_j)$ is separable and A is a network flow matrix, by breaking up the flow z_j on arc j into separate flows z_{kj} for each cost segment of the integer closure $\bar{g}_j$, and then solving by a standard network flow algorithm. This approach extends to the case where A is totally unimodular (TU), since now it suffices to duplicate columns in the matrix. In Hochbaum and Shantikumar [1990] a polynomial algorithm is given for CIP where the objective g is separable, each function of a single variable g_j is an arbitrary (not necessarily piecewise linear) convex function, and A is TU. The algorithm is based on solving successive linear programs, and it depends on sophisticated proximity results. For cases of multi–variate convex integer functions, we are not aware of research that discusses the complexity of solving CIP.

Our approach is to seek to reformulate CIP as a linear program. This leads immediately to a study of the convex hulls of the sets $X_K = \{(\phi, z) \in \mathbb{R}^K_+ \times \mathbb{Z}^1 : \phi_k + z \geq b_k$ for $k = 1, \dots, K\}$, and $Y = \{(\phi, z) \in \mathbb{R}^1_+ \times \mathbb{Z}^K : \phi + z_k \geq b_k$ for $k = 1, \dots, K\}$. The set X_K with $K = 1$ is the basic mixed integer set used to derive the mixed integer rounding (MIR) inequality Nemhauser and Wolsey [1988], so X_K is a natural generalization. In contrast to the approach in Ahuja et al. [1993], we start from the functions $g_j(z_j)$, and provide compact linear programming formulations that generate the integer closures $\bar{g}_j$, and that also solve CIP when the constraint matrix is TU.

For the set Y, Günlük and Pochet [1998] have shown that the mixing procedure for MIR inequalities suffices to give a complete description of the convex hull. Here we show also that there is a compact linear programming formulation solving CIP when the constraints are the dual of a network flow matrix. A compact formulation is also given for a generalization of the set Y of the form $\{(\phi, r, z) \in \mathbb{R}^1_+ \times \mathbb{R}^K_+ \times \mathbb{Z}^K : \phi + r_k + z_k \geq b_k$ for $k = 1, \ldots, K\}$.

In Section 4 we apply our reformulation results for CIP to the discrete lot–sizing problem. Results in the literature about this problem are scarce. For the discrete multi–item constant capacity problem without backlogging, there exists a formulation as a minimum cost network flow problem, so the problem is well-solved. Discrete lot–sizing problems with startup costs and/or times have received considerable attention in the literature (e.g. Fleischmann [1990, 1994], van Hoesel et al. [1994], van Eijl [1996], van Eijl and van Hoesel [1997]), and in general they are more difficult than problems without startup costs. For example, the constant capacity, multi–item discrete lot–sizing problem with start–up costs is $\mathcal{NP}$–hard. With a single item, it is polynomially solvable, but a characterization of the convex hull by inequalities is not known.

We consider three variants of the basic problem involving backlogging and/or initial stock variables. We first use our results for X_K and related sets to show how to formulate the multi–item discrete lot–sizing problem with backlogging so that the resulting LP has integral solutions. Then we consider a variant with initial stocks but without backlogging. Here known results for CIP yield a polynomial reformulation that is tight in the single item case. Finally we apply our results for CIP to derive a compact formulation for a variant with both backlogging and initial stocks.

In Section 4 we present computational results for the practical instance, adapting it in turn to each of the variants studied.

2 Motivation

In the discrete lot–sizing problems we will analyze, there are NI items and NT time periods. Demand for item i in period t is given by d^i_t. If item i is produced in a period, C^i is the amount that must be produced. The parameters h^i_t, b^i_t, and q^i_t are the holding, backlogging, and fixed set-up costs respectively. We assume throughout that all these costs are nonnegative. We then introduce s^i_t, r^i_t, and y^i_t to be the inventory, backlogging, and set-up variables respectively.

The general Discrete Lot–sizing Problem can now be formulated as

$$\min \sum_{i=1}^{NI} \sum_{t=0}^{NT} h^i_t s^i_t + \sum_{i=1}^{NI} \sum_{t=1}^{NT} b^i_t r^i_t + \sum_{i=1}^{NI} \sum_{t=1}^{NT} q^i_t y^i_t \tag{1}$$

$$s^i_{t-1} + r^i_t + C^i y^i_t = d^i_t + s^i_t + r^i_{t-1} \text{ for } i = 1, ..., NI, t = 1, ..., NT \tag{2}$$

$$\sum_{i=1}^{NI} y^i_t \leq 1 \text{ for } t = 1, ..., NT \tag{3}$$

$$s_t^i \geq 0 \text{ for } i = 1, ..., NI, t = 0, ..., NT \tag{4}$$

$$r_t^i \geq 0 \text{ for } i = 1, ..., NI, t = 1, ..., NT \tag{5}$$

$$y_t^i \in \{0, 1\} \text{ for } i = 1, ..., NI, t = 1, ..., NT. \tag{6}$$

Equations (2) are the product conservation constraints, and inequality (3) ensures that at most one item is produced (and thus one set-up takes place) in each period. Note that variables s_0^i allow the possibility of buying in an initial stock of each product. Throughout we assume $r_0^i = 0$ for $i = 1, ..., NI$. We will use the notation: $d_{kt}^i \equiv \sum_{u=k}^t d_u^i$ for $1 \leq k \leq t \leq NT$, and $i = 1, \ldots, NI$.

When initial inventory variables s_0^i and backorder variables r_t^i are not present, (1)–(6) becomes the most basic discrete lot–sizing problem DLS. The problem with backlogging but no initial inventory variables is called $DLSB$, the problem with initial inventory but no backlogging is $DLSI$, and the model with backlogging and initial inventory variables is $DLSBI$.

We now show how two of the variants can be viewed as instances of CIP.

2.1 Viewing DLS with Backlogging as a CIP

For $DLSB$ (recall $s_0^i = 0$, for $i = 1, ..., NI$), we can aggregate the constraints (2) giving

$$C^i \sum_{u=1}^t y_u^i = d_{1t}^i + s_t^i - r_t^i.$$

This leads to an equivalent formulation

$$\min \sum_{i=1}^{NI} \sum_{t=1}^{NT} \max\{h_t^i(C^i \sum_{u=1}^t y_u^i - d_{1t}^i), -b_t^i(C^i \sum_{u=1}^t y_u^i - d_{1t}^i)\} + \sum_{i=1}^{NI} \sum_{t=1}^{NT} q_t^i y_t^i \tag{7}$$

$$\sum_{i=1}^{NI} y_t^i \leq 1 \text{ for } t = 1, ..., NT, \tag{8}$$

$$y_t^i \in \{0, 1\} \text{ for } i = 1, ..., NI, t = 1, ..., NT. \tag{9}$$

Defining variables $z_t^i = \sum_{u=1}^t y_u^i$, we can rewrite (7)–(9) as

$$\min \sum_{i=1}^{NI} \sum_{t=1}^{NT} \max\{h_t^i(C^i z_t^i - d_{1t}^i), -b_t^i(C^i z_t^i - d_{1t}^i)\} + \sum_{i=1}^{NI} \sum_{t=1}^{NT} q_t^i y_t^i \tag{10}$$

$$z_t^i = \sum_{u=1}^t y_u^i \text{ for } i = 1, ..., NI, t = 1, ..., NT \tag{11}$$

$$\sum_{i=1}^{NI} y_t^i \leq 1 \text{ for } t = 1, ..., NT \tag{12}$$

$$y_t^i \in \{0, 1\} \text{ for } i = 1, ..., NI, t = 1, ..., NT. \tag{13}$$

Now the function

$$H_t^i(v) = \max\{h_t^i(C^i v - d_{1t}^i), -b_t^i(C^i v - d_{1t}^i)\}$$

is convex for $i = 1, ..., NI$ and $t = 1, ..., NT$, and the constraint set (11)–(13) can be viewed as a flow in the network shown in Figure 1.

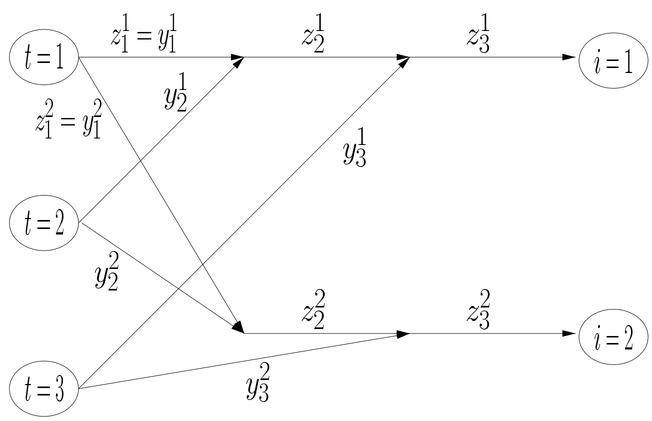

Figure 1: Network representation of DLS

So $DLSB$ has been reformulated as a problem of minimizing a sum of separable, piecewise linear convex functions, subject to network flow and integrality constraints. The reformulation of DLS as a CIP is similar.

2.2 Viewing DLS with Initial Stocks as a CIP

For $DLSI$, aggregating the constraints (2) gives

$$s_0^i + C^i \sum_{u=1}^{t} y_u^i = d_{1t}^i + s_t^i.$$

Now eliminating the variables s_t^i for $i = 1, \ldots, NI$ and $t = 1, \ldots, NT$ leaves the constraints $s_0^i \geq 0$ and

$$s_0^i + C^i \sum_{u=1}^{t} y_u^i \geq d_{1t}^i.$$

Now assuming that the optimal value is bounded, there exists an optimal solution with

$$s_0^i = \max\{0, \max_{t=1,\dots,NT}(d_{1t}^i - C^i \sum_{u=1}^{t} y_u^i)\} \text{ for } i = 1,\dots,NI.$$

Again letting $z_t^i = \sum_{u=1}^t y_u^i$, we can write $DLSBI$ as

$$\begin{gathered}
\min \sum_{i=1}^{NI} h_0^i \max\{0, \max_t(d_{1t}^i - C^i z_t^i)\} + \sum_{i=1}^{NI} \sum_{t=1}^{NT} q_t^i y_t^i \\
z_t^i = \sum_{u=1}^t y_u^i \text{ for } i = 1,\dots,NI, t = 1,\dots,NT \\
\sum_{i=1}^{NI} y_t^i \leq 1 \text{ for } t = 1,\dots,NT \\
y_t^i \in \{0,1\} \text{ for } i = 1,\dots,NI, t = 1,\dots,NT.
\end{gathered}$$

Now the functions

$$g^i(z_1^i,\dots,z_{NT}^i) = h_0^i \max\{0, \max_t(d_{1t}^i - C^i z_t^i)\}$$

appearing in the objective are convex and piecewise affine for each i, so this is an instance of CIP with the same network constraints as in $DLSB$. The reformulation of $DLSBI$ as a CIP is similar.

