
We recommend you cite the published version.
The publisher’s URL is: http://dx.doi.org/10.1016/j.cie.2013.04.003

Refereed: Yes

(no note)

Disclaimer

UWE has obtained warranties from all depositors as to their title in the material deposited and as to their right to deposit such material.

UWE makes no representation or warranties of commercial utility, title, or fitness for a particular purpose or any other warranty, express or implied in respect of any material deposited.

UWE makes no representation that the use of the materials will not infringe any patent, copyright, trademark or other property or proprietary rights.

UWE accepts no liability for any infringement of intellectual property rights in any material deposited but will remove such material from public view pending investigation in the event of an allegation of any such infringement.

PLEASE SCROLL DOWN FOR TEXT.
Accepted Manuscript

A priori reformulations for joint rolling-horizon scheduling of materials processing and lot-sizing problem

Silvio Alexandre de Araujo, Alistair Clark

PII: S0360-8352(13)00118-6
DOI: http://dx.doi.org/10.1016/j.cie.2013.04.003
Reference: CAIE 3443

To appear in: Computers & Industrial Engineering

Received Date: 27 June 2012
Revised Date: 3 April 2013
Accepted Date: 4 April 2013

Please cite this article as: Araujo, S.A.d., Clark, A., A priori reformulations for joint rolling-horizon scheduling of materials processing and lot-sizing problem, Computers & Industrial Engineering (2013), doi: http://dx.doi.org/10.1016/j.cie.2013.04.003

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.
A priori reformulations for joint rolling-horizon scheduling of materials processing and lot-sizing problem

Silvio Alexandre de Araujo
Departamento de Matemática Aplicada, Universidade Estadual Paulista (Unesp)
Rua Cristovão Colombo 2265, São José do Rio Preto SP, Brazil, 15054-000,
saraujo@ibilce.unesp.br

Alistair Clark*
Department of Engineering Design and Mathematics, University of the West of England, Coldharbour Lane, Bristol, BS16 1QY, United Kingdom
Alistair.Clark@uwe.ac.uk
Tel: +44 117 328 3134. Fax: none

* Corresponding author
A priori reformulations for joint rolling-horizon scheduling of materials processing and lot-sizing problem

In many production processes, a key material is prepared and then transformed into different final products. The lot sizing decisions concern not only the production of final products, but also that of material preparation in order to take account of their sequence-dependent setup costs and times. The amount of research in recent years indicates the relevance of this problem in various industrial settings. In this paper, facility location reformulation and strengthening constraints are newly applied to a previous lot-sizing model in order to improve solution quality and computing time. Three alternative metaheuristics are used to fix the setup variables, resulting in much improved performance over previous research, especially regarding the use of the metaheuristics for larger instances.

Keywords: Lot Sizing and Scheduling, Facility Location Reformulation, Valid Inequalities, Metaheuristics.

1. Introduction

The lot sizing problem is frequently encountered in industrial production planning. How does a planner decide the lot size of each product produced on one or more machines in each demand period over a planning horizon? This problem has been extensively researched, as discussed in the reviews by Bahl et al. (1987), Drexl and Kimms (1997), Brahimi et al. (2006), Karimi et al. (2003) and Jans and Degraeve (2007, 2008).

In some industrial sectors, production setup costs and times are sequence-dependent, so that the decisions about the production of lots concern their sequence as well as their size. The two sets of decisions are mutually dependent and so should be modelled and decided simultaneously rather than separately. For example, Araujo et al. (2007) consider the processing of key materials at an initial stage as well as the products that use the materials. In such a two-stage system, the sequencing decisions at the end stage must jointly consider products that use the same materials, so that material changeovers are minimised at the prior stage. In other words, the sequencing at both stages must be integrated. The model in Araujo et al. (2007) is based on a classical
formulation of the lot sizing problem, but its complexity meant that an established industrial-grade optimisation solver was unable to find an optimal solution within acceptable computing time. As a result, the paper developed heuristic solution methods based on relax-and-fix within a rolling horizon approach and incorporating metaheuristics.

Since then, many other authors have also researched the integrated lot-sizing and sequencing problem with material preparation at a prior stage in a wide range of industrial settings. Examples include soft drink production (Toledo et al., 2009 and Ferreira et al., 2009, 2010, 2012), animal feed (Toso et al., 2009 and Clark et al., 2010), electrofused grains (Luche et al., 2009), glass bottles (Almada-Lobo et al., 2007, 2008), foundries (Araujo et al., 2008; Tonaki and Toledo, 2010 and Camargo et al., 2012), yogurt packaging company (Marinelli et al., 2007), pharmaceutical company (Stadtler, 2011) and sand casting operations (Hans and Van de Velde, 2011).

Most of this recent research, including Araujo et al. (2007), is based on the General Lot Sizing and Scheduling Problem (GLSP) model (Fleischmann and Meyr, 1997) in which the planning horizon is subdivided into macro-periods, in each of which multiple products can be produced. To model the sequence of lots, each macro-period is in turn subdivided into micro-periods in which at most one product can be produced. This special structure involving subperiods within macro time periods is similar to a small-bucket framework (Koçlar, 2005).

However, some papers, such as Clark et al. (2010) and Ferreira et al. (2012) take a different approach, using an asymmetric travelling salesman problem (ATSP) representation for sequencing lots rather than a GLSP-type model. The results presented in Ferreira et al. (2012) shown the superiority of their ATSP-type model over a GLSP-type model. One possible reason is the poor quality of the GLSP linear relaxation as a lower bound on the optimal solution.

Taking forward research initiated in Bernardes et al. (2010), the first contribution of this current paper is to demonstrate that certain reformulations applied to the GLSP-type model in Araujo et al. (2007) can provide improved solutions using established optimisation solvers, due mainly to their better quality of linear-relaxation lower-bounds, contributing to the growing research in this area. The second contribution is to show computationally that the use of metaheuristic methods can help
solve the reformulations more quickly and better than established solvers, such as Cplex 12.0.

This paper is structured as follows. In section 2, the original model in Araujo et al. (2007) is presented. Section 3 develops extended formulations and proposes new constraints. In section 4, a reformulated rolling horizon-based model is proposed while section 5 presents the metaheuristics. Section 6 computationally compares the quality of the reformulations using the Branch-and-Cut search within the solver Cplex 12.0 and using the metaheuristics. Section 7 concludes and poses challenges for future research.

