HAL
open science

A tight MILP formulation based on multi-product valid inequalities for a lot-sizing problem

Céline Gicquel, Michel Minoux

To cite this version:

Céline Gicquel, Michel Minoux. A tight MILP formulation based on multi-product valid inequalities for a lot-sizing problem. International Conference on Industrial Engineering and Systems Management, Oct 2013, Rabat, Morocco. pp.778. hal-00948129

HAL Id: hal-00948129

https://hal.science/hal-00948129v1

Submitted on 20 Feb 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A tight MILP formulation based on multi-product valid inequalities for a lot-sizing problem

Céline Gicquel
Université Paris Sud
Laboratoire de Recherche en Informatique
Campus d'Orsay, F-91405 Orsay
Email: celine.gicquel@lri.fr

Michel Minoux
Université Piere et Marie Curie
Laboratoire d'Informatique de Paris 6
4 place Jussieu, F-75005 Paris
Email: michel.minoux@lip6.fr

February 17, 2014

Abstract

We consider a problem arising in the context of industrial production planning, namely the multi-product discrete lot-sizing and scheduling problem with sequence-dependent changeover costs. We aim at developping an exact solution approach based on a standard Branch & Bound procedure for this combinatorial optimization problem. To achieve this, we propose a new family of multi-product valid inequalities which enables us to better take into account in the mixed integer linear programming formulation the conflicts between different products simultaneously requiring production on the resource. We then present both an exact and a heuristic separation algorithm in order to identify the most violated valid inequalities to be added in the initial MILP formulation within a cutting-plane generation algorithm. We finally discuss preliminary computational results which confirm the practical usefulness of the proposed valid inequalities at strengthening the MILP formulation and at reducing the overall computation time.

1 Introduction

Capacitated lot-sizing arises in industrial production planning whenever changeover operations such as preheating, tool changing or cleaning are required between production runs of different products on a machine. The amount of the

related changeover costs usually does not depend on the number of products processed after the changeover. Thus, to minimize changeover costs, production should be run using large lot sizes. However, this generates inventory holding costs as the production cannot be synchronized with the actual demand pattern: products must be held in inventory between the time they are produced and the time they are used to satisfy customer demand. The objective of lot-sizing is thus to reach the best possible trade-off between changeover and inventory holding costs while taking into account both the customer demand satisfaction and the technical limitations of the production system.

An early attempt at modelling this trade-off can be found in [15] for the problem of planning production for a single product on a single resource with an unlimited production capacity. Since this seminal work, a large part of the research on lot-sizing problems has focused on modelling operational aspects in more detail to answer the growing industry need to solve more realistic and complex production planning problems. An overview of recent developments in the field of modelling industrial extensions of lot-sizing problems is provided in [10].

In the present paper, we focus on one of the variants of lot-sizing problems mentioned in [10], namely the multi-product single-resource discrete lot-sizing and scheduling problem or DLSP. As defined in [6] and [10], several key assumptions are used in the DLSP to model the production planning problem:
- A set of products is to be produced on a single capacitated production resource.
- A finite time horizon subdivided into discrete periods is used to plan production.
- Demand for products is time-varying (i.e. dynamic) and deterministically known.
- At most one product can be produced per period (small bucket model) and the facility processes either one product at full capacity or is completely idle (discrete or all-or-nothing production policy).
- Costs to be minimized are the inventory holding costs and the changeover costs.

In the DLSP, it is assumed that a changeover between two production runs for different products results in a changeover cost. Changeover costs can depend either on the next product only (sequence-independent case) or on the sequence of products (sequence-dependent case). We consider in the present paper the DLSP with sequence-dependent changeover costs (denoted DLSPSD in what follows). Sequence-dependent changeover costs are mentioned in [10] as one of the relevant operational aspects to be incorporated into lot-sizing models. Moreover, a significant number of real-life lot-sizing problems involving sequence-dependent changeover costs have been recently reported in the academic literature: see among others [3] for an injection moulding process, [13] for a textile fibre industry or [5] for soft drink production.
A wide variety of solution techniques from the Operations Research field have been proposed to solve lot-sizing problems: the reader is referred to [2] and [9] for recent reviews on the corresponding literature. The present paper belongs

to the line of research dealing with exact solution approaches, i.e. aiming at providing guaranteed optimal solutions for the problem. A large amount of existing solution techniques in this area consists in formulating the problem as a mixed-integer linear program (MILP) and in relying on a Branch & Bound type procedure to solve the obtained MILP. However the computational efficiency of such a procedure strongly depends on the quality of the lower bounds used to evaluate the nodes of the search tree. In the present paper, we seek to improve the quality of these lower bounds so as to decrease the total computation time needed to obtain guaranteed optimal solutions for medium-size instances of the problem. Within the last thirty years, much research has been devoted to the polyhedral study of lot-sizing problems in order to obtain tight linear relaxations and improve the corresponding lower bounds: see e.g. [11] for a general overview of the related literature and [1, 4, 7, 14] for contributions focusing specifically on the DLSP. However, these procedures mainly focus on the underlying single-product subproblems and thus fail at capturing the conflicts between multiple products sharing the same resource capacity. This leads in some cases to significant integrality gaps for multi-product instances of the DLSPSD. In what follows, we propose a new family of multi-product valid inequalities which enables us to partially remedy this difficulty and discuss both an exact and a heuristic algorithm to solve the corresponding separation problem. To the best of our knowledge, this is one of the first attempts focusing on improving the polyhedral description of multi-product lot-sizing problems.

