Lot-sizing on a tree

Marco Di Summa[1] and Laurence A. Wolsey[2,*]

[1]Dipartimento di Matematica Pura ed Applicata, Università degli Studi di Padova.

[2]Center of Operations Research and Econometrics (CORE), Université catholique de Louvain.

May 3, 2007

Abstract

For the problem of lot-sizing on a tree with constant capacities, or stochastic lot-sizing with a scenario tree, we present various reformulations based on mixing sets. We also show how earlier results for uncapacitated problems involving (Q, S_Q) inequalities can be simplified and extended. Finally some limited computational results are presented.

Keywords: Lot-Sizing, Scenario Tree, Mixing Sets.

*Corresponding author: Laurence A. Wolsey, CORE, 34 Voie du Roman Pays. 1348 Louvain-la-Neuve, Belgium. email: wolsey@core.ucl.ac.be

1 Introduction

In two recent papers Guan et al. have examined the problem of stochastic uncapacitated lot-sizing based on a tree of scenarios. In the first paper [5], using a mixed integer programming formulation of the problem, the authors developed a family of valid inequalities, called (Q, S_Q) inequalities, and carried out some computational experiments to demonstrate their utility. In the second [6] they showed that the (Q, S_Q) inequalities suffice to describe the convex hull of the set of solutions when there is just one period with uncertain outcomes. More recently in [7] Guan and Miller have also developed a polynomial dynamic programming algorithm for the same problem. The goal here is to clarify and extend these results on the use of valid inequalities to tighten formulations to problems with constant capacities using an underlying model of lot-sizing on a tree, and to show the role of mixing sets in providing strong relaxations for the problem (see also [4]).

Specifically in Section 2 we introduce the problem of lot-sizing on a tree. We then demonstrate two ways to obtain mixing set relaxations. In Section 3 we present a simple description of the (Q, S_Q) inequalities and then prove that these inequalities are all mixing inequalities. In Section 4 we consider when the mixing sets suffice to describe the convex hull of the lot-sizing problem on a tree and in particular we show that this holds for a discrete version of the one period newsboy problem with constant capacities.

Finally in Section 5 we present limited computational results. Even though the default cutting planes (such as flow cover and path inequalities) of the MIP solvers increase the lower bounds considerably on the instances tried, for a fixed running time the mixing set reformulations and inequalities lead to significantly improved lower and upper bounds compared to those of the original formulation.

2 Lot-Sizing on a Tree and Mixing Sets

Given a rooted directed out-tree $T = (V, A)$, let $D(v)$ be the set of direct successors of v, $S(v)$ the set of all successors of v, and $P(j, k)$ with $k \in S(j)$ the set of nodes on the path from j to k. Node $r = 0 \in V$ is the root. $L = \{v \in V : S(v) = \emptyset\}$ are the leaves. We add a dummy node -1 and an arc $(-1, 0)$. Let $p(v)$ be the unique predecessor of v for all $v \in V$.

The lot-sizing problem on a tree *LS-TREE* is defined as the following mixed integer program,

$$\min \textstyle\sum_{v\in V}(P_v x_v + Q_v y_v) + \sum_{v\in V\cup\{-1\}} H_v s_v$$

$$s_{p(v)} + x_v = d_v + s_v \quad \text{for all } v \in V \tag{1}$$

$$x_v \leq C_v y_v \quad \text{for all } v \in V \tag{2}$$

$$s \in \mathbb{R}_+^{|V|+1}, x \in \mathbb{R}_+^{|V|}, y \in \{0,1\}^{|V|}, \tag{3}$$

with production costs P_v, fixed costs Q_v, demands d_v and capacity C_v for all $v \in V$, and storage costs H_v for all $v \in V \cup \{-1\}$. Note the special form of the balance constraints (1), in which the flow s_v out of node $v \in V \setminus L$ is the inflow to each direct successor node $w \in D(v)$, see Figure 1.

Note that an alternative formulation is obtained if we replace the balance constraints (1) by the equations (we write s instead of s_{-1})

$$s + \sum_{u\in P(0,v)} x_u = d_{0v} + s_v \quad \text{for all } v \in V$$

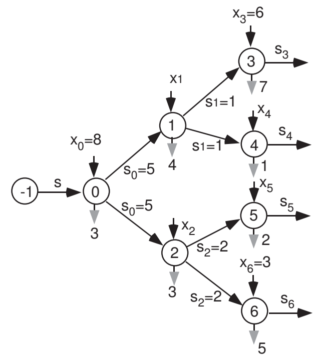

Figure 1: Lot-sizing on a tree.

or by the inequalities

$$s + \sum_{u \in P(0,v)} x_u \geq d_{0v} \quad \text{for all } v \in V,$$

where we set $d_{uv} = \sum_{w \in P(u,v)} d_w$.