2. Original Formulation (OF)

In Araujo et al. (2007), a material may be used in multiple products, but a product is made from just one material. A product must be manufactured in the same time period in which its material is processed. Thus processed materials cannot be held over from one period to the next. In each time period only one material can be processed on a given machine. A setup changeover from one material to another is sequence-dependent, i.e., it consumes capacity time in a manner that is dependent on the sequence in which the materials are processed. The triangle inequality holds for setup costs and times so that it is optimal to produce at most one lot per product per period. The model allows backlogs as well as inventory.

The sequencing decisions are made by dividing a period into smaller subperiods, as in the General Lot Sizing and Scheduling Problem (GLSP) model (Fleischmann and Meyr, 1997; Drexl and Kimms, 1997; Meyr, 2000, 2002). Let $K$ be the total number of materials, $P$ the total number of products, $T$ the total number of periods and $\eta$ the total number of subperiods. Consider the following indices and data:

Indices: 

- $j, k = 1, \ldots, K$ processed materials
- $p = 1, \ldots, P$ products
- $t = 1, \ldots, T$ periods
- $n = 1, \ldots, \eta$ subperiods
Data: Capacity available on the machine in each subperiod.

\( \rho_p \) Capacity required to produce one unit of product \( p \).

Demand for product \( p \) at the end of period \( t \).

\( S(k) \) Set of products \( p \) that use material \( k \). Each product uses just one material, i.e., \( \{1, ..., P\} = S(1) \cup ... \cup S(K) \), and \( S(k) \cap S(j) = \emptyset \), for all materials \( k \neq j \), implying \( \sum_k |S(k)| = P \).

\( h_{pt}^- \) Backlog penalty for delaying delivery of a unit of product \( p \) at the end of period \( t \).

\( h_{pt}^+ \) Inventory penalty for holding a unit of product \( p \) at the end of period \( t \).

\( s_{jk} \) Setup penalty (or cost) for changing over from material \( j \) to material \( k \), where \( s_{jj} = 0 \).

\( st_{jk} \) Setup time (loss of machine capacity) for changing over from material \( j \) to material \( k \), where \( st_{jj} = 0 \).

Variables: \( x_{pn} \) Quantity (lot-size) of product \( p \) to be produced in subperiod \( n \)

\( I_{pt}^+ \) Inventory of product \( p \) at the end of period \( t \), where \( I_{p0}^+ \) is the initial inventory at the start of period 1.

\( I_{pt}^- \) Backlog of product \( p \) at the end of period \( t \), where \( I_{p0}^- \) is the initial backlog at the start of period 1.

\( y_{kn}^k \) Binary variable: \( y_{kn}^k = 1 \) if the machine is configured for production of material \( k \) in subperiod \( n \), otherwise \( y_{kn}^k = 0 \). Note that the initial setup state \( y_{0n}^k \) is set to zero and so not a variable.
\( z_{nk}^{jk} \) Binary setup variable: \( z_{nk}^{jk} = 1 \) if there is a machine changeover from material \( j \) to material \( k \) at the start of subperiod \( n \), otherwise \( z_{nk}^{jk} = 0 \).

Thus \( z_{nk}^{jk} = 1 \) if \( y_{n-1}^j = 1 \) & \( y_n^k = 1 \), and \( z_{nk}^{jk} = 0 \) if \( y_{n-1}^j = 0 \) or \( y_n^k = 0 \). It is relaxed to be continuous for reasons explained below.

Consider also the following definitions from the GLSP model:

- \( \eta_t \) Maximum number of subperiods in period \( t \)
- \( F_t = 1 + \sum_{\tau=1}^{t-1} \eta_\tau \) First subperiod in period \( t \)
- \( L_t = F_t + \eta_t - 1 \) Last subperiod in period \( t \)
- \( \eta = \sum_{t=1}^{T} \eta_t \) Total number of subperiods over periods 1,...,\( T \)

The **Original Formulation** (OF) (Araujo et al., 2007) of the mathematical model is:

\[
\begin{align*}
\text{Minimize} & \quad \sum_{p} \sum_{t} \left( h_{pt}^- I_{pt}^- + h_{pt}^+ I_{pt}^+ \right) + \sum_{j} \sum_{k} \sum_{n=F} \sum_{k} \sum_{n=F} z_{nk}^{jk} \\
\text{subject to} & \quad I_{p,t-1}^- - I_{p,t}^- + \sum_{n=F} x_{pn} = I_{pt}^+ + I_{pt}^- = d_{pt} \quad \forall \ p, t \\
& \quad \sum_{p \in S(k)} \rho_p x_{pn} + \sum_{j} \sum_{k} \sum_{n=F} \sum_{k} \sum_{n=F} z_{nk}^{jk} \leq C y_n^k \quad \forall \ j, k, n=F,\ldots,L_T \\
& \quad z_{nk}^{jk} \geq y_{n-1}^j + y_n^k - 1 \quad \forall \ j, k, n=F,\ldots,L_T \\
& \quad \sum_{k} \sum_{j=1} z_{nk}^{jk} = 1 \quad \forall \ t \\
& \quad \sum_{j=1}^K \sum_{k=1}^K z_{nk}^{jk} = 1 \quad \forall \ t \\
& \quad y_n^k \in \{0,1\} \text{ with } y_0^k = 0 \quad \forall \ k, n=F,\ldots,L_T \\
& \quad z_{nk}^{jk} \geq 0 \quad \forall \ j, k, n=F,\ldots,L_T \\
& \quad x_{pn} \geq 0 \text{ and integer} \quad \forall \ p, n=F,\ldots,L_T \\
& \quad I_{pt}^+ \text{ and } I_{pt}^- \geq 0 \quad \forall \ p, t
\end{align*}
\]

The objective function (1) minimizes a weighted sum of inventory & backlog penalties and sequence-dependent setup penalties. Constraints (2) balance inventories,
backlogs, demand and production in each period. Constraints (3) not only keep production within the machine’s capacity, but also ensure that only products of the same material are produced in the same machine loading. As $y_n^k$ and $y_{n-1}^j$ are both binary variables, constraints (4) and the objective function (1) force the continuous variable $z_n^{jk}$ to have value 1 if there is a changeover from material $j$ to material $k$ and, along with constraints (7), to have value 0 otherwise. Constraints (5) and (7) ensure that there is only a single machine loading in each subperiod. Constraints (6) ensure one changeover in the first subperiod of each period.

