7^e *Conférence Francophone de MOdélisation et SIMulation - MOSIM'08 - du* 31 *mars au* 2 *avril* 2008 *- Paris - France*
«Modélisation, Optimisation et Simulation des Systèmes : Communication, Coopération et Coordination»

DISCRETE LOT SIZING AND SCHEDULING USING PRODUCT DECOMPOSITION INTO ATTRIBUTES

C. Gicquel, N. Miègeville, Y.Dallery

Laboratoire de Génie Industriel
Grande Voie des Vignes
92290 Chatenay-Malabry, France
celine.gicquel@ecp.fr
Nicolas.Miegeville@saint-gobain.com
yves.dallery@ecp.fr

M. Minoux

Laboratoire d'Informatique de Paris VI
4 place Jussieu
75005 Paris, France
Michel.Minoux@lip6.fr

RÉSUMÉ : *In the present paper, a production planning problem, known as the Discrete Lot sizing and Scheduling Problem with sequence-dependent changeover costs, is considered. We propose a new way of modelling the production system based on the use of a multi attribute product structure encountered in many industrial situations. The basic idea is to describe the products as combinations of physical attributes and to exploit this description to reduce the size of the mixed-integer programme to be solved. The results of our computational experiments show the practical usefulness of the proposed formulation at improving the efficiency of a standard solver.*

MOTS-CLÉS : *Production planning, Discrete Lot Sizing and Scheduling, Sequence-dependent changeover costs, MIP formulation, Multi attribute product structure*

1. INTRODUCTION

There is a wide variety of models for production planning and inventory management. Capacitated lot sizing models aim at determining a minimal cost production schedule complying with given capacity restrictions and such that demand for all products is satisfied without backlogging. Recent overviews on the lot-sizing litterature can be found among others in (Drexl & Kimms 1997), (Wolsey 2002), (Jans & Degraeve 2007) and (Jans & Degraeve 2008).

In the present paper, the Discrete Lot sizing and Scheduling Problem (DLSP) is considered. As was first defined by (Fleischmann 1990), the DLSP is based on several crucial assumptions:

- Demand for products is deterministic and time-varying.
- The production plan is established for a finite time horizon subdivided in several discrete periods.
- At most one item can be produced per period ("small bucket" model) and the facility processes either one product at full capacity or is completely idle ("all-or-nothing assumption").
- Costs to be minimized are the inventory holding costs and the changeover costs.

Here the single level single machine variant of this problem is studied: all items to be produced are end items and share the same constrained resource. In the DLSP, the changeover costs to be incurred when the production of a new lot begins can depend either on the next item only (sequence-independent case) or on both the previous and the next items (sequence-dependent case). We consider here the more complicated case of sequence-dependent changeover costs.

The DLSP with sequence-dependent changeover costs was first studied by (Fleischmann 1994) and (Salomon, Solomon, van Wassenhove, Dumas & Dauzère-Pérès 1997). They both reformulate the problem as a Travelling Salesman Problem with Time Windows and use either Lagrangean relaxation or a dynamic programming based algorithm to solve it. (Jordan & Drexl 1998) show the equivalence between the DLSP with sequence-dependent changeover costs and the Batch Sequencing Problem (BSP) and use a specific branch-and-bound type algorithm for solving the BSP to optimality. In these papers, the number of items considered in the computational experiments is relatively small (no more than 10 items) whereas the horizon length can be up to 100 periods. (Wolsey 2002) proposes to strengthen a basic MIP formulation of the DLSP with sequence-dependent changeover costs using both a reformulation of the changeover variables and valid inequalities. Thanks to this strengthened formulation, the lower bounds provided by the linear relaxation of the problem are significantly better, enabling a standard solver to solve the problem more efficiently. However, as pointed out by (Belvaux & Wolsey 2001), the large number of variables needed in the reformulation to model changeovers is an important drawback of this approach.

The purpose of the present paper is to propose a new way of modelling the production system to be planned using a *multi attribute product structure* encountered in many industrial situations. The basic idea is to use a set of M relevant physical attributes (e.g. color, diameter...) to describe the products. If this is possible, each item to be produced will be identified, not only by a unique index as it is usually done, but also by a M-tuple, each component of which indicates the value of the corresponding attribute for the given item. When this particular multi attribute product structure can be highlighted in the industrial context under study, we propose to exploit it to reduce the size of the problem to be solved. This can be achieved by looking at changeovers at an aggregate level using the relevant physical attributes instead of considering each individual changeover between items. By doing so, we are able to significantly reduce the number of changeover variables and associated constraints in the formulation, while maintaining the quality of the bounds provided by the linear relaxation of the problem. We then specialize the approach used by (Wolsey 2002) to solve the resulting mixed-integer linear program with a standard solver.