To treat the stochastic lot-sizing problem, where the tree structure corresponds to the multistage structure of production decisions and demand observations, the problem of minimizing total expected cost can be modeled by weighting the terms in the objective function by appropriate probabilities.

Mixing Sets

The convex hull of the mixing set

$$X^{MIX} = \{(x, z) \in \mathbb{R}_+ \times \mathbb{Z}^n_+ : x + z_t \geq b_t \text{ for } t = 1, \ldots, n\}$$

has been studied in several papers. In particular a compact extended formulation appears in [9]. The *mixing inequalities* that suffice to describe the convex hull in the original (x, z) space are presented in [8] and an $O(n \log n)$ separation algorithm for these inequalities is given in [11].

The result that we use below is for the special case in which $b_t \leq 1$ for all t.

Proposition 1 *Consider the rescaled "uncapacitated" mixing set*

$$X^{MIX}_U = \{(x, z) \in \mathbb{R}_+ \times \mathbb{Z}^n_+ : x + Mz_t \geq b_t \text{ for } t = 1, \ldots, n\},$$

where $0 = b_0 \leq b_1 \leq \cdots \leq b_n < M$. *Let* $T = \{i_1, \ldots, i_{|T|}\} \subseteq \{1, \ldots, n\}$ *with* $i_j < i_{j+1}$ *for* $j = 1, \ldots, |T| - 1$, *and* $i_0 = 0$. *Then the* simple mixing inequalities

$$x \geq \textstyle\sum_{t=1}^{|T|} (b_{i_t} - b_{i_{t-1}})(1 - z_{i_t})$$

(together with $z_t \geq 0$*) give the convex hull of* X^{MIX}_U.

2.1 Mixing Set Relaxations with Constant Capacities

Here we suppose that the capacities are constant at each node: $C_v = C$ for all $v \in V$.

Given two distinct nodes v, w, let $\mu(v, w)$ be the common root, i.e. the root of the smallest subtree containing both v and w. Also given a path $P(u, v)$, let $\bar{P}(u, v) = P(u, v) \setminus \{u\}$.

Proposition 2 *For all $v \in V \cup \{-1\}$, the mixing set $X_1^{MIX}(v)$:*

$$s_v + Cz_{vw} \geq d_{0w} - d_{0v},\ z_{vw} \in \mathbb{Z}_+ \text{ for all } w \in W(v),\ s_v \in \mathbb{R}_+,$$

where $W(v) = \{w \in V : d_{0w} - d_{0v} > 0\}$ and $z_{vw} = \sum_{u \in \bar{P}(\mu(v,w),w)} y_u$, is a relaxation of the set $X^{LS-TREE}$ defined by (1)–(3).

Proof: The inequality is obtained by combining the equation $s_{\mu(v,w)} + \sum_{u \in \bar{P}(\mu(v,w),v)} x_u = \sum_{u \in \bar{P}(\mu(v,w),v)} d_u + s_v$ obtained as the sum of the balance constraints (1) along the path $\bar{P}(\mu(v, w), v)$, and the inequality $s_{\mu(v,w)} + \sum_{u \in \bar{P}(\mu(v,w),w)} Cy_u \geq \sum_{u \in \bar{P}(\mu(v,w),w)} d_u$ obtained as the surrogate of the sum of the balance constraints along the path $\bar{P}(\mu(v, w), w)$, together with the non-negativity of x_u and the fact that $\sum_{u \in \bar{P}(\mu(v,w),w)} d_u - \sum_{u \in \bar{P}(\mu(v,w),v)} d_u = d_{0w} - d_{0v}$. □

Starting from the initial balance constraints (1), we can also include any subset of the x_v variables in the continuous variable in order to build other mixing set relaxations.

Proposition 3 *For all $v \in V \cup \{-1\}$ and $U \subseteq V$, the mixing set $X_2^{MIX}(v, U)$:*

$$s_v^U + C\zeta_{vw}^U \geq d_{0w} - d_{0v},\ \zeta_{vw}^U \in \mathbb{Z}_+ \text{ for all } w \in W(v),\ s_v^U \in \mathbb{R}_+,$$

where $s_v^U = s_v + \sum_{u \in U} x_u$ and $\zeta_{vw}^U = \sum_{u \in \bar{P}(\mu(v,w),w) \setminus U} y_u$, is a relaxation of the set $X^{LS-TREE}$.

3 (Q, S_Q) Inequalities

Let $Q \subseteq V$ be such that no two nodes of Q lie on the same path from the root. This defines a unique rooted subtree $T_Q = (V_Q, A_Q)$ having the nodes of Q as leaves and 0 as the root node. We suppose that the $N + 1$ nodes of this tree are (re-)numbered from 0 to N, where 0 corresponds to the root and the leaves following a prefix ordering are numbered $N - K + 1, \ldots, N$, where $|Q| = K$. Note that given a planar representation of the tree, a prefix or infix ordering on the nodes always leads to the same ordering of the leaves. In addition we must satisfy the condition that $d_{0,N-K+1} < \cdots < d_{0,N}$.