The model assumes that the system is not initially setup for any material, and that every setup must be completed in a subperiod. This means that the capacity time $st_{jk}$ needed to setup a material $k$ must not be greater than $C$, the capacity available on the machine in each subperiod, as enforced in constraints (3).

Most models generally consider the lot-size variables $x_{pn}$ to be continuous. If an $x_{pn}$ value is large, then simply rounding it to the nearby integer value is usually feasible and near-optimal. However, for problems where capacity or demand is low then the integer conditions (9) on $x_{pn}$ must hold. Constraints (10) measure inventory $I_{pt}^+$ and backlogs $I_{pt}^-$ as non-negative variables, but for a given pair $(p,t)$, $I_{pt}^+$ and $I_{pt}^-$ will not both be strictly positive in an optimal solution, given their positive coefficients in the objective function (1).

3. A priori Reformulations

Recently, many authors (Alfieri et al., 2002, Pochet and Van Vyve, 2004; Jans and Degraeve, 2004a; Denizel and Süral, 2006; Denizel et al., 2008; Jans, 2009, and Süral et al., 2009) have used alternative formulations with stronger linear relaxations that can provide better lower bounds or improved integer rounding of fractional solutions.

Two such reformulations stand out in particular. The first is based on the Shortest Path (SP) problem and makes use of variable redefinition (Eppen and Martin, 1987). The second reformulation is based on the Facility Location (FL) problem (Krarup and Bilde, 1977).
Several reformulation results, both theoretical and computational, have been published. Nemhauser and Wolsey (1988) show that for the single item capacitated problem, the SP and FL reformulations are equivalent and that their linear relaxations coincide with the integer convex hull. In fact, their reformulation matrices are unimodular. Denizel et al. (2008) consider the capacitated multi-item problem with setup times, and also show that the SP and FL reformulations are equivalent. Alfieri et al. (2002) apply these reformulations to the same problem without setup times. The computational tests indicated that the SP and FL linear relaxations’ objective value is 60% larger than those of the classical formulation, but 30% larger if setup times are considered (Denizel and Süral, 2006). Diaby et al. (1992) use Lagrangean relaxation together with reformulation as a transport problem. Süral et al. (2009) consider an alternative formulation and also use Lagrangean relaxation to solve the problem. Jans and Degraeve (2004b) present a method to obtain lower bounds on the problem with setups using an SP-based formulation.

In addition, several researchers have proposed similar formulations for extensions of the classical lot sizing problem. Wu et al. (2011) propose two new mixed integer programming models for capacitated multi-level lot-sizing problems with backlogging, whose linear programming relaxations provide good lower bounds on the optimal solution value. Gramani and França (2006) consider the lot sizing problem integrated with the bidimensional cutting stock problem applied to the furniture industry, proposing a method based on an analogy with the SP problem.

In the same spirit, in section 3.1 an extended formulation (EF) is now developed from the original formulation (OF) using the facility location concept studied in Krarup and Bilde (1977) and Denizel et al. (2008). Additional EF constraints will also be proposed. The EF model is then strengthened in section 3.2 using setup considerations (Koçlar, 2005; Wolsey, 2002; Belvaux and Wolsey, 2001).

3.1. Extended Reformulation (EF)

The extended formulation (EF) can be obtained by interpreting the original formulation (OF) in a manner analogous to the Facility Location Problem (Krarup and Bilde, 1977). Let \( N = \{1, \ldots, \eta\} \) be the set of facility candidates and \( M = \{1, \ldots, T\} \) a set of clients. The Facility Location Problem (FLP) consists of deciding which facilities to install so that
demand is satisfied with minimum cost. To interpret the original formulation as an FLP, the production in subperiod \( n \) to satisfy demand in period \( t \) is taken as equivalent to using installed facility \( n \) to meet the demand of client \( t \). Thus subperiods represent facilities and periods represent clients.

The FLP-based extended formulation uses the same indices, parameters and variables as the original formulation in section 2, and an additional variable, \( Q_{pnt} \), the quantity of product \( p \) produced in subperiod \( n \) to meet the demand at the end of period \( t \). The production in a subperiod can have multiple destinations.

Note that if the production of subperiod \( n \) within period \( t \) meets demand for period \( t' > t \) after period \( t \), then it is held in inventory until the end of period \( t' - 1 \), with consequent costs over periods \( t, ..., t' - 1 \). Conversely, if the production of subperiod \( n \) within period \( t \) meets demand for period \( t'' < t \) before period \( t \), then the backlog costs over periods \( t'', ..., t - 1 \) must be charged.

Using \( \lambda_n \) to indicate the period of subperiod \( n \), the inventory and backlog costs \( CQ^+_{pnt} \) and \( CQ^-_{pnt} \) must be redefined as follows:

\( CQ^+_{pnt} \) is the unit cost of producing product \( p \) in subperiod \( n \) to meet demand at the end of period \( t \) after \( \lambda_n \). Note that \( CQ^+_{pnt} = \sum_{t=\lambda_n}^{t' - 1} h^+_{pt} \) (\( \lambda_n \leq t \)).

\( CQ^-_{pnt} \) is the unit cost of producing product \( p \) in subperiod \( n \) to meet backlogged demand at the end of period \( t \) before \( \lambda_n \). Note that \( CQ^-_{pnt} = \sum_{t=\lambda_n}^{t'' - 1} h^-_{pt} \) (\( \lambda_n \geq t \)).

The extended formulation (EF) is then given by the objective function (11) and constraints (12)-(15), as now explained.

**Extended Formulation (EF)**

\[
\text{Min} \quad \sum_{p=1}^{P} \left( \sum_{n=1}^{L_1} \sum_{t=\lambda_n}^{t' - 1} CQ^+_{p, \lambda_n} Q_{pnt} + \sum_{n=F_1}^{L_2} \sum_{t=1}^{t'' - 1} CQ^-_{p, t} Q_{pnt} \right) + \sum_{j=1}^{K} \sum_{k=1}^{L_j} \sum_{m=F_j}^{F_j} \sum_{l=1}^{L_k} s_{jk} z_{n}^{jk} \quad (11)
\]

Subject to:
\[
\sum_{n=1}^{L_1} Q_{pnt} = d_{pt} \quad \forall \ p, t \quad (12)
\]
\[
\sum_{p \in S(k)} \rho_p \left( \sum_{t=1}^{T} Q_{pnt} \right) + s t_{jk} z_{jk} \leq C y^k_n \quad \forall j, k, n = F_1, \ldots, L_T \tag{13}
\]
\[
Q_{pnt} \geq 0 \text{ and integer} \quad \forall p, t, n = F_1, \ldots, L_T \tag{14}
\]
and constraints (4)-(9) from the original formulation. \tag{15}

The objective function (11) again minimizes a weighted sum of inventory & backlog penalties and sequence-dependent setup penalties. Constraints (12) ensure that the demand in period \( t \) is met by the production from all subperiods. The variable \( Q \) is defined for \( \eta + 1 \) subperiods so as to permit backlogs in the last period. Note that the initial stock is zero. Constraints (13) are identical to (3) in the original formulation.