3 Integer Programs with Convex Objective Functions

We consider convex integer programs of the form

$$(CIP) \qquad \min\{g(z) : Az \geq b, z \in \mathbb{Z}^n\} \tag{14}$$

where g is convex and piecewise affine. A natural question to ask is for which convex functions g and constraints (A,b), the convex integer program CIP is polynomial. Here we have the more specific goal of reformulating CIP as a compact linear program. We are therefore interested in the sets

$$\begin{gathered}
\mathrm{Epi}(g) = \{(\phi, z) \in \mathbb{R}^1 \times \mathbb{R}^n : \phi \geq g(z)\}, \text{ the epigraph of } g, \\
Y(g) = \{(\phi, z) \in \mathbb{R}^1 \times \mathbb{Z}^n : \phi \geq g(z)\}, \text{ and} \\
\mathrm{conv}(Y(g)).
\end{gathered}$$

Note that $\mathrm{conv}(Y(g))$ is in turn the epigraph of a function $\bar{g}$, called the *integer closure* of g. We will study three specific functions: first those of the form $g(z) = \max\{0, \max_k(a_k z - c_k)\}$ where $z \in \mathbb{Z}^1$, secondly $g(z) = \max\{0, \max_k(b_k - z_k)\}$ where $z \in \mathbb{Z}^K$, and finally a generalization of the latter involving an additional K continuous variables.

3.1 Functions of a Single Variable - Separable Functions

Suppose that we have the function $h(z) = \max_{k=0,1,\dots,K}(a'_k z - c'_k)$ where each pair (a'_k, c'_k) is necessary in the description of h, and $a'_0 < a'_1 < \dots < a'_K$. Now setting $a_k = a'_k - a'_0, c_k = c'_k - c'_0$ for $k = 0, \dots, K$, we can write $h(z) = a'_0 z - c'_0 + g(z)$ where

$$g(z) = \max\{0, a_1 z - c_1, \dots, a_K z - c_K\}$$

with $0 = a_0 < a_1 < \dots < a_K$. We first study ways to model g and its epigraph, and then how to model its integer closure $\bar{g}$.

Observation 1. The breakpoints $b_k = \frac{c_k - c_{k-1}}{a_k - a_{k-1}}$ for $k = 1, \dots, K$ satisfy $b_1 < \dots < b_K$ as each pair (a'_k, c'_k) is necessary in the description of $g(z)$.

Observation 2. $a_k z - c_k = (a_{k-1} z - c_{k-1}) + (a_k - a_{k-1})(z - b_k) = \sum_{i=1}^{k}(a_i - a_{i-1})(z - b_i)$, $k = 1, ..., K$. Let $b_0 = -\infty$ and $b_{K+1} = \infty$. Then $a_k z - c_k = \max\{0, \max_{i=1,\dots,K}(a_i z - c_i)\}$ if and only if $b_k \le z \le b_{k+1}$, for $k = 0, ..., K$.

Observation 3. For any z in the domain of g,

$$\begin{aligned} g(z) &= \textstyle\sum_{k=1}^{K}(a_k - a_{k-1})(z - b_k)^+ \\ &= \min\{\textstyle\sum_{k=1}^{K}(a_k - a_{k-1})\sigma_k : \sigma_k \ge z - b_k, \sigma_k \ge 0, \text{ for } k = 1, \dots, K\} \\ &= \max\{\textstyle\sum_{k=1}^{K} u_k(z - b_k) : 0 \le u_k \le a_k - a_{k-1} \text{ for } k = 1, \dots, K\}. \end{aligned}$$

Now we add the restriction that z be integer, and show how to calculate and model $\bar{g}$. We will derive three formulations, two of them involving additional variables.

Note that in fact some of the linear segments defining g are not needed in the description of $\bar{g}$. In particular only the segments k in which the interval $[b_k, b_{k+1})$ contains an integer point are necessary. Below we will initially assume that the redundant segments have been removed, and thus $\lfloor b_k \rfloor < \lfloor b_{k+1} \rfloor$ for all k.

The basic integrality result we need involves a very simple mixed integer set

$$X_K = \{(\sigma, z) \in \mathbb{R}_+^K \times \mathbb{Z} : \sigma_k \ge z - b_k \text{ for } k = 1, \dots, K\}.$$

Let $f_k = b_k - \lfloor b_k \rfloor$ for $k = 1, ..., K$. We now show that it suffices to add the K inequalities $\sigma_k \ge (1 - f_k)(z - \lfloor b_k \rfloor)$, known as mixed integer rounding (MIR) inequalities (Nemhauser and Wolsey [1988]), to obtain the convex hull.

The proof is based on a simple Lemma.

Lemma 1 *For $z \in \mathbb{R}^1$, let $\sigma_k = \max\{0, z - b_k, (1 - f_k)(z - \lfloor b_k \rfloor)\}$.*
a) If $b_k \notin \mathbb{Z}^1$,

i) $\sigma_k = z - b_k$ *if and only if* $z \geq \lceil b_k \rceil$,
ii) $\sigma_k = 0$ *if and only if* $z \leq \lfloor b_k \rfloor$,
iii) $\sigma_k = (1 - f_k)(z - \lfloor b_k \rfloor)$ *if and only if* $\lfloor b_k \rfloor \leq z \leq \lceil b_k \rceil$.
b) If $b_k \in \mathbb{Z}^1$,
i) $\sigma_k = z - b_k$ *if and only if* $z \geq \lfloor b_k \rfloor$,
ii) $\sigma_k = 0$ *if and only if* $z \leq \lfloor b_k \rfloor$.

Proof. Case a). $\sigma_k = z - b_k$ holds if and only if $z - b_k \geq (1 - f_k)(z - \lfloor b_k \rfloor)$ and $z - b_k \geq 0$. But

$$
\begin{aligned}
z - b_k \geq (1 - f_k)(z - \lfloor b_k \rfloor) \iff f_k z \geq b_k - (1 - f_k)\lfloor b_k \rfloor \\
\iff f_k z \geq f_k(1 + \lfloor b_k \rfloor) \iff z \geq \lceil b_k \rceil.
\end{aligned}
$$

Similarly, $\sigma_k = 0$ holds if and only if $0 \geq (1 - f_k)(z - \lfloor b_k \rfloor)$ and $0 \geq z - b_k$. The first inequality holds if and only if $z \leq \lfloor b_k \rfloor$. Finally the last possibility, $\sigma_k = (1 - f_k)(z - \lfloor b_k \rfloor)$, holds if and only if $\lfloor b_k \rfloor \leq z \leq \lceil b_k \rceil$. Case b) is similar. □

Theorem 2 *conv*(X_K) *is described by the inequalities*

$$
\begin{aligned}
\sigma_k \geq z - b_k \text{ for } k = 1, \ldots, K \\
\sigma_k \geq (1 - f_k)(z - \lfloor b_k \rfloor) \text{ for } k = 1, \ldots, K \\
\sigma_k \geq 0 \text{ for } k = 1, \ldots, K.
\end{aligned}
$$

Proof. Let T be the polyhedron described by these inequalities. Clearly $\text{conv}(X_K) \subseteq T$. Consider now a bounded face of T of maximum dimension. As each σ_k must be minimal on such a face, $\sigma_k = \max\{0, z - b_k, (1 - f_k)(z - \lfloor b_k \rfloor)\}$. From the previous Lemma, it follows that each such face is of the form $l \leq z \leq u$ with $l, u \in \mathbb{Z}^1$ or infinite. Thus T has extreme points with z integral, and the claim follows. □

Now we present three formulations for $\text{conv}(Y(g))$, keeping in mind that $0 < a_1 < \ldots < a_K$ and $b_1 < \ldots < b_K$. Consider the polyhedron P:

$$\phi \geq \sum_{k=1}^{K} a_k s_k \tag{15}$$
$$0 \leq s_k \leq b_{k+1} - b_k \text{ for } k = 1, \ldots, K - 1 \tag{16}$$
$$\sum_{k=1}^{K} s_k \geq z - b_1 \tag{17}$$
$$\sum_{i=k}^{K} s_i \geq (1 - f_k)(z - \lfloor b_k \rfloor) \text{ for } k = 1, \ldots, K, \tag{18}$$

the polyhedron Q:

$$\phi \geq \sum_{k=1}^{K} (a_k - a_{k-1})\sigma_k \tag{19}$$
$$\sigma_k \geq z - b_k \text{ for } k = 1, \ldots, K \tag{20}$$
$$\sigma_k \geq (1 - f_k)(z - \lfloor b_k \rfloor) \text{ for } k = 1, \ldots, K \tag{21}$$
$$\sigma_k \geq 0 \text{ for } k = 1, \ldots, K, \tag{22}$$

and the polyhedron R:

$$\phi \geq a_k z - c_k \text{ for } k = 1, \dots, K \tag{23}$$

$$\phi \geq a_{k-1} z - c_{k-1} + (a_k - a_{k-1})(1 - f_k)(z - \lfloor b_k \rfloor) \text{ for } k = 1, \dots, K \tag{24}$$

$$\phi \geq 0. \tag{25}$$

Before stating the main result, we derive a result for polyhedron R similar to that of Lemma 1.