The main contributions of the present paper are thus twofold. First we introduce a new family of valid inequalities representing conflicts on multi-period time intervals between several products simultaneously requiring production on the available resource. Second we formulate the corresponding separation problem as a quadratic binary program and propose to solve it either exactly by relying on a quadratic programming solver or approximately through a Kernighan-Lin type heuristic algorithm. The results of the preliminary computational results carried out on medium-size instances show that the proposed valid inequalities are efficient at strengthening the linear relaxation of the problem and at decreasing the overall computation time needed to obtain guaranteed optimal solutions of the DLSPSD.

The remainder of the paper is organized as follows. In Section 2, we recall the initial MILP formulation of the multi-product DSLPSD as well as the previously published valid inequalities for the underlying single-product subproblems. We then present in Section 3 the proposed new multi-product valid inequalities and discuss in Section 4 both an exact and a heuristic algorithm to solve the corresponding separation problem. Preliminary computational results are discussed in Section 5.

2 MILP formulation of the DLSPSD

In this section, we first recall the initial MILP formulation of the DLSPSD. We use the network flow representation of changeovers between products, which

was proposed among others by [1], as this leads to a tighter linear relaxation of the problem. We then discuss the valid inequalities first proposed by [14] to strengthen the underlying single-product subproblems.

2.1 Initial MILP formulation

We wish to plan production for a set of products denoted $p = 1...P$ to be processed on a single production machine over a planning horizon involving $t = 1...T$ periods. Product $p = 0$ represents the idle state of the machine and period $t = 0$ is used to describe the initial state of the production system.

Production capacity is assumed to be constant throughout the planning horizon. We can thus w.l.o.g. normalize the production capacity to one unit per period and express the demands as integer numbers of production capacity units (see [6] and [8]). We denote d_{pt} the demand for product p in period t, h_p the inventory holding cost per unit per period for product p and S_{pq} the sequence-dependent changeover cost to be incurred whenever the resource setup state is changed from product p to product q.

Using this notation, the DLSPSD can be seen as the problem of assigning at most one product to each period of the planning horizon while ensuring demand satisfaction and minimizing both inventory and changeover costs. We thus introduce the following binary decision variables:
- y_{pt} where $y_{pt} = 1$ if product p is assigned to period t, 0 otherwise.
- w_{pqt} where $w_{pqt} = 1$ if there is a changeover from product p to product q at the beginning of t, 0 otherwise.

This leads to the MILP following formulation denoted DLSPSD0 for the problem.

$$
\begin{array}{lll}
Z_{DLSPSD} = min & \sum_{p=1}^{P}\sum_{t=1}^{T} h_p \sum_{\tau=1}^{t}(y_{p\tau} - d_{p\tau}) & \\
 & + \sum_{p,q=0}^{P} S_{p,q} \sum_{t=1}^{T-1} w_{p,q,t} & \\
 & \sum_{\tau=1}^{t} y_{p\tau} \geq \sum_{\tau=1}^{t} d_{p\tau} & \forall p, \forall t \\
 & \sum_{p=0}^{P} y_{pt} = 1, & \forall t \\
 & y_{p,t} = \sum_{q=0}^{P} w_{q,p,t} & \forall p, \forall t \\
 & y_{p,t} = \sum_{q=0}^{P} w_{p,q,t+1} & \forall p, \forall t \\
 & y_{pt} \in \{0,1\} & \forall p, \forall t \\
 & w_{p,q,t} \in \{0,1\} & \forall p, \forall q, \forall t
\end{array}
$$

The objective function (2.1) corresponds to the minimization of the inventory holding and changeover costs over the planning horizon. $\sum_{\tau=1}^{t}(y_{p\tau} - d_{p\tau})$ is the inventory level of product p at the end of period t. Constraints (2.1) impose that the cumulated demand over interval $[1, t]$ is satisfied by the cumulated production over the same time interval. Constraints (2.1) ensure that, in each period, the resource is either producing a single product or idle. Constraints (2.1)-(2.1) link setup variables y_{pt} with changeover variables w_{pqt} through equalities which can be seen as flow conservation constraints in a network (see e.g. [1]). They ensure that in case product p is setup in period t, there is a changeover from another product q (possible $q = p$) to product p to at the beginning of period t and a changeover from product p to another product q (possible $q = p$) at the

end of period t.