Although the number of variables has increased, previous research indicates such a reformulation generally improves the lower bounds obtained via linear relaxation (Chen and Thizy, 1990; Nemhauser and Wolsey, 1988 and Denizel et al., 2008).

### 3.2. Strengthening constraints for an Extended Formulation

This section now presents strengthening constraints based on setup considerations that can be included in an extended formulation (EF).

**Flow equations (V1)**

The setup variables \( y \) and changeover variables \( z \) are weakly linked in the linear relaxation of the extended formulation. The setup variables \( y \) can take positive values, thus allowing production, without the corresponding changeover variables \( z \) also being positive. For example: if \( y_{n-1}^j = 0.5 \) and \( y_n^k = 0.5 \) then constraint (4) means that \( z_{jk} \geq 0.5 + 0.5 - 1 = 0 \). This kind of solution causes a loosening of the lower bounds provided by the linear relaxation (as used in the Branch-and-Cut method).

The relationship between setup variables \( y \) and changeover variables \( z \) can be represented via a network in which the value associated with a node \( (j,n) \) indicates if there is a setup to material \( j \) in subperiod \( n \). The flow along an arc represents a changeover between two consecutive subperiods. The setup variable \( y_{n-1}^j \) indicates the
node capacity of node \((j,n-1)\) and the changeover variable \(z_n^{jk}\) indicates the flow from node \((j,n-1)\) to node \((k,n)\), as illustrated in Figure 1.

**[FIGURE 1: NEAR HERE]**

The constraint family \(V1\) presented below, based on Wolsey (2002), establishes a stronger link between the setup and changeover variables. When there is a setup to material \(k\) in a subperiod, then a changeover from another material to \(k\) must be enforced (constraint 16, \(V1a\)) along with a changeover from \(j\) to some other material (constraint 17, \(V1b\)), thus preserving the flow in the network.

\[
\begin{align*}
V1a: & \quad \sum_{j=1}^{K} z_n^{jk} = y_n^k, \quad k = 1, \cdots, K, \quad n = F_1, \cdots, L_f. \\
V1b: & \quad \sum_{k=1}^{K} z_n^{jk} = y_{n-1}^j, \quad j = 1, \cdots, K, \quad n = F_1 + 1, \cdots, L_f. 
\end{align*}
\]

**Elimination constraints (E1)**

Suppose that a solution has a sequence in a period in which the same material is processed in 2 lots in 2 non-consecutive subperiods. Given triangular setups, a no-worse and possibly better solution can be obtained by resequencing so that the subperiods are consecutive, thus eliminating a setup. In other words, there exists an optimal solution such that no material is setup more than once in a period. Constraints \(E1\) impose this:

\[
\begin{align*}
E1: & \quad \sum_{j=1}^{K} \sum_{k=1}^{L} z_n^{jk} \leq 1, \quad \forall k = 1, \cdots, K, \quad t = 1, \cdots, T.
\end{align*}
\]

**4. Reformulated Rolling Horizon Model**

The original formulation \(OF\) and the strengthened extended formulation \(EF+V1+E1\) (i.e., \(EF\) with the new constraints \(V1a, V1b\) and \(E1\)) were solved and tested with the Cplex 12.0 mixed integer programming (MIP) optimizer. As we will show in the computational results in section 6.3, the performance of Cplex 12.0 for solving the
formulations is unsatisfactory for medium-sized problems. As a result, this section proposes three metaheuristics to solve the EF model on a rolling-horizon basis.

As in Araujo et al. (2007), suppose each period $t$ is a workday. Consider, for example, a planning horizon of $T = 5$ workdays of which only the first day ($t = 1$) will be scheduled in detail. This is achieved by dividing the first day into $L_1 = 10$ subperiods, as up to $L$ material loadings can be processed each day. The remaining days $t = 2, \ldots, 5$ have just one subperiod each ($n_2 = n_3 = n_4 = n_5 = 1$). Thus $F_1 = 1; L_1 = 10; F_2 = L_2 = 11; F_3 = L_3 = 12; F_4 = L_4 = 13; F_5 = L_5 = 14$, i.e., there are $n = 14$ subperiods.

The variables $y_{jk}^n$ for the larger subperiods $n = F_2, \ldots, F_5$ are then redefined as “the number of loadings using material $k$ produced in subperiod $n$”.

Only the scheduled decisions relative to the $n_1 = 10$ subperiods of day 1 are actually implemented. The decisions for the remaining 4 days are used only to evaluate the impact of future available capacity, i.e., to identify a provisional production plan in order to have advance warning of possible production backlogs and be able to act accordingly. The extended formulation for rolling horizon use, denominated EF-RH, is:

**Model EF-RH:**

$$\text{Min} \sum_{p=1}^{F} \left( \sum_{n=1}^{T} \sum_{t=n}^{L} CF_{p,t}^n Q_{pmt} + \sum_{n=F_1}^{L} \sum_{t=1}^{L} CF_{p,t}^n Q_{pmt} \right) + \sum_{j=1}^{K} \sum_{k=1}^{L_1} \sum_{n=F_1}^{L_1} s_{jk} z_{jk}^n$$

Subject to:

$$\sum_{n=1}^{L_1} Q_{pmt} = d_{pt} \quad \forall p, t$$

$$\sum_{p \in S(k)} \rho_p \left( \sum_{t=1}^{T} Q_{pmt} \right) + st_{jk} z_{jk}^n \leq C y_{jk}^n \quad \forall j, k, n = F_1, \ldots, L_1$$

$$\sum_{p \in S(k)} \rho_p \left( \sum_{t=1}^{T} Q_{pmt} \right) \leq C y_{jk}^n \quad \forall j, k, n = F_2, \ldots, L_T$$

$$\sum_{k} y_{jk}^n = \frac{L}{\eta_t} \quad \forall t, n = F_1, \ldots, L_t$$

$$z_{jk}^n \geq y_{j,n-1}^k + y_{jk}^n - 1 \quad \forall j, k, n = F_1, \ldots, L_1$$

$$\sum_{j=1}^{K} z_{jk}^n = y_{jk}^n \quad \forall k, n = F_1, \ldots, L_1$$
\[
\sum_{j=1}^{K} z_{nj}^{jk} = y_{n-1}^{j} \quad \forall j, n = F_{1}+1,...,L_{1}
\]