For nodes $v \in V_Q$ of the tree T_Q, we use the notation:
$m(v) = \max\{q \in Q \cap V_Q(v)\}$, where $V_Q(v)$ is the subtree of T_Q rooted at v;
$\rho(v) = \max\{q \in Q : q < \min[t \in Q \cap V_Q(v)]\}$.

If $\{q \in Q : q < \min[t \in Q \cap V_Q(v)]\} = \emptyset$, then $\rho(v)$ is undefined and we set $d_{0,\rho(v)} = -\infty$. Note that this happens if and only if $v \in P(0, N - K + 1)$. See Figure 2a), where we assume that the order of the leaves is increasing starting from the top of the tree.

In the rest of this section we will use the following property, that is easily checked: if $u \in P(0, v)$ then $\rho(u) \leq \rho(v)$.

We can now present the (Q, S_Q) inequality. Note that we set $d_{0,p(0)} = d_{0,-1} = 0$.

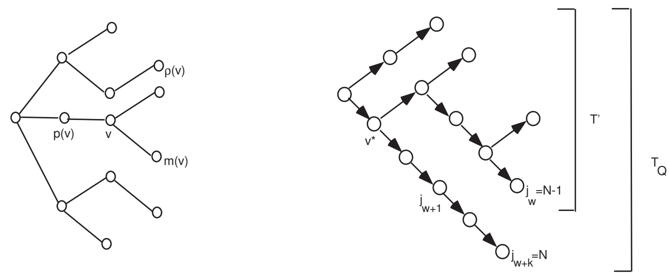

Figure 2: a) Tree indicators. b) Induction step.

Proposition 4 *[5] For any subset $S_Q \subseteq V_Q$, the (Q, S_Q) inequality*

$$s + \sum_{u \in S_Q} x_u + \sum_{u \in V_Q \setminus S_Q} \left(d_{0,m(u)} - \max[d_{0,p(u)}, d_{0,\rho(u)}]\right) y_u \geq d_{0,m(0)} \tag{4}$$

is valid for the uncapacitated problem with $C_v = M$ large for all $v \in V_Q$.

We now show that this inequality is a mixing inequality.

We define $W = \{v \in V_Q : d_{0,v} > d_{0,\rho(v)}\}$. Clearly $Q \cup P(0, n-K+1) \subseteq W$. We define an ordering $W = \{j_0, \ldots, j_w\}$ of the nodes of W by nondecreasing values of d_{0,j_t}. We now explain how this order can be made unique.

Suppose there exist two indices i, k such that $j_i, j_k \in W$ and $d_{0,j_i} = d_{0,j_k}$. We show that then either $j_i \in P(0, j_k)$ or $j_k \in P(0, j_i)$. If neither of the two nodes is a successor of the other then $m(j_i) \neq m(j_k)$. We assume w.l.o.g. $m(j_i) < m(j_k)$. Then $m(j_i) < \min\{v \in Q \cap V_Q(j_k)\}$, which implies $\rho(j_k) \geq m(j_i)$. Then $d_{0,\rho(j_k)} \geq d_{0,m(j_i)} \geq d_{0,j_i} = d_{0,j_k}$ and thus $j_k \notin W$, a contradiction.

Therefore, j_i and j_k lie on the same path from the root, say $j_k \in P(0, j_i)$. Then we assume that the ordering satisfies $k < i$. Note that $j_0 = 0$ and $j_w = m(0)$.

Lemma 5 *Given $v \in V_Q$, $v \in P(0, j_i)$ if and only if $\ell \leq i \leq k$, where $\ell = \min\{t : v \in P(0, j_t)\}$ and $j_k = m(v)$.*

Proof: The proof uses induction on the number $|Q|$ of leaves. The case of $|Q| = 1$ is immediate. Consider now the tree T_Q. Let $v^* = \mu(N-1, N)$. The tree T' obtained from T_Q by removing the path $\bar{P}(v^*, N)$ has $|Q| - 1$ leaves.

We assume the inductive hypothesis for T', and show that it still holds for T_Q.

For all nodes of T' except those on the path $P(0, v^*)$, the values of ρ and m do not change. For nodes $v \in P(0, v^*)$, $m(v) = N - 1$ in T' and $m(v) = N$ in T_Q, but $\rho(v)$ is unchanged. For the new nodes in $\bar{P}(v^*, N)$, $m(v) = N$ and $\rho(v) = N - 1$. In addition the $k \geq 1$ nodes in $\bar{P}(v^*, N)$ for which $d_{0,v} > d_{0,N-1}$ are added to W giving $W = \{j_0, \ldots, j_w, j_{w+1}, \ldots, j_{w+k}\}$, see Figure 2b).