2.2 Single-product valid inequalities

We now recall the expression of the valid inequalities proposed by [14] for the single product DLSP. We denote $d_{p,t,\tau}$ the cumulated demand for product p in the interval $\{t, ..., \tau\}$ and $\Delta_{p,v}$ the vth positive demand period for product p. $\Delta_{p,d_{p,1,t}+v}$ is thus the period with the vth positive unit demand for product p after period t occurs.

$$\sum_{\tau=1}^{t}(y_{p\tau} - d_{p\tau}) \geq w - \sum_{v=1}^{w}\Big[y_{p,t+v} + \sum_{\tau=t+v+1}^{\Delta_{p,d_{p,1,t}+v}} \sum_{q\neq p} w_{q,p,\tau}\Big] \quad \forall p, \forall t, \forall w \in [1, d_{p,t+1,T}]$$

Constraints (2.2) are valid inequalities for the DLSPSD. The underlying idea is to compute a lower bound on the inventory level of a product p at the end of a period t ($\sum_{\tau=1}^{t}(y_{p\tau} - d_{p\tau})$) by considering both the demands and the resource setup states for this product in the forthcoming periods $\tau = t+1...\Delta_{p,d_{p,1,t}+v}$. The reader is refered to [14] for a full proof of validity for these inequalities. In the computational experiments to be presented in Section 5, we use a standard cutting-plane generation algorithm to strengthen the formulation DLSPSD0 by adding violated valid inequalities of family (2.2). The resulting improved formulation is denoted DLSPSD1.

Constraints (2.2) can be understood as a way to strengthen the demand satisfaction constraints (2.1) by expressing in a more detailed way the need for each individual product to access the resource in order to satisfy its own demand on a given subinterval of the planning horizon. However, in the resulting DLSPSD1 formulation, the conflicts between different products simultaneously requiring production on the resource are only handled by the single-period capacity constraints (2.1). In what follows, we propose to improve this representation of the conflicts between different products by considering multi-period multi-product valid inequalities.

3 Multi-product valid inequalities for the DLSPSD

We now present the multi-period multi-product valid inequalities proposed to strengthen the linear relaxation of the multi-product DLSPSD.

Proposition 1

Let $SP \subset \{0...P\}$ and $SD \subset \{0...P\}$ be two disjoint subsets of products.
Let $t \in [1, T]$ be a period within the planning horizon and $[1, \theta] \subset [1, T]$ be a time interval including period t.
The following inequality is valid for the multi-product DLSPSD.

$$\Big[\sum_{q\in SD} d_{q,1,\theta}\Big]\Big[\sum_{p\in SP} y_{pt}\Big] \leq \sum_{\tau=1}^{\theta} \tilde{C}_\tau \tag{1}$$

where $\tilde{C}_\tau$ is defined by:
$\tilde{C}_\tau = min(\sum_{q \in SD} y_{q,\tau}, \sum_{p \in SP} y_{pt}), \tau \in [1, t-2] \cup [t+2, \theta]$
$\tilde{C}_{t-1} = \sum_{p \in SP, q \in SD} w_{q,p,t}$
$\tilde{C}_t = 0$
$\tilde{C}_{t-1} = \sum_{p \in SP, q \in SD} w_{p,q,t+1}$

Before providing the proof for Proposition 1, we briefly explain the idea underlying valid inequalities (1). We choose a subset SP of products. If none of these products is assigned for production in period t (i.e. $\sum_{p \in SP} y_{pt} = 0$), all corresponding valid inequalities are trivially respected. But if one of these products is assigned for production in period t (i.e. $\sum_{p \in SP} y_{pt} = 1$), then we have to make sure that we are able to satisfy the total cumulated demand over the interval $[1, \theta]$ for the products in subset SD (i.e to sastify $\sum_{q \in SD} d_{q,1,\theta}$) on the remaining periods $1...t-1, t+1...\theta$. In this case, the right hand side of inequalities (1) computes a tight upper bound of the production capacity available over these periods for the products in SD.

Let (y, w) be a feasible solution of the DLSPSD. We arbitrarily choose a period t, an interval $[1, \theta]$ including t and two disjoint subsets of products SP and SD and show that all proposed inequalities (1) are valid for the considered feasible solution.

We distinguish two main cases:

Case 1: $\sum_{p \in SP} y_{pt} = 0$

In this case, the left hand side of the inequality is equal to 0 whereas the right hand side is nonnegative. Inequality (1) is thus trivially true.

Case 2: $\sum_{p \in SP} y_{pt} = 1$

In this case, the left hand side of inequality (1) is equal to the total cumulated demand over interval $[1, \theta]$ for the products belonging to SD, i.e. to $\sum_{q \in SD} d_{q,1,\theta}$.

$\sum_{p \in SP} y_{pt} = 1$ means that period t is devoted to the production of one of the products in SP and thus cannot be used to satisfy the cumulated demand for products in SD. Hence (y, w) can be a feasible solution of the DLSPSD if and only if the total cumulated production for products in SD over the remaining periods $1...t-1, t+1...\theta$ is sufficient to satisfy the cumulated demand $\sum_{q \in SD} d_{q,1,\theta}$.