Lemma 3 *For $z \in \mathbb{R}^1$, let*

$$\phi = \min\{0, \min_k (a_k z - c_k), \min_k (a_{k-1} z - c_{k-1} + (1 - f_k)(a_k - a_{k-1})(z - \lfloor b_k \rfloor))\}.$$

a) If $b_k \notin \mathbb{Z}^1$, for each $k = 1, ..., K$,
i) $\phi = a_k z - c_k$ if and only if $\lceil b_k \rceil \leq z \leq \lfloor b_{k+1} \rfloor$, for each $k = 1, ..., K$,
ii) $\phi = a_{k-1} z - c_{k-1} + (a_k - a_{k-1})(1 - f_k)(z - \lfloor b_k \rfloor)$ if and only if $\lfloor b_k \rfloor \leq z \leq \lceil b_k \rceil$, for each $k = 1, ..., K$, and
iii) $\phi = 0$ if and only if $z \leq \lfloor b_1 \rfloor$.
b) If $b_k \in \mathbb{Z}^1$, for each $k = 1, ..., K$,
i) $\phi = a_k z - c_k$ if and only if $\lfloor b_k \rfloor \leq z \leq \lfloor b_{k+1} \rfloor$, for each $k = 1, ..., K$, and
ii) $\phi = 0$ if and only if $z \leq \lfloor b_1 \rfloor$.

Proof. a) i). For each $k = 1, ..., K$,

$$a_k z - c_k \geq a_{k-1} z - c_{k-1} + (a_k - a_{k-1})(1 - f_k)(z - \lfloor b_k \rfloor)$$

if and only if

$$(a_k - a_{k-1})z - (c_k - c_{k-1}) \geq (1 - f_k)((a_k - a_{k-1})z - (a_k - a_{k-1})\lfloor b_k \rfloor),$$

which, dividing by $(a_k - a_{k-1})$ and using the definition of b_k, holds if and only if

$$z - b_k \geq (1 - f_k)(z - \lfloor b_k \rfloor),$$

which holds if and only if $z \geq \lceil b_k \rceil$.

Similarly

$$a_k z - c_k \geq a_k z - c_k + (a_{k+1} - a_k)(1 - f_{k+1})(z - \lfloor b_{k+1} \rfloor).$$

holds if and only if $z \leq \lfloor b_{k+1} \rfloor$. Combined with Observation 2, the claim follows. The arguments for ii) and iii), and for b), are similar. □

Now we can state the main result of this section.

Theorem 4 *conv*$(Y(g))$ =*proj*$_{\phi,z}(P)$ =*proj*$_{\phi,z}(Q) = R$.

Proof. It follows from Theorem 2 that $\text{proj}_{\phi,z}(Q)$ is integral in z. Combined with Observation 3, this shows that $\text{proj}_{\phi,z}(Q) = \text{conv}(Y(g))$. It is also easily seen that $\text{proj}_{\phi,z}(P) \subseteq \text{proj}_{\phi,z}(Q) \subseteq R$, as every inequality describing Q can be obtained from those of P by taking $\sigma_k = \sum_{j=k}^{K} s_j$, and every inequality describing R is valid for Q.

We now show that $R \subseteq \text{proj}_{\phi,z}(Q)$. Given $(\phi, z) \in R$, let

$$\phi^* = \max[0, \max_k(a_k z - c_k), \max_k(a_{k-1}z - c_{k-1} + (a_k - a_{k-1})(1 - f_k)(z - \lfloor b_k \rfloor))].$$

Case 1. If $\phi^* = a_k z - b_k$ for some k, $\lceil b_k \rceil \leq z \leq \lfloor b_{k+1} \rfloor$ from Lemma 3. Set $\sigma_i = (z - b_i)^+$ for $i = 1, \ldots, K$, so (20) and (22) are satisfied. Also for $i \leq k$, $z \geq \lfloor b_k \rfloor + 1 \geq \lfloor b_i \rfloor + 1$ which implies $\sigma_i \geq (1 - f_i)(z - \lfloor b_i \rfloor)$, and for $i > k$, $z \leq \lfloor b_{k+1} \rfloor \leq \lfloor b_{i+1} \rfloor$ which implies that $\sigma_i = 0 \geq (1 - f_i)(z - \lfloor b_i \rfloor)$, so (21) holds. Finally

$$\sum_{i=1}^{K}(a_i - a_{i-1})\sigma_i = \sum_{i=1}^{k}(a_i - a_{i-1})(z - b_i) = a_k z - c_k = \phi^* \leq \phi$$

and (19) holds.

Case 2. If $\phi^* = a_{k-1}z - c_{k-1} + (a_k - a_{k-1})(1 - f_k)(z - \lfloor b_k \rfloor)$ for some k, $\lfloor b_k \rfloor \leq z \leq \lceil b_k \rceil$ by Lemma 3. Set $\sigma_i = z - b_i$ for $i \leq k - 1, \sigma_k = (1 - f_k)(z - \lfloor b_k \rfloor)$ and $\sigma_i = 0$ for $i > k$. Again for $i \leq k-1$, $z \geq \lfloor b_k \rfloor \geq \lfloor b_i \rfloor + 1$ implies $\sigma_i = z - b_i \geq (1 - f_i)(z - \lfloor b_i \rfloor)$. For $i = k$, $\lfloor b_k \rfloor \leq z_k \leq \lceil b_k \rceil$ implies $\sigma_i = (1 - f_i)(z - \lfloor b_i \rfloor) \geq z - b_i$, and for $i > k$, $\sigma_i = 0 \geq z - b_i$ and $\sigma_i = 0 \geq (1 - f_i)(z - \lfloor b_i \rfloor)$. Finally $\sum_{i=1}^{K}(a_i - a_{i-1})\sigma_i = \phi^* \leq \phi$.

So we have shown that $R \subseteq \text{proj}_{\phi,z}(Q)$. The argument that $\text{proj}_{\phi,z}(Q) \subseteq \text{proj}_{\phi,z}(P)$ is similar. It suffices to take $s_K = \sigma_K$ and $s_i = \sigma_i - \sigma_{i+1}$ for $i = 1, \ldots, K - 1$. □

The following example illustrates the results shown to this point. This example will also be used later to illustrate some of our results for discrete lot–sizing problems.

Example 1

Let h be defined by

$$h(z) = \max\{-1z + 2, -\frac{1}{4}z + \frac{7}{8}, \frac{1}{8}z - \frac{7}{16}\}.$$

It is easily checked that the slopes are nondecreasing, and the description of h is nonredundant. Now the function h can be rewritten as $h(z) = -1z + 2 + g(z)$, where

$$g(z) = \max\{0, \frac{3}{4}z - \frac{9}{8}, \frac{9}{8}z - \frac{39}{16}\} = \max\{0, \frac{3}{4}(z - \frac{3}{2}), \frac{3}{4}(z - \frac{3}{2}) + \frac{3}{8}(z - \frac{7}{2})\}.$$

with breakpoints $b_1 = \frac{3}{2}$, and $b_2 = \frac{7}{2}$. Here

$$\text{Epi}(g) = \{(\phi, z) \in \mathbb{R}^1_+ \times \mathbb{R}^1 : \phi \geq \frac{3}{4}z - \frac{9}{8}, \phi \geq \frac{9}{8}z - \frac{39}{16}\},$$

$$Y(g) = \{(\phi, z) \in \mathbb{R}^1_+ \times \mathbb{Z}^1 : \phi \geq \frac{3}{4}z - \frac{9}{8}, \phi \geq \frac{9}{8}z - \frac{39}{16}\},$$

and, by Theorem 4,

$$\text{conv}(Y(g)) = R = \{(\phi, z) \in \mathbb{R}^1_+ \times \mathbb{R}^1 : \phi \geq \frac{3}{8}z - \frac{3}{8}, \phi \geq \frac{3}{4}z - \frac{9}{8}, \phi \geq \frac{15}{16}z - \frac{27}{16}, \phi \geq \frac{9}{8}z - \frac{39}{16}\}.$$

whereas P is the polyhedron

$$\begin{gathered} \phi \geq \tfrac{3}{4}s_1 + \tfrac{9}{8}s_2 \\ 0 \leq s_1 \leq 2, s_2 \geq 0 \\ s_1 + s_2 \geq z - \tfrac{3}{2} \\ s_1 + s_2 \geq \tfrac{1}{2}(z - 1) \\ s_2 \geq \tfrac{1}{2}(z - 3), \end{gathered}$$

and Q is the polyhedron

$$\begin{gathered} \phi \geq \tfrac{3}{4}\sigma_1 + \tfrac{3}{8}\sigma_2 \\ \sigma_1 \geq z - \tfrac{3}{2} \\ \sigma_2 \geq z - \tfrac{7}{2} \\ \sigma_1 \geq \tfrac{1}{2}(z - 1) \\ \sigma_2 \geq \tfrac{1}{2}(z - 3) \\ \sigma_1 \geq 0, \sigma_2 \geq 0. \end{gathered}$$

From Theorem 4, we know that $\text{proj}_{(\phi,z)}P = \text{proj}_{(\phi,z)}Q = R = \text{conv}(Y(g))$.