Now it is easily checked that the condition holds for all nodes on the path $P(0, N)$. □

The above proof also shows the following.

Corollary 6 *If $i \neq 0$ then* $j_{i-1} = \begin{cases} p(j_i) & \text{if } d_{0,p(j_i)} > d_{0,\rho(j_i)}, \\ \rho(j_i) & \text{otherwise.} \end{cases}$

We can now prove the main result of this section.

Proposition 7 *For any subset $S_Q \subseteq V_Q$, the (Q, S_Q) inequality (4) is (dominated by) a mixing inequality.*

Proof: Choose $M \geq \max\{d_{0,u} : u \in V\}$, define $W = \{j_0, \ldots, j_w\}$ as above and $r = \max\{i : P(0, j_i) \subseteq S_Q\}$. (If $j_0 = 0 \notin S_Q$, set $r = -1$ and $d_{0,j_r} = d_{0,j_{-1}} = 0$.)

Taking $v = -1$ and $U = S_Q$, Proposition 3 shows that the following inequalities are valid:

$$s + \sum_{u \in S_Q} x_u + M \sum_{u \in P(0,j_i) \setminus S_Q} y_u \geq d_{0,j_i} \quad \text{for } i = r+1, \ldots, w.$$

Then, if we set $\bar{s} = s + \sum_{u \in S_Q} x_u - d_{0,j_r}$ and $z_i = \sum_{u \in P(0,j_i) \setminus S_Q} y_u$, we obtain that the following mixing set relaxation is valid:

$$\bar{s} + M z_i \geq d_{0,j_i} - d_{0,j_r}, \; z_i \in \mathbb{Z}_+ \; \text{ for } i = r+1, \ldots, w, \; \bar{s} \geq 0, \tag{5}$$

where constraint $\bar{s} \geq 0$ follows from the definition of r.

The mixing inequality using all inequalities (5) is (see Corollary 1)

$$\bar{s} \geq \sum_{i=r+1}^{w} (d_{0,j_i} - d_{0,j_{i-1}})(1 - z_i). \tag{6}$$

In the original variables, the inequality is $s + \sum_{u \in S_Q} x_u - d_{0,j_r} \geq \sum_{i=r+1}^{w} (d_{0,j_i} - d_{0,j_{i-1}})\big(1 - \sum_{u \in P(0,j_i) \setminus S_Q} y_u\big)$, or equivalently $s + \sum_{u \in S_Q} x_u + \sum_{u \notin S_Q} \big[\sum_{i:i>r, u \in P(0,j_i)} (d_{0,j_i} - d_{0,j_{i-1}})\big] y_u \geq d_{0,m(0)}$.

We show that this inequality dominates (4). Specifically, we show that for each $u \in V_Q \backslash S_Q$

$$\sum_{i:i>r, u \in P(0,j_i)} (d_{0,j_i} - d_{0,j_{i-1}}) \leq d_{0,m(u)} - \max\{d_{0,p(u)}, d_{0,\rho(u)}\}. \tag{7}$$

If $u = 0$ then the left-hand-side of inequality (7) is $\sum_{i=r+1}^{w} (d_{0,j_i} - d_{0,j_{i-1}}) = d_{0,m(0)} - d_{0,j_r}$ and the right-hand-side is $d_{0,m(0)} - \max\{d_{0,p(0)}, d_{0,\rho(0)}\} = d_{0,m(0)} - \max\{0, -\infty\} = d_{0,m(0)}$, thus inequality (7) holds.

Now assume $u \in V_Q \setminus S_Q$, $u \neq 0$. Define $\ell = \min\{i : u \in P(0, j_i)\}$. Note that $\ell > 0$ as $u \neq 0$. Then by Corollary 6, $d_{0,j_{\ell-1}} = \max\{d_{0,p(j_\ell)}, d_{0,\rho(j_\ell)}\}$. By Lemma 5, $\{i : u \in P(0, j_i)\} = \{\ell, \ldots, k\}$, where $j_k = m(u)$.

This implies that if $\ell > r$ then $\sum_{i:i>r, u \in P(0,j_i)} (d_{0,j_i} - d_{0,j_{i-1}}) = d_{0,m(u)} - d_{0,j_{\ell-1}} = d_{0,m(u)} - \max\{d_{0,p(j_\ell)}, d_{0,\rho(j_\ell)}\} \leq d_{0,m(u)} - \max\{d_{0,p(u)}, d_{0,\rho(u)}\}$, where the inequality follows from the inequalities $d_{0,p(u)} \leq d_{0,p(j_\ell)}$ and $d_{0,\rho(u)} \leq d_{0,\rho(j_\ell)}$, which both hold since $u \in P(0, j_\ell)$. Thus inequality (7) is satisfied if $\ell > r$.