We now seek to compute a tight upper bound for the production capacity C_τ available in each period $\tau \in [1, t-1] \cap [t+1, \theta]$ for the products in SD:
- By capacity constraints (2.1), we have $C_\tau \leq 1$, i.e. $C_\tau \leq \sum_{p \in SP} y_{pt}$.
- Moreover, demands for products in SD can only be satisfied by a production for these products, i. τ can be used to satisfy part of demand $\sum_{q \in SD} d_{q,1,\theta}$ only if the resource is setup for one of these products in period τ. This gives $C_\tau \leq \sum_{q \in SD} y_{q,\tau}$.

We thus obtain:

$$C_\tau \leq min(\sum_{q \in SD} y_{q,\tau}, \sum_{p \in SP} y_{pt}), \forall \tau \in [1, \theta], \tau \neq t \tag{2}$$

This leads to the following valid inequality stating that, in a feasible solution

(y, w) of the DSLSP, the cumulated capacity available for products in SD over periods $[1, t-1] \cap [t+1, \theta]$ should be large enough to produce the corresponding cumulated demand:

$$\sum_{q \in SD} d_{q,1,\theta} \leq \sum_{\tau \in [1,\theta], \tau \neq t} \Big[min(\sum_{q \in SD} y_{q,\tau}, \sum_{p \in SP} y_{pt}) \Big] \tag{3}$$

Now, we can exploit our knowledge of the setup state of the resource in period t to further strengthen this inequality. Namely, we know that a product p belonging to SP is produced in period t. A changeover to (resp. from) this product p thus has to take place at the beginning (resp. at the end) of period t. This means that:

- For $t \neq 1$, if a product q belonging to SD is produced in period $t-1$, there must be a changeover from this product $q \in SD$ to the product $p \in SP$ at the beginning of period t. The production capacity available in period $\tau = t-1$ for the products in SD is thus limited by $C_{t-1} \leq \sum_{p \in SP, q \in SD} w_{q,p,t} \leq min(\sum_{q \in SD} y_{q,t-1}, \sum_{p \in SP} y_{pt})$.
- For $t \neq \theta$, if a product q belonging to SD is produced in period $t+1$, there must be a changeover to this product $q \in SD$ from the product $p \in SP$ at the end of period t. The production capacity available in period $\tau = t+1$ for the products in SD is thus limited by $C_{t+1} \leq \sum_{p \in SP, q \in SD} w_{p,q,t+1} \leq min(\sum_{q \in SD} y_{q,t+1}, \sum_{p \in SP} y_{pt})$.

Depending on the value of t, we can thus replace in the right hand side of (3) the term for $\tau = t-1$ (resp. $\tau = t+1$) by $\sum_{p \in SP, q \in SD} w_{q,p,t}$ (resp. $\sum_{p \in SP, q \in SD} w_{p,q,t+1}$), which shows the validity of inequalities (1) for the DLSPSD.

We point out here that, for any integer feasible solution of the DLSPSD, in case $\sum_{p \in SP} y_{pt} = 1$, we have:

$$\begin{array}{ll} \sum_{j \in SD} y_{q,\tau} \leq \sum_{p \in SP} y_{pt}, & \forall \tau \in [1, \theta], \tau \neq t \\ \sum_{p \in SP, q \in SD} w_{q,p,t} = \sum_{q \in SD} y_{q,t-1} & \text{if } t \neq 1 \\ \sum_{p \in SP, q \in SD} w_{p,q,t+1} = \sum_{q \in SD} y_{q,t+1} & \text{if } t \neq \theta \end{array}$$

We will thus have $C_\tau = \sum_{q \in SD} y_{q,\tau}, \forall \tau \in [1, t-1] \cap [t+1, \theta]$ in any integer feasible solution of the problem. However, in a fractional solution obtained by solving the linear relaxation of formulation DLSPSD, we may encounter situations where $0 < \sum_{p \in SP} y_{pt} < 1$ so that we may have $\sum_{p \in SP} y_{pt} \leq \sum_{q \in SD} y_{q,\tau}$, $\sum_{p \in SPnq \in SD} w_{q,p,t} \leq \sum_{q \in SD} y_{q,t-1}$ and $\sum_{p \in SP, q \in SD} w_{p,q,t+1} \leq \sum_{q \in SD} y_{q,t+1}$. In these cases, it is interesting to have the flexibility to select for each period τ the smallest upper bound for the available production capacity C_τ as this will lead to tighter valid inequalities.

4 Exact and heuristic algorithms for solving the separation problem

The number of valid inequalities (1) grows very fast with the problem size. Namely, we have a series of valid inequalities for the $\frac{(T+1)T}{2}$ pairs of periods

(t, θ) with $t \leq \theta$. Moreover, for a given pair of periods (t, θ), the number of available valid inequalities is given by $3Part(P+1, 3)$. Here $Part(P+1, 3)$ is the number of partitions of a set of $P+1$ elements into 3 subsets (SP, SD and $\{0...P\} \setminus (SD \cap SP)$) and can be computed by a mathematical induction. Thus, for an instance involving e.g. $P = 10$ products and $T = 25$ periods, we have $0.5 * 26 * 25 * 3 * 28501 = 27788475$ valid inequalities.