In Figure 2 the epigraphs of g and its closure $\bar{g}$ are shown. The inequalities (24) are indicated as dashed lines, and the part of $\text{Epi}(g)$ cut off by these inequalities is in black. $\square$

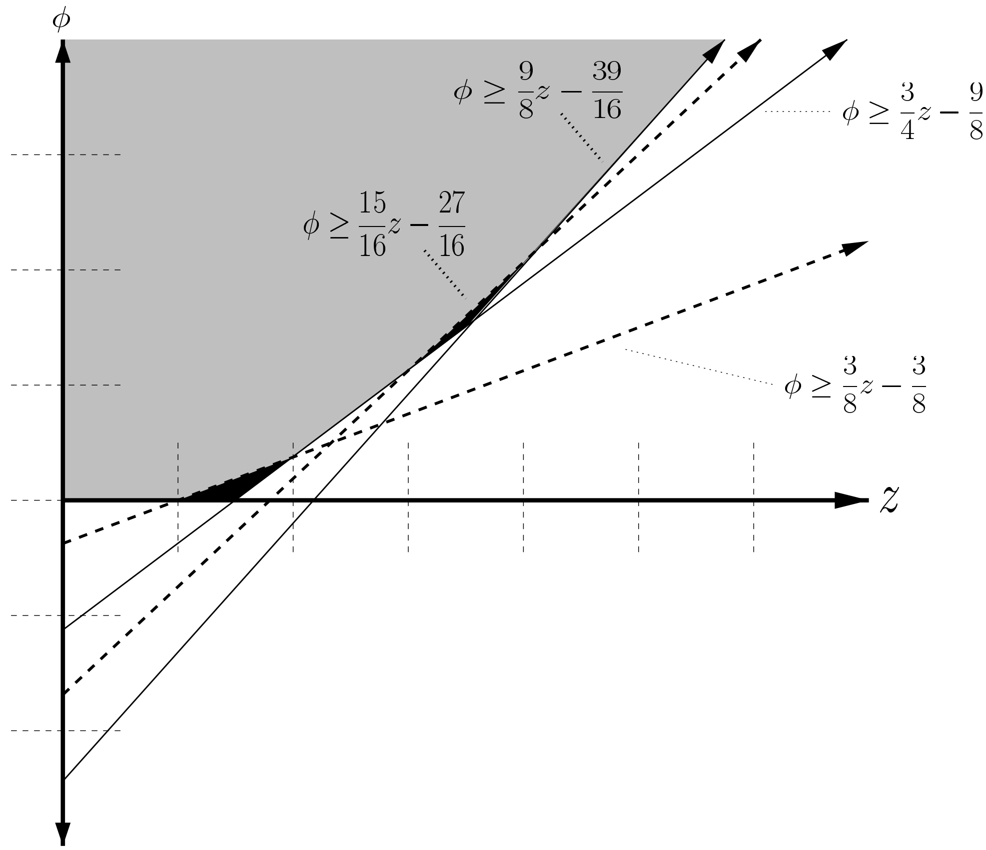

Figure 2: The epigraphs of a convex function and its closure

When we drop the assumption that $\lfloor b_k \rfloor < \lfloor b_{k+1} \rfloor$ for all k (while still keeping the hypothesis that g is nonredundant with $b_k \leq b_{k+1}$ for all k), Q remains an integral polyhedron, and $\text{proj}_{(\phi,z)}P = \text{proj}_{(\phi,z)}Q = \text{conv}(Y(g))$ is still integral. However, in this case, we have to modify the formulation of R slightly to ensure that $R = \text{conv}(Y(g))$ still holds. Let $U = \{k : k > 1, \lfloor b_{k-1} \rfloor < \lfloor b_k \rfloor\}$. For $k \in U$, let $p(k)$ denote the predecessor of k in U. The constraints (24) in formulation R now take the form

$$\phi \geq a_{p(k)}z - c_{p(k)} + \sum_{j=p(k)+1}^{k} (a_j - a_{j-1})(1 - f_j)(z - \lfloor b_k \rfloor) \text{ for } k \in U.$$

Observe that the breakpoints $\{b_k\}_{k=1}^K$ are crucial to the formulations P and Q independently of the values of the slopes $\{a_k\}_{k=1}^K$. This, plus their simplicity, and the fact that it is often natural to describe a function g in terms of the variables $\{s_k\}$ or $\{\sigma_k\}$ suggest the use of one of the formulations P or Q rather than formulation R.

We now also take into account the constraints in CIP.

Theorem 5 *If $z \in \mathbb{R}^n$ and $g(z) = \sum_{j=1}^{n} g_j(z_j)$ with each function g_j a piecewise linear convex function of a single variable, and A is totally unimodular, the polyhedron*

$$\begin{aligned} (\phi_j, z_j) \in \; & \mathrm{conv}(Y(g_j)) \text{ for } j = 1, \ldots, n \\ Az & \leq b \\ z & \geq 0 \end{aligned}$$

is integral whenever b is integral.

Proof. Using the same argument as in the proof of Theorem 2, it follows from Lemma 3 that all maximal bounded faces of R are of the form : $l_j \leq z_j \leq v_j$ for $j = 1, \ldots, K, Az \leq b$ where l_j, v_j are integers or infinite. As A is TU, the face is integral, and the claim follows. □

3.2 Functions of the Form $g(z_1, \ldots, z_n) = \max_{j=1,\ldots,n}(b_j - z_j)^+$.

Günlük and Pochet [1998] provide a complete answer to the question of how to model $\mathrm{conv}(Y(g))$ without additional constraints. In addition a case involving additional constraints almost as general as those treated below is dealt with in Pochet and Wolsey [1994]. Let $f_t = b_t - \lfloor b_t \rfloor$ for $t = 1, \ldots, n$.

Proposition 6 *The polyhedron*

$$\phi + z_j \geq b_j \text{ for } j = 1, \ldots, n \tag{26}$$

$$\phi \geq \textstyle\sum_{p=1}^{P} (f_{t_p} - f_{t_{p-1}})(\lceil b_{t_p} \rceil - z) \quad \text{for all } 1 \leq P \leq n, 0 = f_{t_0} < f_{t_1} \leq \ldots \leq f_{t_P} \tag{27}$$

$$\phi \geq \textstyle\sum_{p=1}^{P} (f_{t_p} - f_{t_{p-1}})(\lceil b_{t_p} \rceil - z) + (1 - f_{t_P})(\lceil b_{t_1} \rceil - z_{t_1} - 1) \quad \text{for all } 1 \leq P \leq n, 0 = f_{t_0} < f_{t_1} \leq \ldots \leq f_{t_P} \tag{28}$$

$$Az \leq b \tag{29}$$

$$\phi, z \geq 0 \tag{30}$$

is integral (in z) when A is the dual of a network flow matrix and b is integer.

Proof. We consider the different faces of the polyhedron in which s is bounded. Note that in such a face, at least one of the inequalities (27), (28) and $\phi \geq 0$ holds at equality.

The following claims essentially follow from the separation algorithm for the mixing inequalities (Pochet and Wolsey [1994], Günlük and Pochet [1998]).

i) $\phi = 0$ if and only if $z_j \geq \lceil b_j \rceil$ for all j.

ii) $\phi = \sum_{p=1}^{P}(f_{i_p} - f_{i_{p-1}})(\lceil b_{i_p} \rceil - z_{i_p})$ if and only if

$$1 \geq \lceil b_{i_1} \rceil - z_{i_1} \geq \ldots \geq \lceil b_{i_P} \rceil - z_{i_P} \geq 0$$

$$\lceil b_{i_p} \rceil - z_{i_p} \geq \lceil b_j \rceil - z_j \text{ for } i_{p-1} < j < i_p \text{ and } p = 1, \ldots, P$$

$$0 \geq \lceil b_j \rceil - z_j \text{ for } j > i_P.$$

iii) $\phi = \sum_{p=1}^{P}(f_{i_p} - f_{i_{p-1}})(\lceil b_{i_p} \rceil - z_{i_p}) + (1 - f_{i_P})(\lceil b_{i_1} \rceil - 1 - z_{i_1})$ if and only if

$$\lceil b_{i_1} \rceil - z_{i_1} \geq \ldots \geq \lceil b_{i_P} \rceil - z_{i_P} \geq \lceil b_{i_1} \rceil - 1 - z_{i_1} \geq 0$$

$$\lceil b_{i_p} \rceil - z_{i_p} \geq \lceil b_j \rceil - z_j \text{ for } i_{p-1} < j < i_p \text{ and } p = 1, \ldots, P$$

$$\lceil b_{i_1} \rceil - 1 - z_{i_1} \geq \lceil b_j \rceil - z_j \text{ for } j > i_P.$$

Thus every face is of the form

$$\begin{gathered} l_{ij} \leq z_i - z_j \leq v_{ij} \text{ for } i, j \in \{1, \ldots, n\} \\ l_j \leq z_j \leq v_j \text{ for } j \in \{1, \ldots, n\} \\ Az \leq b, \end{gathered}$$

with l_{ij}, l_j either integer or $-\infty$, and v_{ij}, v_j either integer or $+\infty$, and the claim follows. □

As an alternative to the exponential description of conv$(Y(g))$ provided by the mixing inequalities, we now give a simple derivation of an extended formulation from Pochet and Wolsey [1994].

Proposition 7 *An extended formulation for conv($Y(g)$) where $Y(g) = \{(\phi, z) \in \mathbb{R}_+^1 \times \mathbb{Z}^n : \phi + z_i \geq b_i$ for $i = 1, \ldots, n\}$ is*

$$\phi \geq \textstyle\sum_{i=0}^{n} f_i \delta_i + \mu \tag{31}$$

$$z_t \geq \textstyle\sum_{i=0}^{n} \lceil b_t - f_i \rceil \delta_i - \mu \text{ for } t = 1, \ldots, n \tag{32}$$

$$\textstyle\sum_{i=0}^{n} \delta_i = 1 \tag{33}$$

$$\mu \geq 0, \delta_i \geq 0 \text{ for } i = 1, \ldots, n \tag{34}$$

(here $f_0 = 0$).

Proof. The $n+1$ extreme points $\{\phi^j, z^j\}_{j=0}^n$ of $\text{conv}(Y(g))$ are given by $\phi^0 = 0, z_t^0 = \lceil b_t \rceil$ for $t = 1, \ldots, n$, and $\phi^j = f_j, z_t^j = \lceil b_t - f_j \rceil$ for $t = 1, \ldots, n$ as j varies over $\{1, \ldots, n\}$. Following Balas [1979], the convex hull of the union of these points is given by the constraints (31)-(34) with (31) and (32) as equalities, and $\mu = 0$. Letting e_j be j^{th} unit vector, and $\underline{1}$ be the vector of all 1's, the claim follows as the extreme rays of $Y(g)$ are $(\phi, z) = (1, 0)$, $(\phi, z) = (0, e_j), j = 1, \ldots, n$, and $(\phi, z) = (1, -\underline{1})$ represented by the variable μ. □

Theorem 8 *Let $Y(g) = \{(\phi, z) \in \mathbb{R}_+^1 \times \mathbb{Z}^n : \phi + z_i \geq b_i$ for $i = 1, \ldots, n\}$, let A be an $m \times n$ matrix, $b \in \mathbb{R}^m$ and $A(g) = \{(\phi, z) \in Y(g) : Az \leq b\}$. If A is the dual of a network flow matrix, and if $b \in \mathbb{Z}^m$, then an extended formulation for conv($A(g)$) is given by (31)–(34) and the constraints $Az \leq b$.*

Note that neither Proposition 6 nor Theorem 8 hold when A is TU, and it is not clear whether there is a polynomial algorithm for optimization over $A(g)$ when A is TU.