Now suppose $\ell \leq r$. Then $\sum_{i:i>r, u \in P(0,j_i)} (d_{0,j_i} - d_{0,j_{i-1}}) = d_{0,m(u)} - d_{0,j_r} \leq d_{0,m(u)} - d_{0,j_\ell} \leq d_{0,m(u)} - \max\{d_{0,p(j_\ell)}, d_{0,\rho(j_\ell)}\} \leq d_{0,m(u)} - \max\{d_{0,p(u)}, d_{0,\rho(u)}\}$, where the second inequality holds because $d_{0,j_\ell} > d_{0,\rho(j_\ell)}$ as $j_\ell \in W$, and $d_{0,j_\ell} \geq d_{0,p(j_\ell)}$ as $d_{j_\ell} \geq 0$. Thus inequality (7) is also satisfied when $\ell \leq r$ and the proof is complete. □

In the proof of Proposition 7 we have constructed a mixing inequality which dominates a given (Q, S_Q) inequality. In Section 4 we will give an example in which a mixing inequality constructed as above is not implied by the (Q, S_Q) inequalities.

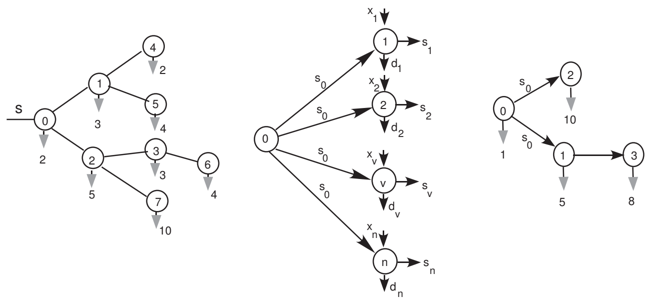

Figure 3: a) (Q, S_Q) tree. b) One period newsboy problem. c) Two period instance.

Example 1 *We consider the instance shown in Figure 3a). The conditions required for a* (Q, S_Q) *inequality hold as* $d_{04} = 7 < d_{05} = 9 < d_{06} = 14 < d_{07} = 17$. *Calculating* $p = (-1, 0, 0, 2, 1, 1, 3, 2)$, $m = (7, 5, 7, 6, 4, 5, 6, 7)$ *and* $\rho = (\emptyset, \emptyset, 5, 5, \emptyset, 4, 5, 6)$, *the* $(Q, \emptyset)$ *inequality is*

$$s + (17 - \max[0, -\infty])y_0 + (9 - \max[2, -\infty])y_1 + (17 - \max[2, 9])y_2 + (14 - \max[7, 9])y_3$$
$$+(7 - \max[5, -\infty])y_4 + (9 - \max[5, 7])y_5 + (14 - \max[10, 9])y_6 + (17 - \max[7, 14])y_7 \geq 17,$$

that is, $s + 17y_0 + 7y_1 + 8y_2 + 5y_3 + 2y_4 + 2y_5 + 4y_6 + 3y_7 \geq 17$.

Now we generate the inequality as a mixing inequality. Note that $3 \in W$ *as* $d_{03} = 10 > d_{0,\rho(3)} = d_{05} = 9$, *and* $0, 1 \in W$ *as* $\rho(0), \rho(1)$ *are undefined. Thus* $W = Q \cup \{0, 1, 3\}$. *The ordering of* W *is* $\{0, 1, 4, 5, 3, 6, 7\}$ *with* $d_{0u} = (2, 5, 7, 9, 10, 14, 17)$, *and the corresponding mixing inequality (6) is the same inequality as above.*

4 Strength of the Mixing Reformulations

Here we consider briefly when the addition of the convex hulls of the mixing sets proposed in Propositions 2 and 3 suffices to give the convex hull of the lot-sizing problem on a tree. On the positive side we see that for a discrete version of the one-period newsboy problem, the convex hull is obtained in the constant capacity case with and without backlogging. On the negative side we show a two-period uncapacitated instance for which the mixing set reformulations are insufficient.

First we examine a variant of the classical newsboy problem, see for instance Ch. 10 in [10]. An initial amount s can be produced at time 0 at a unit cost of h without any fixed cost. Then with probability p_v the vth outcome in period 1 is observed. This consists of the demand d_v, as well as the new production costs involving a unit production cost c_v, a fixed cost q_v per batch of size C and a unit disposal cost of h_v for $v = 1, \ldots, n$. The corresponding lot-sizing tree is shown in Figure 3b).