(26)

\[
\sum_{j=1}^{K} \sum_{k} z_{nj}^{jk} \leq 1 \quad \forall k
\]

(27)

\[
y_{nk}^{k} \in \{0,1\} \quad \text{with } y_{n0}^{k} = 0
\]

(28)

\[
y_{nk}^{k} \geq 0 \text{ and integer} \quad \forall k, n = F_{1},...,L_{1}
\]

(29)

\[
z_{nj}^{jk} \geq 0 \quad \forall j, k, n = F_{1},...,L_{1}
\]

(30)

\[
Q_{pnt} \geq 0 \text{ and integer} \quad \forall p, t, n = F_{1},...,L_{1}
\]

(31)

\[
Q_{pnt} \geq 0 \quad \forall p, t, n = F_{2},...,L_{T}
\]

(32)

\[
I_{pt}^{*} \text{ and } I_{pt} \geq 0 \quad \forall p, t
\]

(33)

Constraints (13) are now replaced by (21) for the first period and (22) for the remaining periods. As in Araujo et al. (2007), constraints (5) are replaced by (23) which impose exactly \(L/\eta_{t}\) setups in subperiod \(n\), i.e., the number of loads in period \(t\) divided by the number of subperiods in period \(t\). For example, period 1 has 10 subperiods and can handle 10 loads, so \(L/\eta_{1} = 10/10 = 1\), whereas period 2 has one subperiod and can handle 10 loads, so \(L/\eta_{2} = 10/1 = 10\). Constraints (24), as well as the strengthening constraints (25), (26) and (27) are now restricted to the first period.

5. Metaheuristics for the Extended Formulation (EF)

The models were solved on a rolling horizon basis in two steps using the relax-and-fix method, as follows:

Relax all integer variables, except the first day’s binary variables \(y_{n}^{k}\) for \(n = F_{1},...,L_{1}\), these being the most important decisions in the rolling horizon method. Solve this relaxed problem.

Fix the first day’s binary variables \(y_{n}^{k}\) at their solution values in step 1. Retighten to be integer the \(y_{n}^{k}\) variables for the subsequent days \((n = F_{2},...,L_{T})\) and the \(x_{pm}\) variables for the first day \((n = F_{1},...,L_{1})\). Solve this partially fixed problem.
Note that solving the model on a rolling horizon basis means that the initial setup state is zeroed as we proceed through the rolling horizon. This can result in slightly different solutions to the Cplex solution of the OF and EF formulations.

Step 1 is still time-consuming to solve optimally, so a solution is obtained using one of the three metaheuristics methods described in this section.

Step 2 can be optimally solved in a few seconds with Cplex, since a binary variable $y_n^k$ which is fixed to 1 implies that $Q_{pn} = 0$ for all $p \not\in S(k)$, i.e., products that do not use material $k$ are not manufactured in subperiod $n$, thus eliminating many integer variables and constraints.

Consequently, the three solution methods developed in the rest of this paper focus on step 1 and are now briefly described. Details can be found in Araujo et al. (2007)

**Descent Heuristic (DH)**

Step 1 of the relax-and-fix method can be solved using a local search descent heuristic (Araujo et al., 2007) to find the values of the first day's binary variables $y_n^k$ for $n = F_1,...,L_1$. The search starts with a random initial solution whose objective function value is obtained by solving the linear programme (LP) that results from fixing the values of the binary variables. The $y_n^k$ solution values for day 1 are then randomly modified to provide a neighbouring solution whose objective function value is then calculated via the dual simplex method to efficiently resolve the modified LP. If it is an improvement, the neighbouring solution is adopted as the current one. Experiments showed that 1000 iterations are sufficient to obtain a good solution within an acceptable running time.

**Diminishing Neighbourhood (DN) Search**

This method adapts the local search descent heuristic described above by beginning with a large neighbourhood to encourage diversity, and then gradually diminishing its size so as to increasingly intensify the search. The search starts with the largest possible neighbourhood and after a certain number of iterations the neighbourhood size is reduced. The search ends with the smallest neighbourhood where just one position is changed. 1000 iterations in total were made.
**Simulated Annealing (SA)**

Simulated Annealing (Eglese, 1990) is a variant of the local search descent heuristic that tries to avoid getting trapped at a local optimum by permitting worsening moves away with probability:

\[ p(\Delta ofv) = e^{-\frac{\Delta ofv}{\text{Temp}}} \]  

(34)

where \( \text{Temp} \) is a gradually-cooling “temperature” and \( \Delta ofv \) the amount by which the new move worsens the objective value of the solution. As the search progresses, the best solution encountered is recorded. 1000 iterations in total were made. The parameters used in the tests below were the same as in Araujo et al. (2007).

6. **Computational Results**

This section first describes how the test data was generated. It then goes on to analyse the use of the Cplex 12.0 solver against the metaheuristics from Araujo et al. (2007) on the Original Formulation (OF). Then Cplex 12.0 is compared against the metaheuristics from Araujo et al. (2007) under the Extended Formulation incorporating the a priori constraints (EF + V1+ E1).

The models were implemented in the AMPL modelling language (Fourer et al., 2002). The optimizer Cplex 12.0 used its default parameters. The tests were carried out on an Intel Core 2 Duo CPU P8400 at 2.26 GHz 2.27 GHz with 3.0 GB of RAM, under Window Vista Business.

6.1 **Data Generation**

The data set is presented in Table 1. It is the same as that generated by Araujo et al. (2007). The instances have 10, 50 and 100 products (small, medium and large, respectively). The capacity can be (loose, tight or too tight) and setup costs are high or low, making a total of 3×3×2=18 combinations. Ten replications were run for each of the 18 combinations. The planning horizon extends over 5 periods, each divided into
10 subperiods.

[TABLE 1: NEAR HERE]

6.2. Comparing Cplex on the OF formulation against the metaheuristics on the OF-RH formulation.

Table 2 compares the objective function value resulting from an hour of Cplex 12.0 solving time applied to the original formulation (OF) over the values obtained by the Descent Heuristic (DH), Diminishing Neighbourhood Search (DN) and Simulated Annealing (SA).