The paper is organized as follows. In section 2, we first introduce the tight reformulation proposed by (Wolsey 2002) for the DLSP with sequence-dependent changeover costs. In our computational experiments, we use it as a reference for comparison with our model. In section 3, we describe our proposal to model the production system using product attributes and discuss our main assumptions. When these assumptions hold, we propose in section 4 to exploit them to formulate the DLSP with sequence-dependent changeover costs with a reduced number of variables and constraints. Some computational results are given in section 5 and section 6 states the concluding remarks.

2. A STRONG FORMULATION FOR THE DLSP WITH SEQUENCE-DEPENDENT SETUP COSTS

In this section, we describe a strong formulation for the DLSP with sequence-dependent changeover costs. This formulation was first presented by (Karmarkar & Schrage 1985) for the variant of the DLSP referred to as CSLP (Continuous Setup Lot sizing Problem), where the all-or-nothing assumption is relaxed. More recently, (Belvaux & Wolsey 2001) and (Wolsey 2002) propose to use it to solve the DLSP with sequence-dependent changeover costs.

We wish to optimize the production schedule for a set of N items over an horizon featuring T planning periods. A period is indexed by $t = 1, ..., T$, an item by $k=0,..,N$. We agree to use item $k = 0$ to denote idle periods.

We use the following notation for the parameters:

- D_{kt}: demand (in units) for item k in t,
- P_{kt}: production capacity (in units per period) for item k in t,
- h_k: holding costs per unit and period for item k,
- c_{kl}: changeover costs from item k to item l.

Decision variables are defined as follows:

- I_{kt}: inventory of item k at the end of t,
- y_{kt}: setup variables. $y_{kt} = 1$ if the resource is setup for item k in period t, and 0 otherwise,
- w_{klt}: changeover variables. $w_{klt} = 1$ if the resource is switched from item k to item l at the beginning of period t, and 0 otherwise.

With this notation, (Belvaux & Wolsey 2001) propose to formulate the DLSP with sequence-dependent changeover costs as follows:

(DLSP1)

$$min \sum_{k=1}^{N}\sum_{t=1}^{T} h_k I_{kt} + \sum_{k=0}^{N}\sum_{l=0}^{N}\sum_{t=1}^{T} c_{kl} w_{klt} \qquad (1)$$

$$\forall k, \forall t, I_{kt} = I_{k,t-1} + P_{kt} y_{kt} - D_{kt} \qquad (2)$$

$$\sum_{k=0}^{N} y_{k0} = 1 \qquad (3)$$

$$\forall k, \forall t, y_{k,t-1} = \sum_{l=0}^{N} w_{klt} \quad (4)$$

$$\forall l, \forall t, y_{lt} = \sum_{k=0}^{N} w_{klt} \quad (5)$$

$$\forall k, \forall l, \forall t, w_{klt} \geq 0 \quad (6)$$

$$\forall k, \forall t, I_{kt} \geq 0 \quad (7)$$

$$\forall k, \forall t, y_{kt} \in \{0, 1\} \quad (8)$$

The objective, minimizing the sum of inventory holding costs and changeover costs, is expressed by (1). Changeover costs c_{kl} are incurred between two successive production batches of item k and item l, in the first period of production of item l. Constraints (2) express the inventory balance. The "all-or-nothing" assumption is enforced by the term $P_{kt}y_{kt}$ in the equality: if the resource is setup for k in period t, then all the available capacity is used and the production quantity of item k must be equal to P_{kt}. (3) is also linked to the "all-or-nothing" assumption: together with constraints (4)-(5), they ensure that in each period, the resource either produces one single product at full capacity, or is idle (i.e $y_{0t} = 1$). Equalities (4) and (5) link the setup variables with the changeover variables. (4) guarantee that item k can be produced in period *t-1* if and only if a changeover from k to another item l (possibly $l = k$) takes place at the beginning of period t. Similarly, (5) guarantee that item l can be produced in period t if and only if a changeover from another item k (possibly $k = l$) to item l takes place at the beginning of period t. (6) state the non-negativity of the changeover variables: observe, as pointed out by (Belvaux & Wolsey 2001), that thanks to constraints (3)-(5) and (8), there is no need to define variables w_{klt} as binary variables. The set of constraints (2) and (7) ensure that demand for each item is fulfilled without backlogging. The binary character of the setup variables is represented by (8).