3.3 Functions of the Form $g(z, r) = \max_{j=1,\ldots,n}(b_j - r_j - z_j)^+$.

Here

$$Y(g) = \{(\phi, r, z) \in \mathbb{R}_+^1 \times \mathbb{R}_+^n \times \mathbb{Z}^n : \phi + r_t + z_t \geq b_t \text{ for } t = 1, \ldots, n\}.$$

For such sets, we can define valid "mixing" inequalities for $\text{conv}(Y(g))$ that are analogous to the inequalities (27) and (28):

$$\begin{aligned} \phi + \textstyle\sum_{p=1}^P r_p \geq \sum_{p=1}^P (f_{t_p} - f_{t_{p-1}})(\lceil b_{t_p} \rceil - z), \\ \text{for all } 1 \leq P \leq n, f_{t_1} < \ldots < f_{t_P} \end{aligned} \tag{35}$$

$$\begin{aligned} \phi + \textstyle\sum_{p=1}^P r_p \geq \sum_{p=1}^P (f_{t_p} - f_{t_{p-1}})(\lceil b_{t_p} \rceil - z) + (1 - f_{t_P})(\lceil b_{t_1} \rceil - z_{t_1} - 1), \\ \text{for all } 1 \leq P \leq n, f_{t_1} < \ldots < f_{t_P}. \end{aligned} \tag{36}$$

However, these inequalities do not suffice to give a description of $\text{conv}(Y(g))$.

Example 2

Let $n = 3$ and $b = (2.1, 0.4, 0.6)$, and consider

$$\begin{aligned} \phi + r_1 + z_1 &\geq 2.1 \\ \phi + r_2 + z_2 &\geq 0.4 \\ \phi + r_3 + z_3 &\geq 0.6 \\ \phi, r \geq 0, z &\in \mathbb{Z}^3. \end{aligned}$$

It is straightforward to check that the inequality

$$\phi + 0.5r_1 + 0.5r_2 + 0.5r_3 + 0.35z_1 + 0.2z_2 + 0.25z_3 \geq 1.2$$

is valid and facet-defining for conv(X). Clearly it is not a mixing inequality of the form (35) or (36). □

We do not know a characterization of the facets of conv($Y(g)$), but there is an extended formulation.

Theorem 9 *An extended formulation for conv*$(Y(g))$ *is*

$$\phi \geq \textstyle\sum_{i=0}^{n} f_i\delta_i + \mu \tag{37}$$

$$r_t \geq \textstyle\sum_{i=0}^{n} f_i^t \beta_i^t + \nu_t \text{ for } t = 1, \ldots, n \tag{38}$$

$$z_t \geq \lfloor b_t \rfloor + \textstyle\sum_{i:f_i<f_t}(\delta_i - \beta_i^t) - \sum_{i:f_i>f_t} \beta_i^t - \mu - \nu_t \text{ for } t = 1, \ldots, n \tag{39}$$

$$\beta_i^t \leq \delta_i \text{ for } t = 1, \ldots, n, i = 0, \ldots, n \tag{40}$$

$$\textstyle\sum_{i=0}^{n} \delta_i = 1 \tag{41}$$

$$\beta, \delta, \mu, \nu \geq 0, \tag{42}$$

where $f_0 = 0$*,* $f_i^t = f_t - f_i$ *if* $f_i < f_t$ *and* $f_i^t = 1 + f_t - f_i$ *if* $f_i > f_t$*.*

To prove this, we need the following lemma.

Lemma 10 *In each extreme point of conv($Y(g)$), either* $\phi = 0$*, or else there exists a* $t \in \{1, ..., n\}$ *such that both* $r_t = 0$ *and* $\phi + z_t = b_t$*.*

Proof. Let $(\bar{\phi}, \bar{r}, \bar{z}) \in Y(g)$ be such that $\bar{\phi} > 0$, and either $\bar{r}_t > 0$ or $\bar{\phi} + \bar{z}_t > b_t$, $t = 1, \ldots, n$. If $\bar{\phi} + \bar{r}_t + \bar{z}_t > b_t$ for all t, then $(\bar{\phi}, \bar{r}, \bar{z})$ is clearly not an extreme point of conv($Y(g)$). So let $\mathcal{T} = \{t = 1, ..., n : \bar{\phi} + \bar{r}_t + \bar{z}_t = b_t\}$, and let $\bar{\mathcal{T}} = \{t = 1, ..., n : t \notin \mathcal{T}\}$. By hypothesis, for each $t \in \mathcal{T}$, $r_t > 0$ must hold. Now define $\epsilon = \min\{\bar{\phi}, \min_{t\in\mathcal{T}}[\bar{r}_t], \min_{t\in\bar{\mathcal{T}}}[\bar{\phi} + \bar{r}_t + \bar{z}_t - b_t]\}$. It is clear that $\epsilon > 0$, and thus the following two points are in $Y(g)$:

$$\phi = \bar{\phi} - \epsilon; r_t = \bar{r}_t + \epsilon, t \in \mathcal{T}, r_t = \bar{r}_t, t \in \bar{\mathcal{T}}; z_t = \bar{z}_t, t = 1, \ldots, n$$
$$\phi = \bar{\phi} + \epsilon; r_t = \bar{r}_t - \epsilon, t \in \mathcal{T}, r_t = \bar{r}_t, t \in \bar{\mathcal{T}}; z_t = \bar{z}_t, t = 1, \ldots, n$$

Moreover, $(\bar{\phi}, \bar{y}, \bar{z})$ is a convex combination of these points and so cannot be an extreme point of conv($Y(g)$). The claim follows. □

Proof of Theorem 9: Given Lemma 10, it follows that $\mathrm{conv}(Y(g))$ has the following extreme points:

Case 0: $\phi = 0$. (Variable $\delta_0 = 1$).
For each $j = 1, \ldots, n$, either $r_j = 0$ and $z_j = \lceil b_j \rceil$ (variable $\beta_j^0 = 0$), or $r_j = f_j = f_0^j$, $z_j = \lfloor b_j \rfloor$ (variable $\beta_j^0 = 1$). Thus, in this case, both (38)–(39) hold at equality for $j = 1, \ldots, n$, where δ and β take the values indicated in the above sentences.

Case i: $r_i = 0$ and $\phi + z_i = b_i$ (variable $\delta_i = 1$) for some $i \in \{1, \ldots, n\}$.
If $f_j > f_i$, either $r_j = 0, z_j = \lceil b_j - f_i \rceil = \lceil b_j \rceil$ (variable $\beta_i^j = 0$), or $r_j = f_j - f_0^i, z_j = \lfloor b_j \rfloor$ (variable $\beta_i^j = 1$). If $f_j < f_i$, either $r_j = 0, z_j = \lceil b_j - f_i \rceil = \lceil b_j - 1 \rceil$ (variable $\beta_i^j = 0$), or $r_j = 1 + f_j - f_i, z_j = \lfloor b_j - 1 \rfloor$ (variable $\beta_i^j = 1$). If $f_j = f_i$, then $r_j = 0, z_j = b_j - f_i = \lfloor b_j \rfloor$ must hold, or else (ϕ, r, z) is not an extreme point of $\mathrm{conv}(Y(g))$. Thus, in this case as well, (37)–(39) hold at equality for $j = 1, \ldots, n$, where δ and β take the values indicated.

Therefore, similarly to the proof of Proposition 7, the convex hull of the extreme points of $\mathrm{conv}(Y(g))$ is exactly the projection of (37)–(42), with (37)–(39) at equality and $\mu = 0 = \nu_j$ for $j = 1, \ldots, n$, onto the (ϕ, r, z) space. The result follows as the extreme rays of $\mathrm{conv}(Y(g))$ are $(\phi, r, z) = (1, 0, 0)$, $(\phi, r, z) = (0, e_j, 0), j = 1, \ldots, n$, $(\phi, r, z) = (0, 0, e_j), j = 1, \ldots, n$, $(\phi, r, z) = (1, 0, -\underline{1})$ represented by the variable μ, and $(\phi, r, z) = (0, e_j, -e_j)$ represented by the variable ν_j for $j = 1, \ldots, n$. $\square$

This extended formulation is related to that presented in Subsection 3.2, but because of the extra continuous variable in each term $b_j - r_j - z_j$, it requires a quadratic rather than linear number of extra variables. Moreover, here it can be shown that dual network flow constraints in general *cannot* necessarily be added without losing integrality.

4 The Discrete Lot–sizing Problem

Here we consider the four variants of the discrete lot–sizing problem presented in Section 2, and apply the results of Section 3 to these models. Our goal is two-fold: to describe the convex hull of the solution sets for the different variants, if possible, and to give at an extended formulation that can be used easily in practice, for which computational results will be presented in the next section. Recall that *DLSBI* can be formulated as

$$\min \sum_{i=1}^{NI}\sum_{t=0}^{NT} h_t^i s_t^i + \sum_{i=1}^{NI}\sum_{t=1}^{NT} b_t^i r_t^i + \sum_{i=1}^{NI}\sum_{t=1}^{NT} q_t^i y_t^i \tag{1}$$

$$s_{t-1}^i + r_t^i + C^i y_t^i = d_t^i + s_t^i + r_{t-1}^i \text{ for } i = 1,...,NI, t = 1,...,NT \tag{2}$$

$$\sum_{i=1}^{NI} y_t^i \leq 1 \text{ for } t = 1,...,NT, \tag{3}$$

$$s_t^i \geq 0 \text{ for } i = 1,...,NI, t = 0,...,NT \tag{4}$$

$$r_t^i \geq 0 \text{ for } i = 1,...,NI, t = 1,...,NT \tag{5}$$

$$y_t^i \in \{0,1\} \text{ for } i = 1,...,NI, t = 1,...,NT \tag{6}$$

and that DLS, $DLSB$, and $DLSI$ can be obtained from this formulation by setting the variables s_0^i and/or r_t^i to 0 as appropriate. The solution sets of each the four variants of (2)-(6) are denoted $X^{DLSBI}, X^{DLS}, X^{DLSB}$ and X^{DLSI} respectively.