It it therefore not possible to include them *a priori* in the MILP formulation of the problem. This is why we use a cutting-plane generation strategy to add to the MILP formulation only the most violated valid inequalities of the family. This requires solving the corresponding separation algorithm which, given a fractional solution $(\overline{y}, \overline{w})$ of the DLSPSD, will either identify a violated valid inequality or prove that no such inequality exists.

4.1 Exact separation algorithm

We first discuss an exact separation algorithm, i.e. an algorithm which is guaranteed to find an inequality violated by the fractional solution $(\overline{y}, \overline{w})$ if one exists. We consider each possible pair of periods (t, θ) and look for the partition of $\{0...P\}$ into 3 subsets which provides the largest violation of inequalities (1). To achieve this, we formulate the separation problem for a given (t, θ) as follows.

We first introduce the following decision variables:
- $\alpha_p = 1$ if product p belongs to subset SP, 0 otherwise.
- $\beta_p = 1$ if product p belons to subset SD, 0 otherwise.
- $\gamma_\tau = 1$ if capacity C_τ is limited by $\sum_{p=0}^{P} \overline{y_{pt}} \alpha_p$, 0 if C_τ is limited by $\sum_{q=0}^{P} \overline{y_{q\tau}} \beta_q$.

With this notation, the separation problem for a given (t, θ) and a solution $(\overline{y}, \overline{w})$ is formulated as the following quadratic binary program $QBP_{t,\theta}$:

$$
\begin{array}{lll}
max & \sum_{p=0}^{P} \sum_{q=0}^{P} \left[d_{q,1,\theta} - \overline{w_{q,p,t}} - \overline{w_{p,q,t+1}} \right] \alpha_p \beta_q & \\
& - \sum_{\tau=1...t-2} \left[\sum_{p=0}^{P} \overline{y_{pt}} \alpha_p \gamma_\tau + \sum_{q=0}^{P} \overline{y_{q,\tau}} \beta_q (1 - \gamma_\tau) \right] & \\
& - \sum_{\tau=t+2...\theta} \left[\sum_{p=0}^{P} \overline{y_{pt}} \alpha_p \gamma_\tau + \sum_{q=0}^{P} \overline{y_{q,\tau}} \beta_q (1 - \gamma_\tau) \right] & \\
& \alpha_p + \beta_p \leq 1 & \forall p = 0...P \\
& \alpha_p \in \{0, 1\}, \beta_p \in \{0, 1\} & \forall p = 0...P \\
& \gamma_\tau \in \{0, 1\} & \forall \tau = 1...T
\end{array}
$$

The objective function (4.1) corresponds to the maximimization of the violation of the inequalities, i.e. we seek to identify the subsets SP and SD for which the difference between the left hand side and the right hand side of the inequality takes the largest value. If this value is strictly positive, we obtain a violated valid inequality corresponding to (t, θ, SP, SD). In case this value is less than or equal to 0, it means that all valid inequalities for (t, θ) are satisfied by the fractional solution $(\overline{y}, \overline{w})$. Constraints (4.1) state that a given product p cannot be simultaneously included in subsets SP and SD.

Problem $QBP_{t,\theta}$ is a binary program with a quadratic objective function and a series of linear constraints. It can be solved to optimality by a mixed-integer quadratic programming solver such as the one embedded in CPLEX 12.5.

4.2 Heuristic separation algorithm

As can be seen from the computational experiments to be presented in Section 5, solving to optimality a sequence of quadratic binary programs $QBP_{t,\theta}$ leads to prohibitively long computation times for the cutting-plane generation algorithm, even for small-size instances. This is why we propose in what follows a Kernighan-Lin type heuristic which enables us to more quickly identify a violated valid inequality for a given pair (t, θ).

Start with a tripartition of $\{0...P\}$, Π_{ref}, and compute its violation V_{ref}.
While (test =0):
    Let $test = 1$, $PossMove = P + 1$ and $\Pi_{cur} = \Pi_{ref}$.
    Allow all possible moves to explore the neighbourhood of Π_{cur}.
    While ($PossMove > 0$):
        Consider the partitions obtained by carrying out all allowed moves in the neighbourhood of Π_{cur} and evaluate the violation for each obtained partition.
        Select the best partition obtained in this neighbourhood of Π_{cur}, Π_{best}, forbid the move used to obtain Π_{best} from Π_{cur}, decrease $PossMove$ by 1 and set $\Pi_{cur} = \Pi_{best}$.
        If $V_{best} > V_{ref}$, $test = 0$ and $\Pi_{ref} = \Pi_{best}$.