As suggested by (Wolsey 2002), the formulation DLSP1 can be further strengthened using a family of strong valid inequalities developed for the single-item DLSP with Wagner-Whitin costs, constant capacity and no backlogging. In this case, demands and production capacity can be normalized without loss of generality: $D_{kt} \in \{0, 1\}$ and $P_{kt} = 1$. The reader is referred to (van Eijl & van Hoesel 1997) for more details. In our computational experiments, we add these valid inequalities to the formulation according to a cutting-plane generation strategy.

One important drawback of the DLSP1 formulation is that the number of variables needed in the formulation to model changeovers, $(N + 1)^2T$, grows very rapidly with the problem size. In the sequel, we present one way to avoid this issue in certain situations, namely when products can be described as combinations of a number of physical attributes.

3. THE DLSP WITH PRODUCTS DESCRIBED AS COMBINATIONS OF PHYSICAL ATTRIBUTES

In most papers dealing with the DLSP, each individual item to be produced is described with a single index (k in the formulation presented above) and is considered independently of the other items. However, in many industrial situations, it is possible to identify a set of physical characteristics or attributes (e.g. color, diameter, size, shape, mixture composition, quality level...) that can be used to describe the items to be produced. Moreover it is frequently the case that many items share a common value for some attribute so that we can define a (small) finite number of possible values for each attribute. In what follows, we propose to exploit this remark to state and solve the DLSP. In order to do this, we need to make several assumptions on the production system.

3.1 Main assumptions

1. We first suppose that each item to be produced can be described by a set of physical attributes, each of them takes a finite number of discrete values. We also suppose that each item is uniquely identified thanks to a M-tuple, each component of which gives the value of the corresponding attribute for the given item.

2. Second, we assume that the setup state of the resource can also be described using product attributes. Thus, we will not describe the setup state of the resource by indicating the item that the resource is able to produce, but by indicating, for each attribute, for which value of this attribute the resource is setup. The resource setup state will therefore also be described by a M-tuple, each component of which gives the value of the corresponding attribute for the present state of the resource. To ensure consistency, we will require that a given item can be produced on the resource if and only if the resource is setup with the correct value for each attribute.

3. Third, we assume that we are able to evaluate the changeover costs on the resource for each attribute separately. This means that given an attribute and two possible values for this attribute, we are able to evaluate the cost of a changeover from one value to the other and that this cost does not depend on the setup state of the resource with respect to the other attributes.

4. Finally, we need to make an assumption on the way

the costs relative to different attributes will combine, i.e. on the way we will compute changeover costs when changeovers for different attributes happen simultaneously on the resource. In this paper, we will consider the case where the global changeover costs will be the *sum* of all individual changeover costs for the different attributes. Another possible assumption would be that global changeover costs equal the *maximum* of the individual costs.

Thanks to these assumptions, we are able to decide about the production plan on the resource using the product attributes. In this case, the production plan consists of a set of parallel sequences, one for each attribute. Each of these sequences indicates, for every planning period, for which value of the corresponding attribute the resource is setup. Thus, in each planning period, combining the values for the different attributes, we are able to deduce the item for which the resource is setup. A detailed mathematical programming formulation is given in section 4, but in order to clarify and illustrate our proposition for the production system model, we first present some industrial situations where our model could possibly apply.

3.2 Possible industrial applications

In order to illustrate the proposed model, we provide examples of industrial situations found in the literature where using physical attributes to describe the products could be useful.

- In (Dobson 1992), a production planning problem for a bottle filling line is considered. For this type of production line, two physical attributes of the products have to be taken into account to plan production: the size or shape of the bottle and the liquid used to fill it. Hence each item can be described by means of two attributes: the bottle size/shape and the liquid to be used. Each individual item would be described by a pair (i_1, i_2) where i_1 would give the index of the corresponding bottle size and shape and i_2 the index of the corresponding liquid.