A mixed integer programming formulation for this discrete newsboy problem is now

$$\min \; hs_0 + \textstyle\sum_{v=1}^{n} p_v(c_v x_v + q_v y_v + h_v s_v)$$
$$s_0 + x_v = d_v + s_v, \; x_v \leq C y_v \text{ for } v = 1, \ldots, n, \; s \in \mathbb{R}_+^{n+1}, x \in \mathbb{R}_+^n, y \in \mathbb{Z}_+^n.$$

After elimination of the variables s_v for $v = 1, \ldots, n$, we obtain the feasible region X^{N1}:

$$s_0 + x_v \geq d_v, \; x_v \leq C y_v \text{ for } v = 1, \ldots, n, \; s_0 \in \mathbb{R}_+, x \in \mathbb{R}_+^n, y \in \mathbb{Z}_+^n.$$

Proposition 8 *[6] In the uncapacitated case, when C is large and $y \in \{0,1\}^n$, $\text{conv}(X^{N1})$ is completely described by the initial constraints and (Q, S_Q) inequalities.*

For the case with constant C, the set X^{N1} has been studied recently by Conforti et al. under the name of mixing set with flows. They show that the initial constraints plus the convex hulls of the $n+1$ mixing set relaxations described in Proposition 2 give a tight extended formulation of the convex hull.

Proposition 9 *[1]*

$$\text{conv}(X^{N1}) = \text{proj}_{(s_0,x,y)} \cap_{v=0}^{n} \text{conv}(X_v^{MIX}) \cap \{(x,y) : 0 \leq x \leq Cy\},$$

where $X_v^{MIX} = \{(s_v, y) \in \mathbb{R}_+ \times \mathbb{Z}_+^n : s_v + Cy_k \geq d_k - d_v$ for all k such that $d_k > d_v\}$ for $v = 0, \ldots, n$, with $d_0 = 0$ and $s_0 + x_v = d_v + s_v$.

If one allows backlogging once the demands are known, the formulation becomes

$$\min\ hs_0 + \textstyle\sum_{v=1}^{n} p_v(c_v x_v + q_v y_v + h_v s_v + b_v r_v)$$
$$s_0 + x_v = d_v + s_v - r_v,\ x_v \leq Cy_v \text{ for } v = 1, \ldots, n,\ s \in \mathbb{R}_+^{n+1}, x, r \in \mathbb{R}_+^n, y \in \mathbb{Z}_+^n,$$

which, after elimination of the variables s_v for $v = 1, \ldots, n$, gives the feasible region

$$s_0 + r_v + x_v \geq d_v,\ x_v \leq Cy_v \text{ for } v = 1, \ldots, n,\ s_0 \in \mathbb{R}_+, r, x \in \mathbb{R}_+^n, y \in \mathbb{Z}_+^n,$$

denoted X^{N2}, known as a continuous mixing set with flows.

Proposition 10 *[2] In the constant capacity case, there is a polynomial size extended formulation for $\text{conv}(X^{N2})$ and a polynomial time optimization algorithm.*

To complete this section we present a small instance showing that mixing inequalities are insufficient to give a complete description of the convex hull of the solution set as soon as there is a second period with random outcomes. The instance is shown in Figure 3c). We take $C = 20$, so the problem is uncapacitated, and we assume that there is no initial stock, i.e. $s = 0$.

Since $d_0 = 1$, $x_0 > 0$ and $y_0 = 1$ in any feasible solution, we can eliminate variables x_0 and y_0 from the model and formulate the instance in terms of variables $s_0, x_1, \ldots, x_3, y_1, \ldots, y_3$.

Two of the fifteen nontrivial facet-defining inequalities are listed below (the complete list can be found in [3]). The first is clearly not a mixing or (Q, S_Q) inequality. We show below that the second one is a mixing inequality but not a (Q, S_Q) inequality.

$$\begin{array}{llllll} s_0 & +\frac{3}{8}x_1 & +\frac{25}{8}y_1 & +5y_2 & +3y_3 & \geq & 13 \\ s_0 & +x_1 & & +5y_2 & +3y_3 & \geq & 13 \end{array}$$

To obtain the second inequality, it suffices to mix the inequalities

$$s_0 + 20y_1 \geq 5,\ s_0 + 20y_2 \geq 10,\ s_0 + 20(y_1 + y_3) \geq 13,\ s_0 \geq 0,$$

corresponding to the paths $P(0,1)$, $P(0,2)$ and $P(0,3)$. The resulting inequality $s_0 \geq 5(1 - y_1) + 5(1 - y_2) + 3(1 - y_1 - y_3)$, is precisely the required inequality. Substituting for s_0, using $x_0 = s_0 + d_0$, the inequality reads $x_0 + 8y_1 + 5y_2 + 3y_3 \geq 14$. If this inequality were a (Q, S_Q) inequality (4), then necessarily $Q = \{2, 3\}$ and $S_Q = \{0\}$. However the corresponding (Q, S_Q) inequality is the facet-defining inequality $x_0 + 3y_1 + 10y_2 + 3y_3 \geq 14$.