The neighbourhood of a tripartition Π of $\{0...P\}$ is defined as the set of tripartitions obtained by moving a single product from its current subset in Π to one of the two other subsets. Moreover, in the computational experiments to be presented in Section 5, five different types of partitions are used to initialize the heuristic.

4.3 Cutting-plane generation algorithm

We now briefly describe the cutting-plane generation used to strengthen formulation DLSPSD1 by adding to it some multi-product valid inequalities (1).

Compute the initial LP relaxation of the DLSPSD using formulation DLSPSD1.
While ($test = 0$):
    Denote $(\overline{y}, \overline{w})$ the solution of the current linear relaxation.
    For t=1...T such that $\exists p$ such that $0.0001 < \overline{y_{pt}} < 0.9999$;
        Let $\theta = t$ and $found$ =0.
        While ($\theta \leq T$) and ($found == 0$),
            Solve the separation problem for periods (t, θ) using either the exact algorithm or the heuristic algorithm with one of the 5 predefined initial partitions.
            If a violated valid inequality has been found, let $found = 1$.
            $\theta = \theta + 1$.
    If at least one violated valid inequality is found, add all the found violated valid inequalities to the current formulation and compute its LP relaxation.
    Else set $test = 1$ to stop the cutting-plane generation.

5 Preliminary computational results

We now discuss the results of some preliminary computational experiments carried out to evaluate the effectiveness of the proposed multi-product valid inequalities at strengthening the formulation of the multi-product DLSPSD and to assess their impact on the total computation time.
We randomly generated instances of the problem using a procedure similar to the one described in [12] for the DLSP with sequence-dependent change-over costs and times. More precisely, the various instances tested have the following characteristics:
- *Problem dimension.* The problem dimension is represented by the number of products P and the number of periods T: we solved medium-size instances involving 4 to 10 products and 15 to 75 periods.
- *Inventory holding costs.* For each product, inventory holding costs have been randomly generated from a discrete uniform $DU(5, 10)$ distribution.
- *Changeover costs.* We used two different types of structure for the changeover cost matrix S. Instances of sets A1-A7 have a general cost structure: the cost of a changeover from product p to product q, S_{pq}, was randomly generated from a discrete uniform $DU(100, 200)$ distribution. Instances of sets B1-B7 correspond to the frequently encountered case where products can be grouped into product families: there is a high changeover cost between products of different families and a smaller changeover cost between products belonging to the same family. In this case, for products p and q belonging to different product families, S_{pq} was randomly generated from a discrete uniform $DU(100, 200)$ distribution; for products p and q belonging to the same product family, S_{pq} was randomly generated from a discrete uniform $DU(0, 100)$ distribution.
- *Production capacity utilization.* Production capacity utilization ρ is defined as

the ratio between the total cumulated demand ($\sum_{p=1}^{P}\sum_{t=1}^{T} d_{pt}$) and the total cumulated available capacity (T). We set $\rho = 0.95$ for all instances.
- *Demand pattern.* Binary demands $d_{pt} \in \{0, 1\}$ for each product have been randomly generated according to the following procedure:

> 1. We randomly select a product p^* from a discrete uniform $DU(1, N)$ distribution and set $d_{p^*T} = 1$.
> 2. For each product p, except product p^*, we randomly select a period t_p from a discrete uniform $DU(1, T)$ distribution and set $d_{p,t_p} = 1$.
> 3. For each entry in a $P \times T$ matrix, except for the entries corresponding to the (p, t) combinations for which we set $d_{pt} > 0$ in steps 1 or 2, we randomly generate a number α_{pt} from a discrete uniform $DU(1, PT)$ distribution.
> 4. While the total cumulated demand ($\sum_{p=1}^{P}\sum_{t=1}^{T} d_{pt}$) does not exceed ρT, we consider the entries (p, t) one by one in the increasing order of the corresponding value α_{pt} and set $d_{pt} = 1$.
> 5. When the total cumulated demand reaches ρT, we examine whether the corresponding instance is feasible by checking that $\sum_{p=1}^{P}\sum_{\tau=1}^{t} d_{p\tau} \leq t$ for all t. If the instance is infeasible, we repeat steps 1 to 4.

For each considered problem dimension, we generated 10 instances, leading to a total of 140 instances.

All tests were run on an Intel Core i5 (2.7 GHz) with 4 Go of RAM, running under Windows 7. We used a standard MILP software (CPLEX 12.5) with the solver default settings to solve the problems with one of the following formulations:
- DLPSD1: initial MILP formulation DLSPSD0, i.e. formulation (2.1)-(2.1), strengthened by single-product valid inequalities (2.2). We used a standard cutting-plane generation strategy based on a complete enumeration of all possible valid inequalities to add them into the formulation.
- DLSPSD2e: formulation DLSPSD1 strengthened by multi-product valid inequalities (1). We used the cutting-plane generation algorithm presented in Section 4.3 to add only the most violated valid inequalities and relied on the exact separation algorithm discussed in Section 4.1.
- DLSPSD2h: formulation DLSPSD1 strengthened by multi-product valid inequalities (1). We used the cutting-plane generation algorithm presented in Section 4.3 to add only the most violated valid inequalities and relied on the heuristic separation algorithm discussed in Section 4.2.