- (dos Santos-Meza, dos Santos & Arenales 2002) discuss a lot sizing problem they found in an automated foundry. Each item to be produced can be described by two attributes: the type of metal alloy it is made of and the shape it takes from the used mould. Here we could use as well a pair (i_1, i_2) where i_1 would give the index of the alloy type and i_2 the index of the mould shape.

- (Silva & Magalhaes 2006) study a production planning problem arising in the textile industry in a company producing acrylic fibers. The authors report that two physical characteristics of the products have an impact on the scheduling of the plant spinning unit, namely the fibers composition and their diameter. Thus, we could use a pair (i_1, i_2) to describe each item: i_1 would indicate the fiber composition and i_2 would refer to its diameter.

- (Miegeville 2005) considers the production planning problem for a float glass production line. Here each item (a glass sheet) can be described using several physical characteristics: glass color and quality, dimensions of the sheet (thickness, width and length). An item could thus be described using a 5-tuple, with components corresponding to the color, quality and dimensions of the corresponding glass sheet. In fact, the production system model studied in this paper was first introduced and used on this industrial case.

As can be seen here, the proposed product description using physical attributes is likely to be applied on a large variety of industrial situations.

4. A FORMULATION FOR THE DLSP WITH PRODUCT ATTRIBUTES AND SEQUENCE-DEPENDENT CHANGEOVER COSTS

We now present a formulation for the DLSP with product attributes and sequence-dependent changeover costs. This formulation can be used to solve the DLSP when a product description using physical attributes is possible and when the assumptions discussed above hold.

We use the same notation as in section 2 for the parameters relative to items:

- D_{kt}: demand for item k in t,
- P_{kt}: production capacity for item k in t,
- h_k: holding costs per unit and period for k.

We assume that each item can be described using M physical characteristics or attributes. Correspondence between products and attributes is given by a matrix $\mathcal{A}$ of dimensions $M \times (N+1)$. $\mathcal{A}_{mk}$ represents the value of the attribute m for product k and the k^{th} column of $\mathcal{A}$ gives the M-tuple describing product k in terms of product attributes. For each attribute m, we have:

- a set of possible values: $i \in [1, V^m]$. We agree to use an additional value $i = 0$ to describe the idle state of the resource with respect to attribute m.
- a changeover cost matrix: $\mathcal{C}^m$. C^m_{ij} is the cost of a transition from the value $i \in [0, V^m]$ to the value $j \in [0, V^m]$ of attribute m.

We agree to use the M-tuple (0,0,....,0) to describe the item $k = 0$. Although our model could be easily modified to integrate it, we do not consider in the sequel the case where the resource is idle (i.e. setup for the value $i = 0$) with respect to some attributes and active (i.e. setup for a value $i \geq 1$) with respect to some others. Thus we have:

$$\mathcal{A}_{mk} = \begin{cases} 0 & \text{if } k = 0, \\ \geq 1 & \text{if } k \in [1; N] \end{cases}$$

We use the following decision variables:

- I_{kt}: inventory of item k at the end of t,
- y_{kt}: setup variables at the item level. $y_{kt} = 1$ if the resource is setup for item k in period t, and 0 otherwise,
- w^m_{ijt}: changeover variables at the attribute level. $w^m_{ijt} = 1$ if a switch from the value i to the value j of attribute m takes place at the beginning of period t, and 0 otherwise.

Under the assumption that changeover costs related to different attributes are added whenever two transitions occur simultaneously, the DLSP can be formulated as follows:

(DLSP2)

$$min \sum_{k=1}^{N}\sum_{t=1}^{T} h_k I_{kt} + \sum_{m=1}^{M}\sum_{i=0}^{V^m}\sum_{j=0}^{V^m}\sum_{t=1}^{T} C^m_{ij} w^m_{ijt} \quad (9)$$

$$\forall k, \forall t, I_{kt} = I_{k,t-1} + P_{kt} y_{kt} - D_{kt} \quad (10)$$

$$\sum_{k=0}^{N} y_{k0} = 1 \quad (11)$$

$$\forall m, \forall i \in [0, V^m], \forall t, \sum_{k \text{ st } \mathcal{A}_{mk}=i} y_{k,t-1} = \sum_{j=0}^{V^m} w^m_{ijt} \quad (12)$$

$$\forall m, \forall j \in [0, V^m], \forall t, \sum_{k \text{ st } \mathcal{A}_{mk}=j} y_{kt} = \sum_{i=0}^{V^m} w^m_{ijt} \quad (13)$$