5 Computation

Here we report briefly on the effectiveness of the mixing sets in solving the lot-sizing problem on a tree. For a problem with T periods, we take Δ outcomes in each period giving a Δ-ary tree with Δ^{T-1} scenarios (leaves) and a total of $N = \frac{\Delta^T - 1}{\Delta - 1}$ nodes. The data is randomly generated with d_t a random integer in [0,100], h_t a random integer in [1,11], p_t a random integer in [0,20], q_t 25 times a random integer in [0,80], and at each node the Δ random outcomes have equal probability $\frac{1}{\Delta}$ (in other words, the costs at distance k from the root are weighted by Δ^{-k}).

For each value $\Delta \in \{2, 3, 4\}$, we have generated four instances: two with capacity $C = 100$ and two with capacity $C = 500$, that are essentially uncapacitated. For each Δ, we chose T so that the total number of nodes N (which is also the number of binary variables) was close to 1000.

All computations were carried out under IVE version 1.16.00, Mosel version 1.7.8 using Xpress-MP as the mixed integer programming solver, version 16.01.01, running on an IBM Thinkpad with a 1.6GHz Intel Pentium processor.

What strategy to use when possibly combining extended formulations for mixing sets, separation of mixing inequalities, system cuts and branch-and-bound is not at all obvious a priori. Extended formulations lead to improved bounds, but much larger linear programs. Cuts also lead to improved bounds, but the mixing inequalities may be dense. So there is a real trade-off between the strengthening of the bounds and the difficulty in solving the resulting linear programs during the branch-and-bound/cut process.

After some preliminary tests we adopted one strategy for the instances with $\Delta \in \{2, 3\}$ and another for those with $\Delta = 4$. For the instances with $\Delta \in \{2, 3\}$, we start with the formulation (1)–(3). Then, for each non-leaf node $v \in V$, we add a tight extended formulation [9] of the mixing set described in Proposition 2. However, in the construction of the mixing set $W(v)$ is restricted to the nodes in the subtree rooted at v and the distance between v and w is restricted to at most 4. This gives the reformulated mixed integer programming formulation that is fed to the optimizer. We solve the linear program at the top node and add Xpress-MP system cuts conservatively (cutstrategy=1) for ten rounds. Next for another ten rounds we call the mixing inequality separation routine [8, 11] for the same mixing sets as above, but without any restrictions on $W(v)$, and finally we run default branch-and-cut.

For the instances with $\Delta = 4$, we use the extended formulations of the same sets as above, with $W(v)$ restricted to the nodes in the subtree rooted at v, but with no restriction on the path length between v and w. System cuts are then added aggressively (cutstrategy=3) for ten rounds, followed by default branch-and-cut. This strategy was adopted because, although the addition of mixing cuts improves the bounds significantly, the resulting linear programs become significantly slower to solve and thus the overall performance deteriorates.

In Table 1 there are two lines for each instance Δ-seed-[o,r], where seed denotes the random number used to generate the instance, and o,r denote the original and the reformulated problem respectively. The first line gives the results for default Xpress-MP on the original formulation (1)–(3), except that the cut strategy is aggressive (cutstrategy=3). In the second line we give the results for the reformulated instance using the strategy described above. The first column indicates the name of the instance and the next two columns indicate the capacity C and the total number of nodes (binary variables) N. The next two give the number r of rows and the number c of columns of the initial LP matrix. The values LP, $XLP1$, $XLP2$, BLB and BIP indicate the linear programming value of the reformulation, the value $XLP1$

after the addition of the system cuts of Xpress-MP, $XLP2$ the value after the addition of the mixing cuts (if generated), BLB the value of the best lower bound on termination and BIP the value of the best integer solution found. *secs* gives the total run time and *gap*, given by $\frac{BIP-BLB}{BIP} \cdot 100$, is the percentage duality gap on termination.

For each instance 15-20 seconds are required to generate the reformulation including the extended formulations. For each instance we give the results obtained after 300 seconds, or the total run time, excluding matrix generation time, if optimality is proven.