Tables 1 and 2 display the computational results. We provide for each set of 10 instances:
- P and T: the number of products and planning periods involved in the production planning problem.
- V and Cst : the number of variables and constraints in the initial formulation DLSPSD0.
- SP: the number of single-product violated valid inequalities (2.2) added in

Table 1: Preliminary computational results: exact separation algorithm

						DLSPSD1			DLSPSD2e			
	P	T	V	Cst	SP	Gap_{LP1}	N_{IP1}	T_{IP1}	MPe	Gap_{LP2e}	N_{IP2e}	T_{IP2e}
A1	4	15	425	250	106	2.6%	2	0.3s	9	0.0%	0	11.6s
A2	6	15	840	315	108	0.9%	0	0.3s	4	0.2%	0	13.5s
A3	4	20	600	300	193	2.6%	5	0.4s	15	0.2%	0	445.4s
B1	4	15	425	250	105	11.5%	6	0.3s	16	0.1%	0	19.5s
B2	6	15	840	315	107	5.3%	1	0.3s	12	2.1%	0	60.1s
B3	4	20	600	300	192	8.3%	9	0.5s	24	0.4%	2	698.7s

the three formulations.
- MPe and MPh: the number of multi-product violated valid inequalities added in formulation DLSPSD2e by the exact separation algorithm and in formulation DLSPSD2h by the heuristic separation algorithm.
- Gap_{LP1} (resp. Gap_{LP2e}, Gap_{LP2h}): the average percentage gap between the linear relaxation of formulation DLSPSD1 (resp. DLSPSD2e, DLSPSD2h) and the value of an optimal integer solution.
- N_{IP1} (resp. N_{IP2e}, N_{IP2h}): the average number of nodes explored by the Branch & Bound procedure before a guaranteed optimal integer solution is found or the computation time limit of 2700s is reached.
- T_{IP1} (resp. T_{IP2e}, T_{IP2h}): the total computation time (cutting-plane generation and Branch & Bound search) needed to find a guaranteed optimal integer solution (we used the value of 2700s in case a guaranteed optimal integer solution could not be found within the computation time limit).

Results from Table 1 show that the proposed valid inequalities (1) are efficient at strengthening formulation DLSPSD1. Namely, the integrality gap is reduced from an average of 5.3% with formulation DLSPSD1 (see Gap_{LP1}) to an average of 0.5% with formulation DLSPSD2e (see Gap_{LP2e}). We note that this reduction is particularly significant for instances B1-B3 featuring a product family changeover cost structure. Moreover this formulation strengthening is obtained thanks to a relatively small number of multi-product inequalities as can be seen from the average value of MPe (13). However, even if the number of nodes needed by the Branch & Bound procedure to find a guaranteed optimal solution is slightly reduced when using formulation DLSPSD2e, it does not lead to an overall reduction of the computation time. This is mainly explained by the fact that the cutting-plane generation algorithm based on an exact separation algorithm requires prohibitively long computation times to identify the violated multi-product valid inequalities to be added to the formulation. It is thus necessary to resort to a heuristic separation algorithm such as the one proposed in Section 4.2.

Table 2: Preliminary computational results: heuristic separation algorithm

						DLSPSD1			DLSPSD2h			
	P	T	V	Cst	SP	Gap_{LP1}	N_{IP1}	T_{IP1}	MPh	Gap_{LP2h}	N_{IP2h}	T_{IP2h}
A1	4	15	425	250	106	2.6%	2	0.3s	9	0.0%	0	0.1s
A2	6	15	840	315	108	0.9%	0	0.3s	3	0.2%	0	0.2s
A3	4	20	600	300	193	2.6%	5	0.4s	15	0.2%	0	0.3s
A4	6	25	1400	625	315	4.3%	9	1.0s	27	0.7%	4	1.0s
A5	6	50	2800	1050	1153	1.6%	32	6.7s	20	0.9%	11	4.7s
A6	10	50	6600	1650	1949	2.1%	99	21.0s	51	1.1%	30	22.7s
A7	8	75	6750	2025	2776	2.7%	856	151.9s	23	2.5%	660	147.5s
B1	4	15	425	250	105	11.5%	6	0.3s	16	0.1%	0	0.1s
B2	6	15	840	315	107	5.3%	1	0.3s	10	2.1%	1	0.3s
B3	4	20	600	300	192	8.3%	9	0.5s	21	0.4%	0	0.4s
B4	6	25	1400	625	307	9.2%	13	1.2s	30	0.8%	1	0.7s
B5	6	50	2800	1050	1248	12.2%	1753	47.7s	48	9.5%	983	37.6s
B6	10	50	6600	1650	1274	15.7%	25937	901.0s	97	11.9%	11284	496.0s
B7	8	75	6750	2015	2681	15.3%	25015	1961.9s	53	10.7%	22323	1904.7.0s

Comparison of the results obtained with the exact and the heuristic separation algorithm for the instances A1-A3 and B1-B3 (Tables 1 and 2) shows that the proposed heuristic is very efficient at finding violated valid inequalities for small size instances. Namely, the average integrality gap for these 60 instances when using the heuristic algorithm is the same as the one when using the exact algorithm (i.e. $Gap_{LP2e} = Gap_{LP2h}$). Moreover, the number of violated valid inequalities found by the heuristic algorithm is nearly the same as the number of violated valid inequalities found by the exact algorithm.