$$\forall m, \forall (i,j) \in [0, V^m] \times [0, V^m], \forall t, w^m_{ijt} \geq 0 \quad (14)$$

$$\forall k, \forall t, I_{kt} \geq 0 \quad (15)$$

$$\forall k, \forall t, y_{kt} \in \{0, 1\} \quad (16)$$

The objective, minimizing the sum of changeover costs and inventory holding costs, is expressed by (9). Note that inventory holding costs are computed item by item whereas changeover costs are computed attribute by attribute. Constraints (10) express the inventory balance. Combined with the non negativity constraints (15), they prevent any backlogging. (11), together with constraints (12)-(13), guarantee that in each period the resource either produces a single item or is idle.

Equalities (12) and (13) link the setup variables with the changeover variables. First note that the term $\sum_{k \text{ st } \mathcal{A}_{mk}=i} y_{kt}$ equals 1 if and only if an item k requiring the resource to be setup for the value i of the attribute m is produced in period t, i.e. if and only if the resource is setup for the value i of attribute m in period t. Thus (12) guarantee that the resource is setup for the value i of attribute m in period *t-1* if and only if a changeover from value i to another possible value j of attribute m (possibly $j = i$) takes place at the beginning of period t. Similarly, (13) guarantee that the resource can be setup for the value j of attribute m in period t if and only if a changeover from another possible value i of attribute m (possibly $i = j$) to value j takes place at the beginning of period t. The non negativity of the changeover variables is stated by (14) and the binary character of the setup variables is represented by (16).

Let us now compare the number of changeover variables in the formulations DLSP1 and DLSP2. For the sake of simplicity, we do not consider the item $k = 0$ in the computation. As shown in section 2, in this case, DLSP1 includes N^2T changeover variables, one for each possible pair of items and for each period. Note that when the product description using attributes is possible, we can compute the number of products as the number of possible combinations obtained by choosing for each attribute m one value out of V^m. Thus we have: $N^2 = \prod_{m=1}^{M} V^{m2}$. Now, as can be seen above, in the DLSP2 formulation, there are $\sum_{m=1}^{M} V^{m2} T$ changeover variables, one for each pair of possible values of each attribute and for each period. In most cases where the product description using attributes will be implemented, we will have: $\sum_{m=1}^{M} V^{m2} \ll \prod_{m=1}^{M} V^{m2}$, thus leading to a significant reduction in the number of changeover variables needed in the formulation.

In words, in our proposed model, we do not consider each individual changeover between items, but rather look at changeovers at a more aggregate level using product attributes. By doing so, we are able to significantly reduce the size of the mixed integer linear programm to be solved by a standard solver.

As for the DLSP1 formulation, the DLSP2 formulation can be strengthened under the assumption of Wagner-Whitin costs, constant capacity and no backlogging. This is done through a family of valid inequalities adapted for the DLSP2 formulation from the ones developed by (van Eijl & van Hoesel 1997). In the sequel, these valid inequalities are added to the formulation according to a cutting-plane generation strategy.

4.1 Simple example

We use a very simple example to illustrate the proposed model and to show an application of the DLSP2 formulation. We consider a bottle filling line where 4 items can be produced. An item is described by the corresponding bottle size (attribute 1 with two possible values) and the composition of the liquid to be used (attribute 2 with two possible values). Table 1 shows how each of the 4 items can be described using the two attributes. We agree to use the item $k = 0$ described by the pair (0,0) for the idle period. Table 2 gives the changeover costs for each attribute and table 3 provides the demand for each product.

Figure 1 shows the optimal production plan obtained while using the DLSP2 formulation. The first two lines give the sequence of setup states for each attribute. In each planning period, we can deduce from these sequences the item for which the resource is setup. The changeover costs to be incurred between each lot are shown below. We used the assumption that changeover costs relative to different attributes are added whenever transitions for different attributes occur simultaneously. This is the case here at the beginning of periods 1, 4, 9 and 10 where both the bottle size and the liquid composition are changed.