We first take note of the “additional” constraints that we will encounter.

Observation 4. The constraint matrix arising in the system

$$\begin{aligned} z_t^i = \textstyle\sum_{u=1}^t y_t^i & \text{ for all } i,t \\ \textstyle\sum_i y_t^i \leq 1 & \text{ for all } t \\ y_t^i \geq 0 & \text{ for all } i,t \end{aligned}$$

is a network matrix, see Figure 1, and is thus totally unimodular (TU).

Observation 5. The constraint matrix arising in the system

$$\begin{aligned} z_t = \textstyle\sum_{u=1}^t y_t & \text{ for all } t \\ 0 \leq y_t \leq 1 & \text{ for all } t \end{aligned}$$

can be written in the z-space as

$$\begin{aligned} 0 \leq z_1 \leq 1 & \\ 0 \leq z_t - z_{t-1} \leq 1 & \text{ for all } t \geq 2 \end{aligned}$$

and is thus the dual of a network flow matrix.

4.1 The Basic Problem DLS

Recall here that $r_t^i = 0$, for $i = 1,...,NI, t = 1,...,NT$ and $s_0^i = 0$ for all i. By summing the equations (2) for periods 1 up to t, we obtain $\sum_{u=1}^t C^i y_u^i = d_{1t}^i + s_t^i$. Eliminating s_t^i, the objective function (1) becomes $\min \sum_{i=1}^{NI}\sum_{t=1}^{NT} \hat{q}_t^i y_t^i + K_{DLS}$ where

$\hat{q}_t^i = q_t^i + C^i \sum_{u=t}^{NT} h_u^i$ and $K_{DLS} = -\sum_{i=1}^{NI} \sum_{t=1}^{NT} h_t^i d_{1t}^i$, for $i = 1, ..., NI$ and $t = 1, ..., NT$, and the nonnegativity of s_t^i leads us to replace (2) by

$$C^i \sum_{u=1}^{t} y_u^i \geq d_{1t}^i. \tag{43}$$

Dividing (43) by C^i and rounding leads to the Gomory cut (Gomory [1960])

$$\sum_{u=1}^{t} y_u^i \geq \lceil \frac{d_{1t}^i}{C^i} \rceil,$$

which clearly has the same set of integer solutions as (43). So Observation 4 immediately implies the following result.

Theorem 11 *(Folklore). conv(X^{DLS}) is described by the constraints*

$$\begin{gathered}
\sum_{i=1}^{NI} y_t^i \leq 1 \text{ for } t = 1, ..., NT \\
\sum_{u=1}^{t} y_u^i \geq \lceil \frac{d_{1t}^i}{C^i} \rceil, \text{ for } i = 1, ..., NI, t = 1, ..., NT \\
y_t^i \geq 0 \text{ for } i = 1, ..., NI, t = 1, ..., NT.
\end{gathered}$$

4.2 With Backlogging $DLSB$

Here again the goal is to find a description of conv(X^{DLSB}). Adopting the same approach as above, we aggregate the constraints (2) giving

$$C^i \sum_{u=1}^{t} y_u^i = d_{1t}^i + s_t^i - r_t^i$$

and then eliminate either the variables s_t^i or the variables r_t^i.

Eliminating the variables r_t^i, we can rewrite $DLSB$ as

$$\begin{gathered}
\min \sum_{i=1}^{NI} \sum_{t=0}^{NT} \hat{h}_t^i s_t^i + \sum_{i=1}^{NI} \sum_{t=1}^{NT} \hat{q}_t^i y_t^i + K_{DLSB} \\
s_t^i \geq C^i \sum_{u=1}^{t} y_u^i - d_{1t}^i \text{ for all } i, t \\
s_t^i \geq 0 \text{ for all } i, t \\
\sum_i y_t^i \leq 1 \text{ for all } t \\
y_t^i \geq 0 \text{ for all } i, t \\
y_t^i \in \mathbb{Z} \text{ for all } i, t
\end{gathered}$$

where $\hat{h}_t^i = h_t^i + b_t^i$, $\hat{q}_t^i = q_t^i - C^i \sum_{u=t}^{NT} b_u^i$ and $K_{DLSB} = \sum_{i=1}^{NI} \sum_{t=1}^{NT} b_t^i d_{1t}^i$.

Now we can combine Theorem 4 with formulation Q and $K = 1$, and Observation 4 showing that the additional constraints are network constraints. Then from Theorem 5 we have

Theorem 12 *$conv(X^{DLSB})$ is described by the constraints*

$$\sum_{i=1}^{NI} y_t^i \leq 1 \text{ for } t = 1, \ldots, NT \tag{44}$$

$$s_t^i \geq C^i \sum_{u=1}^{t} y_u^i - d_{1t}^i \text{ for } i = 1, \ldots, NI, t = 1, \ldots, NT \tag{45}$$

$$s_t^i \geq C^i(1 - f_t^i)(\sum_{u=1}^{t} y_u^i - \lfloor \frac{d_{1t}^i}{C^i} \rfloor) \text{ for } i = 1, \ldots, NI, t = 1, \ldots, NT \tag{46}$$

$$s_t^i, y_t^i \geq 0 \text{ for } i = 1, \ldots, NI, t = 1, \ldots, NT \tag{47}$$

where $f_t^i = \frac{d_{1t}^i}{C^i} - \lfloor \frac{d_{1t}^i}{C^i} \rfloor$ for all i, t. Note that in practice it is not necessary to eliminate the r_t^i variables. It suffices to keep the original formulation and just add the additional constraints (46).

The same argument allows us to treat problems with backlogging and target safety stock levels, denoted $DLSBSS$. For each item, let SS^i be the target safety stock level, and suppose that in each period the stock costs as a function of net inventory $v_t^i = C^i \sum_{u=1}^{t} y_t^i - d_{1t}^i$ are given by

$$\begin{aligned} \tilde{h}^i(v) &= \begin{cases} -\alpha_1 v + \alpha_2 SS^i & \text{for} \quad v \leq 0, \\ \alpha_2(SS^i - v) & \text{for} \quad 0 \leq v \leq SS^i \\ \alpha_3(v - SS^i) & \text{for} \quad v \geq SS^i \end{cases} \\ &= -\alpha_1 v + \alpha_2 SS^i + \max\{0, (\alpha_1 - \alpha_2)v, (\alpha_1 - \alpha_2)v + (\alpha_2 + \alpha_3)(v - SS^i)\}. \end{aligned}$$

An example of such a function with $\alpha_1 = 1$, $\alpha_2 = 0.25$, and $\alpha_3 = 0.125$, is illustrated in Figure 3.

Alternatively for each i, t pair, we have that $v_t^i = s_t^i - r_t^i, s_t^i, r_t^i \geq 0$ and

$$s_t^i = s_t^{i1} + s_t^{i2}, 0 \leq s_t^{i1} \leq SS^i, s_t^{i2} \geq 0, \tag{48}$$

with corresponding costs

$$\alpha_1 r_t^i + \alpha_2(SS^i - s_t^{i1}) + \alpha_3 s_t^{i2}. \tag{49}$$

Note that s_t^{i1} represents the amount of positive stock lying between 0 and SS^i, and s_t^{i2} the amount exceeding SS^i with $s_t^i = s_t^{i1} + s_t^{i2}$.

Now we use formulation P in Theorem 4, and Observation 4 for the additional constraints. Then Theorem 5 gives:

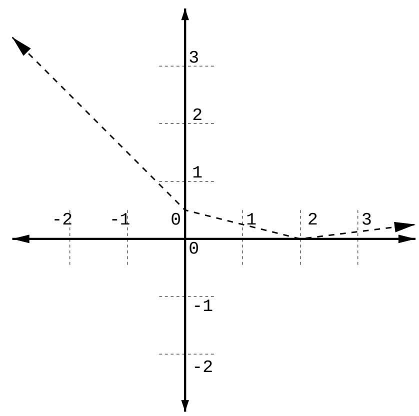

Figure 3: An example of the function $\tilde{h}^i$

Theorem 13 *$conv(X^{DLSBSS})$ is given by:*

$$s_t^{i1} + s_t^{i2} \geq C^i \sum_{u=1}^t y_u^i - d_{1t}^i \text{ for all } i, t \tag{50}$$

$$s_t^{i1} + s_t^{i2} \geq C^i(1 - f_t^{i1})(\sum_{u=1}^t y_u^i - \zeta_t^{i1}) \text{ for all } i, t \tag{51}$$

$$s_t^{i2} \geq C^i(1 - f_t^{i2})(\sum_{u=1}^t y_u^i - \zeta_t^{i2}) \text{ for all } i, t \tag{52}$$

$$\sum_i y_t^i \leq 1 \text{ for all } t \tag{53}$$

$$0 \leq s_t^{i1} \leq SS^i, 0 \leq s_t^{i2}, 0 \leq y_t^i \leq 1 \text{ for all } i, t \tag{54}$$

where $\zeta_t^{i1} = \lfloor \frac{d_{1t}^i}{C^i} \rfloor, f_t^{i1} = \frac{d_{1t}^i}{C^i} - \zeta_t^{i1}$, $\zeta_t^{i2} = \lfloor \frac{d_{1t}^i + SS^i}{C^i} \rfloor$ and $f_t^{i2} = \frac{d_{1t}^i + SS^i}{C^i} - \zeta_t^{i2}$ for all i, t. Again it is not necessary to eliminate the backlog variables. In practice it suffices to write the initial formulation using (48) and (49) and add the constraints (51)-(52).

The example below illustrates how the results in Section 3 apply to $DLSBSS$. In this example, we consider one item and one time period and take values d_t^i, SS^i, and C^i that have all been scaled so that $C^i = 1$.