The heuristic procedures are applied on the rolling horizon-based model RH proposed in Araujo et al. (2007). In this paper this model will be called OF-RH. A solution to model OF-RH is not a solution to model OF (1)-(9), as just the first loadings are scheduled and actually implemented, while the other days are planned only approximately. However, the application $T$ times of model OF-RH, starting consecutively at periods 1, 2, ..., 5, with an always-shortening horizon ($T = 5, 4, 3, 2, 1$), will provide a feasible solution to model (1)-(9), enabling a comparison of results.

The table shows the percentage mean difference between the Cplex solution and the indicated metaheuristic, calculated as:

$$Difference = \left( \frac{\text{Heuristic solution value} - \text{Cplex solution value}}{\text{Cplex solution value}} \right) \times 100\%.$$  

Thus a positive difference value provides a % measure of how much better the Cplex solution is compared to the heuristic solution. Conversely, a negative difference value indicates how much worse the Cplex solution is compared to the heuristic one.

[TABLE 2: NEAR HERE]
Note from Table 2 that Cplex generally outperforms the metaheuristics for small and medium sizes problems, but certainly not for large problems where the metaheuristics perform far better. This is consistent with the generally acknowledged advantage that metaheuristics show over mathematical programming for large instances.

6.3. Using Cplex to compare the Original Formulations (OF) and Extended Formulations (EF+V1+E1)

Tables 3 to 5 show the results using Cplex on both the original formulation (OF) and the extended one (EF+V1+E1). The values shown are the means from 10 runs for each of the 18 combinations in Table 1 using Cplex for up to an hour, along with upper (UB) and lower (LB) bounds. The gap shown is the percentage mean difference between the upper and lower bounds UB and LB. The nodes and cuts columns show respectively the number of nodes and cuts used by Cplex up to the time limit.

[TABLE 3: NEAR HERE]

Note that for small instances (Table 3) the results are practically identical, while for medium-sized instances (Table 4), the EF always provided slightly better solutions. For large instances (Table 5), the EF always provided much better solutions. These test results clearly illustrate that the extended formulation is more efficient and thus more effective than the original formulation. However, note that, for medium and large problems, the gaps obtained by Cplex are still large, being more than 60% on average, even for the tighter EF+V1+E1 reformulation.
6.4. Comparing Cplex on the EF+V1+E1 formulation against the metaheuristics on the OF-RH formulation.

Table 6 shows the differences between Cplex 12.0 on the EF+V1+E1 formulation and the metaheuristics on the OF-RH formulation. Note that the EF+V1+E1 formulation generally outperforms the metaheuristics in small and medium problems, but not for large instances, an observation very similar to that in section 6.2.

6.5. Comparing Cplex on the EF+V1+E1 formulation against the metaheuristics on the EF-RH+V1+E1 formulation.

Considering the good results obtained after the resolution of the reformulation with the Cplex 12.0, the metaheuristics from Araujo et al. (2007) were adapted for solving the EF+V1+E1 reformulation. The results are shown in Table 7. Observe that now the metaheuristics generally outperforms the EF+V1+E1 formulation in small and medium problems, and very clearly so for large instances. These results are the really interesting ones, showing that metaheuristics are very competitive even against a strengthened mathematical programming reformulation. Comparing the three
metaheuristics, note that simulated annealing (SA) performs best, followed by diminishing neighbourhood (DN).

[TABLE 7: NEAR HERE]

7. Conclusions.

This paper considered a lot sizing model that sequences the preparation of a key material and decides the production lot sizes of the final products. This class of problem is very prevalent in the literature, using GLSP formulations. However, recent ATSP-inspired formulations have shown better performance. Aiming to tighten the linear relaxation of the GLSP-type model in Araujo et al. (2007), a reformulation as a facility location problem was extended with setup elimination constraints and flow equations, strengthening the link between setup and changeover variables. Three different metaheuristics and an industry-standard optimizer were then successfully applied to accelerate and improve the solution of both the original and extended formulations, particularly applying simulated annealing to the latter.

Computational tests showed that on the larger instances the metaheuristics outperformed Cplex and the reformulated model was more effective than the original model. Jointly applied, the metaheuristics and the extended formulation were effective on both small and large sized instances, particularly the latter where metaheuristics really show their advantage.

Future work will extend and adapt the metaheuristics to the case where multiple machines process the raw materials, the predominant case in many industrial processes. Considering the prevalence of this type of problem, future research should analyse and compare the GLSP and ATSP formulations, both theoretically and computationally.
References


Table 1: Parameters used for generating uniformly-distributed test data.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Products and Materials: ((P, K)) pairs</td>
<td>Small Problem: ((10, 2))</td>
</tr>
<tr>
<td></td>
<td>Medium Problem: ((50, 10))</td>
</tr>
<tr>
<td></td>
<td>Large Problem: ((100, 20))</td>
</tr>
<tr>
<td>Number of Periods ((T))</td>
<td>5</td>
</tr>
<tr>
<td>Inventory Penalty ((h^+_{pt}))</td>
<td>(U[2, 10])</td>
</tr>
<tr>
<td>Backlog Penalty ((h^-_{pt}))</td>
<td>(U[20, 100])</td>
</tr>
<tr>
<td>Necessary capacity to produce one unit of product (p) ((\rho_p))</td>
<td>(U[0.1, 3])</td>
</tr>
<tr>
<td>Setup Time ((st_{jk}))</td>
<td>(U[5, 10])</td>
</tr>
<tr>
<td>Setup Penalty ((s_{jk}))</td>
<td>Low: (50 \times st_{jk})</td>
</tr>
<tr>
<td></td>
<td>High: (500 \times st_{jk})</td>
</tr>
<tr>
<td>Demand ((d_{pt}))</td>
<td>(U[40, 60])</td>
</tr>
<tr>
<td>Tightness of Capacity ((C))</td>
<td>Loose: (LFLC / 0.6)</td>
</tr>
<tr>
<td>Note: (LFLC = \sum_{p=1}^{P} \sum_{t=1}^{T} d_{pt} \rho_p / L_T)</td>
<td>Tight: (LFLC / 0.8)</td>
</tr>
<tr>
<td></td>
<td>Too Tight: (LFLC / 1.0)</td>
</tr>
</tbody>
</table>
## Table 2: Difference (%) between Cplex and the metaheuristics, both on OF