5. COMPUTATIONAL RESULTS

In this section, we discuss the results of some computational experiments carried out to compare the two formulations presented in sections 2 and 4. We created 5 sets of randomly generated instances. The instances differ with respect to the following characteristics:

- *Problem dimension*: The problem dimension is represented by the number of products N and the number of periods T. We use three different combinations: $(N = 10, T = 60)$, $(N = 25, T = 50)$ and $(N = 30, T = 100)$.
- *Multi-attribute product structure*: The product structure is described by the number of attributes M and the number of possible values V^m for each attribute m. We use five different combinations, leading to 5 sets of instances. Table 4 gives the characteristics of the generated instances for each set.

  For set C and E instances, we have $\prod_{m=1}^{M} V^m > N$. Therefore we used the following procedure to generate matrix $\mathcal{A}$:
  1) We generated a matrix $\mathcal{A}'$ with $\prod_{m=1}^{M} V^m$ columns. $\mathcal{A}'$ describes all possible combinations of the attribute values.
  2) For each column of $\mathcal{A}'$, we randomly generated a number w_k from a discrete uniform DU(1,$\prod_{m=1}^{M} V^m$) distribution.
  3) The matrix $\mathcal{A}$ is generated by selecting the N columns of $\mathcal{A}'$ with the smallest numbers w_k.
- *Inventory holding costs*: For each item, inventory holding costs have been generated randomly from a discrete uniform *DU(5,10)* distribution.
- *Production capacity utilization*: Production capacity utilization ρ is defined as the ratio of the total cumulated demand on the total cumulated available capacity. We experimented different values for ρ: 0.5, 0.7 and 0.9.
- *Demand pattern*: Binary demand for each product have been randomly generated according to the procedure described in (Salomon et al. 1997).
- *Changeover costs*: For each attribute m, changeover costs C_{ij}^m have been randomly generated from a discrete uniform $DU(C_{min}^m, C_{max}^m)$ distribution. We tested several possibilities: the changeover costs for all attributes can either be taken from the same interval or the changeover costs for the first attribute are greater than for the other(s). In our study, we define the ratio r as: $r = \frac{C_{mean}^1}{C_{mean}^m}$ where C_{mean}^m denotes the mean of interval $[C_{min}^m, C_{max}^m]$. We tested several values for r: 1, 2, 5, 10 and 30. In all instances, the resulting changeover costs between two items belong to the interval [0,200].

For each possible combination of multi-attribute product structure, production capacity utilization and changeover costs ratio, 5 problems were generated, resulting in $5 \times 3 \times 5 \times 5 = 375$ instances. All tests were run on a Pentium 4 (2.8 Ghz) with 505 Mo of RAM, running under Windows XP. We used a standard MIP software (CPLEX 8.1.0) with the solver default settings to solve the problems, using either formulation DLSP1 or formulation DLSP2.

Tables 5-9 show the computational results obtained with the two formulations, for each set of instances.

item	0	1	2	3	4
attribute 1: bottle size	0	1	1	2	2
attribute 2: liquid composition	0	1	2	1	2
product description	(0,0)	(1,1)	(1,2)	(2,1)	(2,2)
inventory holding costs	0	7	8	5	10

Table 1: Product description using two attributes

	0	1	2
0	0	100	200
1	0	0	200
2	0	100	0

	0	1	2
0	0	10	10
1	0	0	20
2	0	10	0

Table 2: Changeover costs for transitions between bottle sizes (left) and between liquid composition (right)

period	1	2	3	4	5	6	7	8	9	10
item 1	0	1	0	0	1	0	0	1	0	0
item 2	0	0	0	0	0	0	0	0	0	1
item 3	0	0	0	0	1	1	0	1	0	1
item 4	0	0	0	1	0	0	0	0	0	0

Table 3: Demand on products

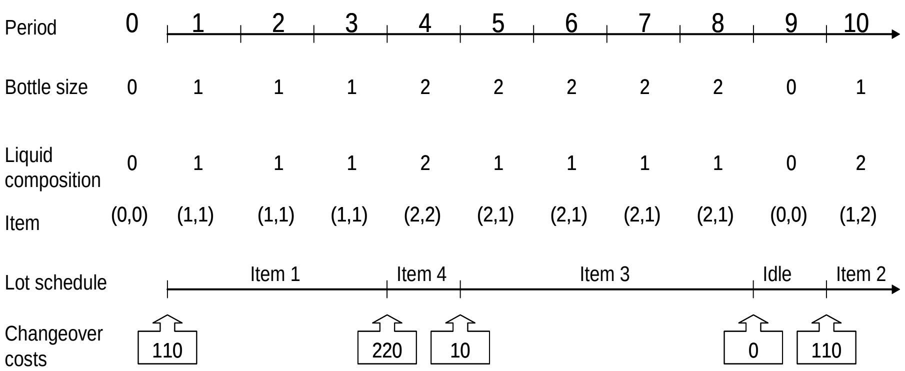

Figure 1: Optimal production plan for the simple example: optimal cost $Z^* = 528$