Example 1 (continued)

For a given i and t, let $d_t^i = 1.5$, $SS^i = 2$, $\alpha_1 = 1$, $\alpha_2 = 0.25$, $\alpha_3 = 0.125$. Further, let $C^i = 1$. Thus we can define

$$\begin{aligned} \tilde{H}^i(v) &= \begin{cases} -1v + 0.5 & \text{for} \quad v \leq 0, \\ 0.25(2 - v) & \text{for} \quad 0 \leq v \leq 2 \\ 0.125(v - 2) & \text{for} \quad v \geq 2 \end{cases} \\ &= -1v + 0.5 + \max\{0, 0.75v, 0.75v + (0.375)(v - 2)\}; \end{aligned}$$

thus $\tilde{H}$ is the net inventory cost function, where v is the net inventory (i.e., stock minus backlog). We next define net inventory cost for i and t as a function of $z_t^i = \sum_{u=1}^t y_t^i$:

$$\tilde{h}_t^i(z_t^i) = \max\{-1z_t^i + 2, -0.25z_t^i + 0.875, 0.125z_t^i - 0.4375\}.$$

But this function is the same as $h(z)$ defined earlier in Example 1; therefore, we can define

$$\tilde{g}_t^i(z) = \max\{0, 0.75z - 1.125, 1.125z - 3.1875\},$$

(the same function as $g(z)$) and use this to obtain the polyhedral descriptions listed earlier in this example and illustrated in Figure 2. More specifically, for this example the constraints (54), (50)–(52), take the form

$$\begin{gathered} 0 \leq s_t^{i1} \leq 2, s_t^{i2} \geq 0 \\ s_t^{i1} + s_t^{i2} \geq z_t^i - \tfrac{3}{2} \\ s_t^{i1} + s_t^{i2} \geq \tfrac{1}{2}(z_t^i - 1) \\ s_t^{i2} \geq \tfrac{1}{2}(z_t^i - 3) \end{gathered}$$

and thus correspond exactly to the formulation of polyhedron P. Note here that $\zeta_t^{i1} = 1$, $f_t^{i1} = \frac{1}{2}$, $\zeta_t^{i2} = 3$ and $f_t^{i2} = \frac{1}{2}$. □

4.3 With Initial Stock Variables *DLSI*

Here we consider first the single item case, and drop the superscript i for simplicity. As usual, we aggregate the constraints (2) giving

$$s_0 + C\sum_{u=1}^{t} y_u = d_{1t} + s_t,$$

and then eliminate the variables s_t for $t \geq 1$ leading to the reformulation

$$\begin{gathered} \min \tilde{h}_0 s_0 + \textstyle\sum_{t=1}^{NT} \tilde{q}_t y_t + K_{DLSI} \\ s_0 \geq d_{1t} - C\textstyle\sum_{u=1}^{t} y_u \text{ for } t = 1, ..., NT, \\ s_0 \geq 0 \\ 0 \leq y_t \leq 1 \text{ for } t = 1, ..., NT, \\ y_t \in \mathbb{Z} \text{ for } t = 1, ..., NT, \end{gathered}$$

where $\tilde{h}_0 = \sum_{t=0}^T h_t$, $\tilde{q}_t = q_t + C\sum_{u=t}^T h_u$, for $t = 1, ..., T$, and $K_{DLSI} = -\sum_{t=1}^T h_t d_{1t}$. Now implicitly using the substitution $z_t = \sum_{u=1}^t y_u$, for $t = 1, ..., NT$, we can use Propositions 6 and 8 to give a reformulation of the convex objective function, noting that Observation 5 shows that the additional constraints are dual network flow constraints.

Theorem 14 *When $NI = 1$, conv(X^{DLSI}) is described by the inequalities*

$$s_0 \geq d_{1t} - C\sum_{u=1}^t y_u \tag{55}$$

$$s_0 \geq \sum_{p=1}^P C(f_{t_p} - f_{t_{p-1}})(\eta_{t_p} - \sum_{u=1}^{t_p} y_u) \quad \text{for all } 1 \leq P \leq n, 0 = f_{t_0} < f_{t_1} \leq \ldots \leq f_{t_P} \tag{56}$$

$$s_0 \geq \sum_{p=1}^P C(f_{t_p} - f_{t_{p-1}})(\eta_{t_p} - \sum_{u=1}^{t_p} y_u) + C(1 - f_{t_P})(\eta_{t_1} - \sum_{u=1}^{t_1} y_u - 1) \quad \text{for all } 1 \leq P \leq n, 0 < f_{t_0} \leq f_{t_1} \leq \ldots \leq f_{t_P} \tag{57}$$

$$s_0 \geq 0, 0 \leq y_t \leq 1 \text{ for } t = 1, \ldots, NT, \tag{58}$$

where $\eta_t = \lceil \frac{d_{1t}}{C} \rceil$ and $f_t = \frac{d_{1t}}{C} - \lfloor \frac{d_{1t}}{C} \rfloor$, for $t = 1, ..., T$, or equivalently as the (s, y) projection of

$$s_0 \geq C\sum_{t=0}^n f_t\delta_t + C\mu, \tag{59}$$

$$\sum_{u=1}^t y_u \geq \sum_{\tau=0}^{NT} \lceil \frac{d_{1t}}{C} - f_\tau \rceil \delta_\tau - \mu \text{ for } t = 1, ..., NT \tag{60}$$

$$\sum_{t=0}^{NT} \delta_t = 1 \tag{61}$$

$$\mu \geq 0, \delta_t \geq 0, \text{ for } t = 1, ..., NT \tag{62}$$

$$0 \leq y_t \leq 1 \text{ for } t = 1, ..., NT \tag{63}$$

(here $f_0 = 0$).

Note that Pochet and Wolsey [1994] have given a complete description of the constant capacity lot–sizing problem with Wagner-Whitin costs. $DLSI$ is a face of this polyhedron obtained by setting $x_t = Cy_t$ where x_t is the variable amount produced in period t. So the above theorem is also a corollary of this earlier result.

For the multi–item case with $NI > 1$, the complexity of $DLSI$ is not known. The following two item example shows that the intersection of the convex hulls of the single-item polyhedra and the linking constraints $\sum_{i=1}^{NI} y_t^i \leq 1$ for $t = 1, \ldots, NT$ does not give the convex hull even for $NI = 2$.

Example 3

Let $NI = 2$ and $NT = 4$. Let $C^1 = C^2 = 10$ with demands $d^1 = (2, 6, 6, 2)$ and $d^2 = (4, 4, 4, 4)$. One facet of the convex hull is induced by

$$2s_0^1 + s_0^2 + 18y_1^1 + 16y_2^1 + 12y_3^1 + 4y_1^2 + 2y_2^2 + 6y_3^2 \geq 34.$$

This inequality is clearly not a mixing inequality of the form (56) or (57). $\square$

Nevertheless the above results suggest that the reformulations of Theorem 14 should provide a good approximation to conv(X^{DLSI}) even when $NI > 1$.

4.4 With Backlogging and Initial Stocks $DLSBI$

Again it is natural to ask whether one can find a formulation for conv(X^{DLSBI}) when $NI = 1$. It is easy to find valid inequalities for this model.

Proposition 15 *The mixing inequalities*

$$\begin{gathered} s_0 + \sum_{p=1}^{P} r_{t_p} \geq \sum_{p=1}^{P} C(f_{t_p} - f_{t_{p-1}})(\eta_{t_p} - \sum_{u=1}^{t_p} y_u) \\ \text{for all } 1 \leq P \leq n, 0 = f_{t_0} < f_{t_1} \leq \ldots \leq f_{t_P} \\ s_0 + \sum_{p=1}^{P} r_{t_p} \geq \sum_{p=1}^{P} C(f_{t_p} - f_{t_{p-1}})(\eta_{t_p} - \sum_{u=1}^{t_p} y_u) + C(1 - f_{t_P})(\eta_{t_1} - \sum_{u=1}^{t_1} y_u - 1) \\ \text{for all } 1 \leq P \leq n, 0 < f_{t_0} \leq f_{t_1} \leq \ldots \leq f_{t_P} \end{gathered}$$

are valid for X^{DLSBI} when $NI = 1$.

Unfortunately Example 2 is easily converted into an instance of $DLSBI$ showing that these inequalities do not suffice to describe conv(X^{DLSBI}), even when $NI = 1$.

Using the results of Theorem 9, we know that a stronger formulation is given by (37)-(42). For the general case ($NI \geq 1$), this leads to the reformulation

$$s_0^i \geq C^i \sum_{t=0}^{NT} f_t^i \delta_t^i + C^i \mu^i, \text{ for } i = 1, ..., NI, \tag{64}$$

$$r_t^i \geq C^i \sum_{u=0}^{NT} f_{ut}^i \beta_t^i + C^i \nu_t^i, \text{ for } i = 1, ..., NI, t = 1, ..., NT, \tag{65}$$

$$\begin{gathered} \sum_{u=1}^{t} y_u^i \geq \eta_t^i - 1 + \sum_{u: f_u^i < f_t^i} (\delta_u^i - \beta_{ut}^i) - \sum_{u: f_u^i > f_t^i} \beta_{ut}^i - \mu^i - \nu_t^i, \\ \text{for } i = 1, ..., NI, t = 1, ..., NT, \end{gathered} \tag{66}$$

$$\beta_{ut}^i \leq \delta_u^i \text{ for } i = 1, ..., NI, t = 1, ..., NT, u = 0, ..., NT \tag{67}$$

$$\sum_{t=0}^{NT} \delta_t^i = 1, \text{ for } i = 1, ..., NI, \tag{68}$$

$$\beta, \delta, \mu, \nu \geq 0 \tag{69}$$

$$\sum_i y_t^i \leq 1 \quad \text{for } t = 1, ..., NT \tag{70}$$

$$y_t^i \geq 0 \quad \text{for } i = 1, ..., NI, t = 1, ..., NT. \tag{71}$$

Here again $f_0^i = 0$ for $i = 1, \ldots, NI$, and

$$f_{ut}^i = \begin{cases} f_u^i - f_t^i & \text{if} \quad f_u^i \geq f_t^i \\ 1 + f_u^i - f_t^i & \text{if} \quad f_u^i < f_t^i, \end{cases}$$

Unfortunately, the (s, r, y) projection of (64)-(69) is not necessarily integral, even when $NI = 1$. The reason is that in this case, the substitution $z_t = \sum_{u=1}^{t} y_u$, for $t = 1, ..., NT$, yields a formulation like (37)-(42), but with additional constraints. Even though these constraints are dual network flow constraints, there are still many examples where the addition of such constraints to (37)-(42) yields fractional extreme points. However, it again appears likely that (64)-(69) provides a good approximation of the convex hull in both the single and the multi–item cases.