<table>
<thead>
<tr>
<th>Methods</th>
<th>DH</th>
<th>DN</th>
<th>SA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Setup Cost</strong></td>
<td>Small</td>
<td>Large</td>
<td>Small</td>
</tr>
<tr>
<td><strong>Problem</strong></td>
<td><strong>Capacity</strong></td>
<td><strong>Setup Cost</strong></td>
<td><strong>Setup Cost</strong></td>
</tr>
<tr>
<td>10 products</td>
<td>Loose</td>
<td>25.82</td>
<td>66.46</td>
</tr>
<tr>
<td></td>
<td>Tight</td>
<td>90.73</td>
<td>60.25</td>
</tr>
<tr>
<td>50 products</td>
<td>Loose</td>
<td>30.01</td>
<td>32.35</td>
</tr>
<tr>
<td></td>
<td>Tight</td>
<td>28.77</td>
<td>15.05</td>
</tr>
<tr>
<td></td>
<td>Too Tight</td>
<td>27.84</td>
<td>6.73</td>
</tr>
<tr>
<td>100 products</td>
<td>Loose</td>
<td>-94.04</td>
<td>-91.69</td>
</tr>
<tr>
<td></td>
<td>Tight</td>
<td>-86.14</td>
<td>-88.03</td>
</tr>
<tr>
<td></td>
<td>Too Tight</td>
<td>-75.31</td>
<td>-82.72</td>
</tr>
</tbody>
</table>
Table 3: Tests with 10-product instances

<table>
<thead>
<tr>
<th>Problem</th>
<th>OF Formulation</th>
<th>UB</th>
<th>LB</th>
<th>Gap (%)</th>
<th>Nodes</th>
<th>Cuts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity</td>
<td>Setup Cost</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Loose</td>
<td>Small</td>
<td>1,961</td>
<td>1,961</td>
<td>0.00</td>
<td>2,940</td>
<td>133</td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>13,040</td>
<td>13,034</td>
<td>0.05</td>
<td>6,845,657</td>
<td>145</td>
</tr>
<tr>
<td>Tight</td>
<td>Small</td>
<td>1,961</td>
<td>1,961</td>
<td>0.00</td>
<td>1,278</td>
<td>146</td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>14,405</td>
<td>14,382</td>
<td>0.16</td>
<td>11,696,057</td>
<td>150</td>
</tr>
<tr>
<td>Too Tight</td>
<td>Small</td>
<td>6,262</td>
<td>6,015</td>
<td>3.94</td>
<td>11,149,917</td>
<td>282</td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>22,575</td>
<td>22,359</td>
<td>0.96</td>
<td>12,292,987</td>
<td>197</td>
</tr>
<tr>
<td>Mean</td>
<td></td>
<td>10,034</td>
<td>9,952</td>
<td>0.82</td>
<td>6,998,134</td>
<td>175</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Problem</th>
<th>EF + V1 + E1 Formulation</th>
<th>UB</th>
<th>LB</th>
<th>Gap (%)</th>
<th>Nodes</th>
<th>Cuts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity</td>
<td>Setup Cost</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Loose</td>
<td>Small</td>
<td>1,961</td>
<td>1,961</td>
<td>0.00</td>
<td>2,198</td>
<td>477</td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>13,040</td>
<td>12,401</td>
<td>4.90</td>
<td>1,806,223</td>
<td>567</td>
</tr>
<tr>
<td>Tight</td>
<td>Small</td>
<td>1,961</td>
<td>1,961</td>
<td>0.00</td>
<td>776</td>
<td>229</td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>14,408</td>
<td>14,383</td>
<td>0.17</td>
<td>4,493,654</td>
<td>323</td>
</tr>
<tr>
<td>Too Tight</td>
<td>Small</td>
<td>6,327</td>
<td>6,027</td>
<td>4.74</td>
<td>3,186,267</td>
<td>377</td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>22,816</td>
<td>22,368</td>
<td>1.96</td>
<td>3,459,528</td>
<td>340</td>
</tr>
<tr>
<td>Mean</td>
<td></td>
<td>10,085</td>
<td>9,850</td>
<td>2.33</td>
<td>2,158,107</td>
<td>386</td>
</tr>
</tbody>
</table>
Table 4: Tests with the 50-product instances

<table>
<thead>
<tr>
<th>Problem</th>
<th>OF Formulation</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Capacity</td>
<td>Setup Cost</td>
<td>UB</td>
<td>LB</td>
<td>Gap (%)</td>
<td>Nodes</td>
<td>Cuts</td>
<td></td>
</tr>
<tr>
<td>Loose</td>
<td>Small</td>
<td>21,154</td>
<td>1,900</td>
<td>91.01</td>
<td>13,314</td>
<td>1,901</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>122,272</td>
<td>2,882</td>
<td>97.64</td>
<td>11,253</td>
<td>2,109</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tight</td>
<td>Small</td>
<td>33,563</td>
<td>8,002</td>
<td>76.16</td>
<td>8,311</td>
<td>2,077</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>147,141</td>
<td>8,255</td>
<td>94.39</td>
<td>7,270</td>
<td>2,272</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Too Tight</td>
<td>Small</td>
<td>93,839</td>
<td>42,425</td>
<td>54.79</td>
<td>14,329</td>
<td>1,381</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>212,540</td>
<td>43,799</td>
<td>79.39</td>
<td>11,578</td>
<td>1,396</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>105,085</td>
<td>17,877</td>
<td>82.99</td>
<td>11,009</td>
<td>1,856</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Problem</th>
<th>EF + V1 + E1 Formulation</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Capacity</td>
<td>Setup Cost</td>
<td>UB</td>
<td>LB</td>
<td>Gap (%)</td>
<td>Nodes</td>
<td>Cuts</td>
<td></td>
</tr>
<tr>
<td>Loose</td>
<td>Small</td>
<td>19,974</td>
<td>4,912</td>
<td>75.41</td>
<td>12,193</td>
<td>2,938</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>117,433</td>
<td>27,807</td>
<td>76.32</td>
<td>9,591</td>
<td>3,244</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tight</td>
<td>Small</td>
<td>30,515</td>
<td>11,400</td>
<td>62.64</td>
<td>9,252</td>
<td>3,067</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>132,776</td>
<td>33,846</td>
<td>74.51</td>
<td>6,673</td>
<td>3,406</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Too Tight</td>
<td>Small</td>
<td>90,254</td>
<td>46,183</td>
<td>48.83</td>
<td>7,524</td>
<td>3,318</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>193,250</td>
<td>68,028</td>
<td>64.80</td>
<td>5,228</td>
<td>3,428</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>97,367</td>
<td>32,029</td>
<td>67.10</td>
<td>8,410</td>
<td>3,233</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 5: Tests with the 100-product instances