	N	T	M	V^m
set A	10	60	2	$V^1 = 2$, $V^2 = 5$
set B	25	50	2	$V^1 = 5$, $V^2 = 5$
set C	25	50	3	$V^1 = 3$, $V^2 = 3$, $V^3 = 3$
set D	30	100	3	$V^1 = 2$, $V^2 = 3$, $V^3 = 5$
set E	30	100	5	$V^1 = 2$, $V^2 = 2$, $V^3 = 2$, $V^4 = 2$, $V^5 = 2$

Table 4: Characteristics of generated instances

As the value of the ratio r appears to have an impact on the results quality, we grouped the instances with respect to the value of r. For both series of results, we provide:

- *#Opt*: for set A, B and C instances, the number of instances out of the corresponding 15 instances that could be solved to optimality within 30 minutes of computation.
- *#Feas*: for set D and E instances, the number of instances out of the corresponding 15 instances for which a feasible solution could be found within 30 minutes of computation.
- *Gap*: for the instances that could not be solved to optimality, the mean gap obtained after 30 minutes of computation between the best integer solution (if one could be found) and the best lower bound found.

We now compare the results obtained with the DLSP1 and DLSP2 formulation. The results from tables 5-9 show that:

- for high values of the ratio r ($r \geq 5$), i.e. when one attribute has corresponding changeover costs clearly higher than the other(s) attribute(s), the results obtained with the DLSP2 formulation are better. This can be seen as:
  - a feasible solution could be obtained for all instances,
  - more instances could be solved to optimality within 30 minutes of computation,
  - when a guaranteed optimal solution could not be found within 30 minutes of computation, the residual gap is smaller.
- for small values of the ratio r ($r \leq 2$), the DLSP1 formulation provides better results for medium-sized instances (sets A, B and C). However, this is not the case for the larger instances in sets D and E. Namely, for these instances,
  - a feasible solution could not always be found with DLSP1 formulation whereas at least one feasible solution could be found for each instance with the DLSP2 formulation.
  - the residual gap is significantly smaller on some instances with the formulation DLSP2.

Comparison between the results obtained with the two formulations thus shows that using the DLSP2 formulation, we are able to improve the efficiency of the Branch & Bound procedure, especially for the high values of ratio r and for the largest instances. This can be explained by two main factors:

- using the DLSP2 formulation, the problem size (i.e. the number of variables and constraints) is significantly reduced. As a consequence, the time spent at each node of the Branch & Bound tree to solve the linear relaxation is shorter and more nodes can be explored within 30 minutes of computation.
- the formulation enhancement obtained thanks to the valid inequalities adapted for the DLSP2 formulation gives better results when ratio r has a high value. More precisely, for high values of r, the lower bounds provided by the enhanced DLSP2 formulation are higher than the ones provided by the enhanced DLSP1 formulation. On the contrary, for small values of r, the lower bounds provided by the enhanced DLSP1 formulation are higher than the ones provided by the enhanced DLSP2 formulation.

Thus the combined advantages of a reduced problem size and of tighter lower bounds enable the DLSP2 formulation to outperform the DLSP1 formulation on many instances.

6. CONCLUSION

In this paper, we presented a new formulation for the DLSP with sequence-dependent setup costs. The main idea is to use a possible product description as combinations of a number of physical attributes. When this description can be used in the industrial context under study, we exploit it to reduce the size of the mixed integer linear programm to be solved. Computational experiments show that the proposed DLSP2 formulation performs better than the tight DLSP1 formulation we chose as a reference for comparison, especially in cases where one of the physical attributes has corresponding changeover costs higher than the other(s) attribute(s).

An interesting subject for future research could be to investigate possible extensions of our model to problems with sequence-dependent changeover times and/or multiple resources.