It is possible to define an extended formulation for $DLSBI$ with $NI = 1$ that is similar to (64)-(69), but with even more variables; thus $DLSBI$ is polynomially solvable when $NI = 1$. As with $DLSI$, the complexity when $NI > 1$ is not known.

5 Computational Results

In this section we test the reformulations proposed in Section 4 on the original practical problem motivating this research and its variants.

The practical problem encountered is a version of $DLSB$ with $NI = 30$ and $NT = 60$, and with additional complicating constraints that limit the choice of production sequences. Specifically each item either belongs to a basic family or to one of five special families, and production of one or more items from a special family must be followed by production of items from the basic family. This (original) instance is called $DLSB - o$. The (pure) instance without the complicating constraints is $DLSB - p$. We have also created two instances of a variant with safety stock levels denoted $DLSBSS - o$ and $DLSBSS - p$. Using the same data we have also created two instances of $DLSI$ and $DLSBI$ respectively. The reformulated instances are indicated by an "r" at the end of their names.

The data (including the capacities and demands) are taken from the original practical instance. There is a unit holding cost $h_t^i = 0.25$, for all i, t, and a unit backlogging cost $b_t^i = 1.0$ for all i, t. For the models $DLSI$ and $DLSBI$, we assign an initial cost of $h_0^i = 2.0$ for all i, based on the assumption that it is more expensive to buy in initial inventory than to have backlogging in a single period.

All the instances were solved with a commercial mixed integer programming system XPRESS, version 12 (Dash Associates [2000]) The tests were run on a Pentium II PC (450 MHz) with 128 Mb of RAM, running under Windows NT 4.0. In all the runs the root node LP was solved with the barrier algorithm. The barrier algorithm, the cut generation, and the branch–and–bound process were run with the system defaults.

(Note that the cuts that XPRESS generates have nothing to do with the inequalities added in our reformulations of $DLSB$ and $DLSBSS$.)

In presenting results we list the number of rows, columns, and nonzero entries in the matrix to illustrate the size of the initial problems and reformulations. For the solution process, we list the value of the lower bound provided by the LP relaxation (LP), the time it takes to solve the LP (LP time), the value of the lower and upper bounds on the termination of the branch–and–bound algorithm, and the total time to solve the problem (IP time)–note that this includes the time taken to solve the LP at the root node. An asterisk * indicates that optimality was not proved within the given time.

5.1 $DLSB$ and related models

First the two instances were run with the initial formulation provided by the user, and then using the reformulation (44)-(47). We know that after reformulation $DLSB-pr$ will be solved just using linear programming. The results of these four runs are shown in Table 1.

Instance	#col	#row	#non-0	LP	LP time	Best LB	Best UB	IP time
DLSB-p	5160	1621	12668	1627056	1	1759969	2188995	900*
DLSB-o	5520	2353	16723	2507612	3	2711504	3274036	900*
DLSB-pr	5160	3181	51974	1858181	7	1858181	1858181	7
DLSB-or	5520	3913	65863	3059263	15	3069539	3069539	56

Table 1: $DLSB$ instances before and after reformulation

Note that for $DLSB-p$, the duality gap after 15 minutes is greater than 20 %. The original practical instance $DLSB-o$ is also intractable, but after reformulation it solves within one minute.

We have also adapted the example to illustrate the effect of applying Theorem 5 to $DLSBSS$, the problem with safety stocks. For the costs we have chosen $\alpha_1 = 1, \alpha_2 = 0.25$ and $\alpha_3 = 0.125$ as in Example 1, and the safety stock is taken to be the average demand $SS^i = \frac{d_{1,NT}}{NT}$. As reformulation we have taken (50)-(54).

Given the results of Section 3, we expect the reformulation of $DLSBSS$ to be as effective as that for $DLSB$. Table 2 shows that this is the case for our example.

5.2 $DLSI$ and related models

In reformulating $DLSI$, we use the extended formulation (3)–(6), in addition to the constraints (59)–(63) for each item. Since $NI = 30$, there is no guarantee that the LP solution will be integer, even for the pure model.

Instance	#col	#row	#non-0	LP	LP time	Best LB	Best UB	IP time
DLSBSS-p	7801	3182	18670	1526600	2	1551703	3267920	900*
DLSBSS-o	8161	3854	20943	2430161	3	2629447	3412809	900*
DLSBSS-pr	7801	6302	118510	1801956	26	1801956	1801956	26
DLSBSS-or	8161	6974	120783	3012803	43	3024099	3024099	124

Table 2: $DLSBSS$ instances before and after reformulation

Table 3 illustrates the advantages of using this reformulation. Note that the instance $DLSI - pr$ is solved by linear programming (without branching) even though $NI > 1$. Also note that one result of the increased size of the reformulations is that the linear programs take longer to solve. However, this is more than compensated for by the improvement in the bounds.

Instance	#col	#row	#non-0	LP	LP time	Best LB	Best UB	IP time
DLSI-p	3390	1621	7830	1627056	1	1873472	2406455	900*
DLSI-o	3750	2353	11885	2143685	2	2296846	2779406	900*
DLSI-pr	5250	3481	104685	2080225	11	2080225	2080225	11
DLSI-or	5610	4213	108740	2436922	17	2436922	2436922	34

Table 3: $DLSI$ instances before and after reformulation

5.3 $DLSBI$ and related models

Here we used the reformulation (3)–(6), (64)–(69). Note that in this case even if $NI = 1$, this reformulation is not guaranteed to be integral. Moreover, since using the reformulation given by (64)–(69) increases the number of variables by quadratic order, there is a question of the effect of this reformulation on the time it takes to solve the LP's (at the root node and in the branch–and–bound tree). Nevertheless, we expect our reformulation to be much tighter than the original formulation, tight enough to be worth the extra time taken to solve the LP's. This expectation is confirmed by the results in Table 4.

Note that the pure, reformulated instance solves at the root node, even though theoretically this is not guaranteed to occur. Even with the complicating constraints, our reformulation allows us to find an optimal solution fairly quickly, although it takes a little longer than 15 minutes to prove that it is optimal.

Instance	#col	#row	#non-0	LP	LP time	Best LB	Best UB	IP time
DLSBI-p	5190	1621	12724	1627056	1	1757539	2165615	900*
DLSBI-o	5550	2353	16779	2051161	2	2120799	2394840	900*
DLSBI-pr	118650	115081	476839	1847029	182	1847029	1847029	182
DLSBI-or	119010	115813	480894	2202354	425	2202486	2203063	900*

Table 4: $DLSBI$ instances before and after reformulation

6 Conclusions

In this paper we have analyzed the polyhedral structure of some basic discrete lot–sizing models. Our analysis was motivated by problems encountered in working with a large international manufacturer. The models that we have considered have not been treated often before in the literature, and for most of them we have provided basic polyhedral results.

Our analysis of these lot–sizing problems has led us to consider related convex integer programs. For separable functions, we have shown how to reformulate CIP as an LP using specific MIR inequalities. For certain specific, non-separable functions, we have given extended formulations that allow the corresponding CIP to be reformulated as an LP. These results enable to us define tight formulations for the lot–sizing problems in question.

We wish to emphasize for practitioners that all the computational results reported in Section 5 have been obtained using standard, commercial mixed integer programming software. The formulations in Section 5 were written in an algebraic modeling language, the data was entered in a spreadsheet, and then a matrix was output in a standard format (such as MPS). This matrix was then read and optimized by a commercial mixed integer programming solver. No special–purpose cutting plane strategies were implemented, and no special optimization strategies were used. These formulations should therefore be easy to use on other instances of discrete lot–sizing problems.

While much is known about the polyhedral structure of DLS with startup variables, it is not clear that this knowledge is sufficient to solve $DLSB$, or the other models considered here, when startup variables are present. The analysis of such models is a challenging area of future research.

In addition, there is much that is still unknown about convex integer programming, and in particular the borderline between theoretically easy and hard problems. Our analysis suggests that even the simplest multi–variate functions pose more difficulties than separable ones, but important issues remain unresolved. Such issues are other possible areas of future research.

References

R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Englewood Cliffs, NJ, 1993.

E. Balas. Disjunctive programming. *Annals of Discrete Mathematics*, 5:3–51, 1979.

Dash Associates. *XPRESS–MP Reference Manual*, 2000. Release 12.

C.A. van Eijl. *A polyhedral approach to the discrete lot–sizing and scheduling problem*. PhD thesis, Technische Universiteit Eindhoven, June 1996.

C.A. van Eijl and C.P.M. van Hoesel. On the discrete lot–sizing and scheduling problem with Wagner–Whitin costs. *OR Letters* 20:7–13, 1997.

B. Fleischmann. The discrete lot–sizing and scheduling problem. *European Journal of Operational Research*, 44:337–348, 1990.

B. Fleischmann. The discrete lot–sizing and scheduling problem with sequence–dependent setup costs. *European Journal of Operational Research*, 75:395–404, 1994.

R.E. Gomory. Solving linear programs in integers. In R.E. Bellman and M. Hall, Jr, editors, *Combinatorial Analysis*, pages 211–216. American Mathematical Society, 1960.

O. Günlük and Y. Pochet. Mixing MIR inequalities for mixed integer programs. *Mathematical Programming*, 2001. To appear.

D.S. Hochbaum and J.G. Shantikumar. Convex separable optimization is not much harder than linear optimization. *Journal of ACM*, 37:843–862, 1990.

C.P.M. van Hoesel, R. Kuik, M. Salomon, and L.N. van Wassenhove. The single item discrete lot–sizing and scheduling problem: optimization by linear and dynamic programming. *Discrete Applied Mathematics*, 48:289–303, 1994.

G.L. Nemhauser and L.A. Wolsey. *Integer and Combinatorial Optimization*. Wiley, New York, 1988.

Y. Pochet and L.A. Wolsey. Polyhedra for lot–sizing with Wagner–Whitin costs. *Mathematical Programming*, 67:297–323, 1994.