	C	N	r	c	LP	XLP1	XLP2	BLB	BIP	secs	gap
2-5612o	100	1023	2046	3070	9430.2	1119.6		11094.0	11178.8	300	0.76%
2-5612r	100	1023	7147	8171	11119.5	11144.7	11159.6	11160.6	11160.6	49	0
2-4567o	100	1023	2046	3070	9495.8	11552.8		11608.9	11819.2	300	1.78%
2-4567r	100	1023	7143	8167	11706.7	11733.9	11751.7	11755.9	11755.9	116	0
2-1234o	500	1023	2046	3070	5729.7	9880.8		9947.4	10052.6	300	1.05%
2-1234r	500	1023	7149	8173	9893.8	9985.7	1022.5	10033.5	10033.5	53	0
2-7777o	500	1023	2046	3070	5617.9	9448.0		9521.3	9616.4	300	0.99%
2-7777r	500	1023	7145	8169	9467.4	9547.6	9570.5	9583.3	9583.3	148	0
3-1238o	100	1093	2186	3280	6333.2	7752.8		7880.7	8055.7	300	2.17%
3-1238r	100	1093	7263	8357	7923.0	7957.4	7965.2	7972.3	7972.3	121	0
3-1240o	100	1093	2186	3280	7362.6	9212.2		9259.2	9347.1	300	0.94%
3-1240r	100	1093	7263	8357	9303.7	9322.7	9325.3	9327.1	9327.1	115	0
3-1241o	500	1093	2186	3280	3195.7	5722.6		5774.3	5929.9	300	2.62%
3-1241r	500	1093	7266	8360	5808.3	5852.3	5875.1	5885.3	5885.3	258	0
3-1242o	500	1093	2186	3280	3972.7	7012.2		7173.8	7232.6	300	0.81%
3-1242r	500	1093	7268	8362	7180.2	7215.6	7222.1	7232.6	7232.6	174	0
4-2224o	100	1365	2730	4096	6141.0	7769.9		7960.2	8056.4	300	1.19%
4-2224r	100	1365	11142	12508	8007.4	8025.7		8034.1	8034.1	228	0
4-2225o	100	1365	2730	4096	4012.9	4922.1		4964.6	5114.0	300	0.97%
4-2225r	100	1365	11147	12513	5038.5	5063.7		5077.3	5077.3	131	0
4-2222o	500	1365	2730	4096	2975.2	5759.5		5776.2	5961.7	300	3.11%
4-2222r	500	1365	11147	12513	5820.8	5858.9		5877.0	5890.8	300	0.23%
4-2223o	500	1365	2730	4096	3251.3	5896.0		6039.5	6407.0	300	5.74%
4-2223r	500	1365	11145	12511	6167.3	6209.7		6226.3	6348.2	300	1.92%

Table 1: Instances of LS-TREE.

For the twelve instances considered, the system cuts used significantly strengthen the initial LP bounds. When the aggressive option is chosen, they include path inequalities that are well-adapted to such problems. However the mixing inequalities appear necessary if one wishes to prove optimality. Running the two unsolved instances for 900 seconds rather than just 300 leads to a gap of 0.10% for instance 4-2222r and of 1.02% for 4-2223r.

What these very preliminary results suggest is that the combining of extended formulations and cutting planes in tackling other problems is an intriguing research topic.

Acknowledgement This work was partly carried out within the framework of ADONET, a European network in Algorithmic Discrete Optimization, contract no. MRTN-CT-2003-504438. This text 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.

References

[1] M. Conforti, M. Di Summa, L.A. Wolsey, The mixing set with flows, Core DP 2005/92, Université catholique de Louvain, 2005, to appear in SIAM J. Discret. Math.

[2] M. Conforti, M. Di Summa, L.A. Wolsey, The intersection of continuous mixing polyhedra and the continuous mixing polyhedon with flows, 12th IPCO, Springer-Verlag LNCS 4513 (2007), 352–366.

[3] M. Di Summa, L.A. Wolsey, Lot-sizing on a tree, CORE DP 2006/44, Université catholique de Louvain, May 2006.

[4] Y. Guan, S. Ahmed, G.L. Nemhauser, Sequential pairing of mixed integer inequalities, 11th IPCO, Springer-Verlag LNCS 3509 (2005), 23–34.

[5] Y. Guan, S. Ahmed, G.L. Nemhauser, A.J. Miller, A branch-and-cut algorithm for the stochastic uncapacitated lot-sizing problem, Math. Program. 105 (2005), 55–84.

[6] Y. Guan, S. Ahmed, G.L. Nemhauser, A.J. Miller, On formulations of the stochastic uncapacitated lot-sizing problem, Oper. Res. Lett. 34 (2006), 241–250.

[7] Y. Guan, A.J. Miller, Polynomial time algorithms for stochastic uncapacitated lot-sizing problems, `http://www.optimization-online.org/DB_FILE/2006/09/1469.pdf`

[8] O. Günlük, Y. Pochet, Mixing mixed integer inequalities, Math. Program. 90 (2001), 429–457.

[9] A.J. Miller, L.A. Wolsey, Tight formulations for some simple MIPs and convex objective IPs, Math. Program. B 98 (2003), 73–88.

[10] E.A. Silver, R. Petersen, D.F. Pyke, Inventory Management and Production Planning and Scheduling, Wiley, New York 1998.

[11] Y. Pochet, L.A. Wolsey, Production Planning by Mixed Integer Programming, Springer, New York 2006.