Results from Table 2 also confirm that the proposed heuristic is rather efficient at finding violated valid inequalities for larger instances. This can be seen by looking at the results for instances A4-A7 and B4-B7. We first note that, for these instances, the integrality gap is reduced from an average of 7.9% while using formulation DLSPSD1 to an average of 4.7% while using formulation DLSPSD2h. Moreover a significant decrease in the overall computation time is obtained for instances B4-B7 when using formulation DLSPSD2h.

6 Conclusion

We considered the multi-product discrete lot-sizing and scheduling problem with sequence-dependent changeover costs and proposed a new family of multi-product valid inequalities for this problem. This enabled us to better take into account in the MILP formulation the conflicts between different products simultaneously requiring production on the resource. We then presented both an exact and a heuristic separation algorithm in order to identify the most violated valid inequalities to be added in the initial MILP formulation within a

cutting-plane generation algorithm. Our preliminary results show that the proposed valid inequalities are efficient at strengthening the MILP formulation and that their use leads to a significant reduction of the overall computation time for instances featuring a product family changeover cost structure. Additional computational experiments are currently carried out to confirm these promising results.

Acknowledgment

The work described in this paper was funded by the French National Research Agency through its research funding program JCJC2011 for young researchers (project ANR-11-JS0002-01 LotRelax).

References

[1] G. Belvaux and L.A. and Wolsey. Modelling practical lot-sizing problems as mixed-integer programs. *Management Science*, vol. 47(7), pp 993-1007, 2001.

[2] L. Buschkühl, F. Sahling, S. Helber and H. Tempelmeier. Dynamic capacitated lot-sizing problems: classification and review of solution approaches. *OR Spectrum*, vol. 32, 231-261, 2010.

[3] S.G. Dastidar and R. Nagi. Scheduling injection molding operations with multiple resource constraints and sequence dependent setup times and costs. *Computers & Operations Research*, vol. 32, pp 2987-3005, 2005.

[4] G.D. Eppen and R.K Martin. Solving multi-item capacitated lot-sizing problems using variable redefinition. *Operations Research*, vol. 35(6), 832-848, 1987.

[5] D. Ferreira, A.R. Clark, B. Almada-Lobo and R. Morabito. Single-stage formulations for synchronized two-stage lot-sizing and scheduling in soft drink production, *International Journal of Production Economics*, vol. 136, pp 255-265, 2012.

[6] B. Fleischmann. The discrete lot sizing and scheduling problem. *European Journal of Operational Research*, vol. 44, pp 337-348,1990.

[7] C. Gicquel, M. Minoux and Y. Dallery. On the discrete lot-sizing and scheduling problem with sequence-dependent changeover times *Operations Research Letters*, vol. 37, pp 32-36, 2009.

[8] C. Gicquel, M. Minoux and Y. Dallery. Exact solution approaches for the discrete lot-sizing and scheduling problem with identical parallel resources. *International Journal of Production Research*, vol. 49(9), pp 2587-2603, 2011.

[9] R. Jans and Z. Degraeve. Meta-heuristics for dynamic lot sizing: a review and comparison of solution approaches, *European Journal of Operational Research*, vol. 177, pp 1855-1875, 2007.

[10] R. Jans and Z. Degraeve. Modelling industrial lot sizing problems: a review. *Industrial Journal of Production Research*, ,vol. 46(6), pp 1619-1643, 2008.

[11] Y. Pochet and L.A. Wolsey. Production planning by mixed integer programming, Springer Science, 2006.

[12] M. Salomon, M. Solomon, L. van Wassenhove, Y. Dumas and S. Dauzère-Pérès. Solving the discrete lotsizing and scheduling problem with sequence dependant set-up costs and set-up times using the Travelling Salesman Problem with time windows. *European Journal of Operational Research*, vol. 100, pp 494-513, 1997.

[13] C. Silva and J.M. Magalhaes. Heuristic lot size scheduling on unrelated parallel machines with applications in the textile industry. *Computers & Industrial Engineering*, vol. 50, pp 79-89, 2006.

[14] C.A. van Eijl and C.P.M. van Hoesel. On the discrete lot-sizing and scheduling problem with Wagner-Whitin costs. *Operations Research Letters*, vol. 20, pp 7-13, 1997.

[15] H.M. Wagner and T.M. Whitin. Dynamic version of the economic lot size model. *Management Science*, vol. 5(1), pp 89-96, 1958.