References

Belvaux, G. & Wolsey, L. (2001). Modelling practical lot-sizing problems as mixed-integer programs,

	Formulation DLSP1		Formulation DLSP2	
Variables	8520		3960	
Constraints	1921		1681	
Avge nb of added VI	1191		2108	
ratio r	#Opt	Gap	#Opt	Gap
r=1	13	3%	5	9%
r=2	10	6%	5	5%
r=5	2	12%	6	4%
r=10	0	21%	11	3%
r=30	1	14%	15	0%

Table 5: Results for set A instances (average values for 15 random instances for each value of r)

	Formulation DLSP1		Formulation DLSP2	
Variables	36350		6150	
Constraints	5101		2451	
Avge nb of added VI	838		1453	
ratio r	#Opt	Gap	#Opt	Gap
r=1	3	7%	0	17%
r=2	0	16%	0	16%
r=5	0	26%	4	11%
r=10	0	31%	7	4%
r=30	0	36%	11	5%

Table 6: Results for set B instances (average values for 15 random instances for each value of r)

	Formulation DLSP1		Formulation DLSP2	
Variables	36350		4950	
Constraints	5101		2451	
Avge nb of added VI	840		1653	
ratio r	#Opt	Gap	#Opt	Gap
r=1	5	11%	0	20%
r=2	3	13%	0	21%
r=5	0	24%	0	14%
r=10	0	24%	9	10%
r=30	0	39%	8	7%

Table 7: Results for set C instances (average values for 15 random instances for each value of r)

	Formulation DLSP1		Formulation DLSP2	
Variables	102200		12200	
Constraints	9201		5601	
Avge nb of added VI	2792		2865	
ratio r	#Feas	Gap	#Feas	Gap
r=1	10	38%	15	40%
r=2	7	42%	15	42%
r=5	10	48%	15	35%
r=10	10	57%	15	29%
r=30	10	62%	15	24%

Table 8: Results for set D instances (average values for 15 random instances for each value of r)

	Formulation DLSP1		Formulation DLSP2	
Variables	102200		10600	
Constraints	9201		6001	
Avge nb of added VI	2792		2911	
ratio r	#Feas	Gap	#Feas	Gap
r=1	12	29%	15	31%
r=2	10	34%	15	36%
r=5	9	47%	15	33%
r=10	10	54%	15	29%
r=30	12	59%	15	26%

Table 9: Results for set E instances (average values for 15 random instances for each value of r)

Management Science **47**(7): 993–1007.

Dobson, G. (1992). The cyclic lot scheduling problem with sequence-dependent setups, *Operations Research* **40**(4): 736–749.

dos Santos-Meza, E., dos Santos, M. & Arenales, M. N. (2002). A lot-sizing problem in an automated foundry, *European Journal of Operational Research* **139**: 490–500.

Drexl, A. & Kimms, A. (1997). Lot sizing and scheduling - survey and extensions, *European Journal of Operational Research* **99**: 221–235.

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

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

Jans, R. & Degraeve, Z. (2007). Meta-heuristics for dynamic lot sizing: a review and comparison of solution approaches, *European Journal of Operational Research* **177**: 1855–1875.

Jans, R. & Degraeve, Z. (2008). Modeling industrial lot sizing problems: a review, *International Journal of Production Research* **46**(6): 1619–1643.

Jordan, C. & Drexl, A. (1998). Discrete lotsizing and scheduling by batch sequencing, *Management Science* **44**(5): 698–713.

Karmarkar, U. & Schrage, L. (1985). The deterministic dynamic product cycling problem, *Operations Research* **33**(2): 326–345.

Miegeville, N. (2005). *Supply Chain Optimization in the process industry. Methods and Case Study of the Glass Industry*, PhD thesis, Ecole Centrale Paris, Paris, France.

Salomon, M., Solomon, M., van Wassenhove, L., Dumas, Y. & Dauzère-Pérès, S. (1997). Solving the discrete lotsizing and scheduling problem with sequence dependent set-up costs and set-up times using the travelling salesman problem with time windows., *European Journal of Operational Research* **100**: 494–513.

Silva, C. & Magalhaes, J. (2006). Heuristic lot size scheduling on unrelated parallel machines with applications in the textile industry, *Computers & Industrial Engineering* **50**: 76–89.

van Eijl, C. & van Hoesel, C. (1997). On the discrete lot-sizing and scheduling problem with wagner-whitin costs, *Operations Research Letters* **20**: 7–13.

Wolsey, L. (2002). Solving multi-item lot-sizing problems with an `MIP` solver using classification and reformulation, *Management Science* **48**(12): 1587–1602.