<table>
<thead>
<tr>
<th>Problem</th>
<th>OF Formulation</th>
<th>Capacity</th>
<th>Setup Cost</th>
<th>UB</th>
<th>LB</th>
<th>Gap (%)</th>
<th>Nodes</th>
<th>Cuts</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Small</td>
<td></td>
<td>3,994,738</td>
<td>35,373</td>
<td>99.11</td>
<td>124</td>
<td>3,085</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Large</td>
<td></td>
<td>3,566,167</td>
<td>102,836</td>
<td>97.12</td>
<td>185</td>
<td>39,101</td>
</tr>
<tr>
<td>Loose</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tight</td>
<td></td>
<td>Small</td>
<td></td>
<td>2,561,219</td>
<td>61,540</td>
<td>97.60</td>
<td>225</td>
<td>2,916</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Large</td>
<td></td>
<td>4,135,672</td>
<td>48,497</td>
<td>98.83</td>
<td>127</td>
<td>2,916</td>
</tr>
<tr>
<td>Too Tight</td>
<td></td>
<td>Small</td>
<td></td>
<td>2,316,719</td>
<td>220,323</td>
<td>90.49</td>
<td>290</td>
<td>220,060</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Large</td>
<td></td>
<td>3,869,352</td>
<td>220,060</td>
<td>94.31</td>
<td>227</td>
<td>2,454</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td></td>
<td></td>
<td>3,407,311</td>
<td>114,771</td>
<td>96.63</td>
<td>2041</td>
<td>38,923</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Problem</td>
<td>FE + V1 + E1 Formulation</td>
<td>Capacity</td>
<td>Setup Cost</td>
<td>UB</td>
<td>LB</td>
<td>Gap (%)</td>
<td>Nodes</td>
<td>Cuts</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Small</td>
<td></td>
<td>140,947</td>
<td>97,523</td>
<td>30.81</td>
<td>723</td>
<td>5,324</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Large</td>
<td></td>
<td>295,012</td>
<td>117,271</td>
<td>60.25</td>
<td>675</td>
<td>5,742</td>
</tr>
<tr>
<td>Loose</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tight</td>
<td></td>
<td>Small</td>
<td></td>
<td>718,120</td>
<td>131,557</td>
<td>81.68</td>
<td>442</td>
<td>5,074</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Large</td>
<td></td>
<td>326,927</td>
<td>151,379</td>
<td>53.60</td>
<td>559</td>
<td>5,647</td>
</tr>
<tr>
<td>Too Tight</td>
<td></td>
<td>Small</td>
<td></td>
<td>612,921</td>
<td>293,273</td>
<td>52.15</td>
<td>149</td>
<td>4,087</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Large</td>
<td></td>
<td>983,102</td>
<td>321,127</td>
<td>67.33</td>
<td>386</td>
<td>5,035</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td></td>
<td></td>
<td>512,838</td>
<td>185,355</td>
<td>63.86</td>
<td>1,975</td>
<td>4,735</td>
</tr>
</tbody>
</table>
Table 6: Difference (%) between Cplex on EF+V1+E1 and the metaheuristics on OF-RH

<table>
<thead>
<tr>
<th>Methods</th>
<th>Setup Cost</th>
<th>DH</th>
<th>DN</th>
<th>SA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem</td>
<td></td>
<td>Small</td>
<td>Large</td>
<td>Small</td>
</tr>
<tr>
<td>Capacity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Loose</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 products</td>
<td>Tight</td>
<td>90.73</td>
<td>60.21</td>
<td>12.89</td>
</tr>
<tr>
<td></td>
<td>Too Tight</td>
<td>61.80</td>
<td>30.97</td>
<td>-19.95</td>
</tr>
<tr>
<td>Loose</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50 products</td>
<td>Tight</td>
<td>41.63</td>
<td>27.50</td>
<td>29.40</td>
</tr>
<tr>
<td></td>
<td>Too Tight</td>
<td>32.92</td>
<td>17.38</td>
<td>21.43</td>
</tr>
<tr>
<td>Loose</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100 products</td>
<td>Tight</td>
<td>-50.57</td>
<td>51.43</td>
<td>-51.60</td>
</tr>
<tr>
<td></td>
<td>Too Tight</td>
<td>-6.67</td>
<td>-31.98</td>
<td>-10.51</td>
</tr>
</tbody>
</table>
Table 7: Difference (%) between Cplex on EF+V1+E1 and the metaheuristics on EF-RH+V1+E1

<table>
<thead>
<tr>
<th>Methods</th>
<th>DH</th>
<th>DN</th>
<th>SA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem</td>
<td>Setup Cost</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 products</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Loose</td>
<td>Small</td>
<td>Large</td>
<td>-4.44</td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tight</td>
<td>8.96</td>
<td>58.59</td>
<td>-4.44</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Too Tight</td>
<td>162.86</td>
<td>8.33</td>
<td>-66.92</td>
</tr>
<tr>
<td>50 products</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Loose</td>
<td>Small</td>
<td>Large</td>
<td>-43.27</td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tight</td>
<td>79.48</td>
<td>-29.62</td>
<td>1.92</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Too Tight</td>
<td>77.94</td>
<td>17.71</td>
<td>78.47</td>
</tr>
<tr>
<td>100 products</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Loose</td>
<td>-27.60</td>
<td>-58.15</td>
<td>-32.38</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tight</td>
<td>67.26</td>
<td>2.03</td>
<td>-79.87</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Too Tight</td>
<td>31.83</td>
<td>-37.64</td>
<td>-50.71</td>
</tr>
</tbody>
</table>
\[ y_{n-1}^j = \text{Flow through node} = 0 \text{ or } 1 \]

\[ y_n^k = \text{Flow through node} = 0 \text{ or } 1 \]

\[ z_{n}^{jk} = \text{Flow along arc} = 0 \text{ or } 1 \]
Highlights

- A lot sizing and sequencing model with prepared materials is proposed.
- The prepared materials have sequence-dependent setup costs and times.
- Reformulation and strengthened constraints improve the model.
- Three alternative metaheuristics are used to fix the setup variables.
- The metaheuristics greatly improve performance.