Available online at www.sciencedirect.com
ScienceDirect
Computers & Operations Research 35 (2008) 3644 – 3656
computers & operations research
www.elsevier.com/locate/cor

Rolling-horizon and fix-and-relax heuristics for the parallel machine lot-sizing and scheduling problem with sequence-dependent set-up costs

Patrizia Beraldi[a,*], Gianpaolo Ghiani[b], Antonio Grieco[b], Emanuela Guerriero[b]

[a] *Dipartimento di Elettronica, Informatica e Sistemistica and Excellence Center on High Performance Computing, Università degli Studi della Calabria, 87030 Rende (CS), Italy*

[b] *Dipartimento di Ingegneria dell'Innovazione, Università degli Studi di Lecce, 73100 Lecce, Italy*

Available online 19 April 2007

Abstract

In this paper we develop new rolling-horizon and fix-and-relax heuristics for the identical parallel machine lot-sizing and scheduling problem with sequence-dependent set-up costs. Unlike previous papers, our procedures are based on a compact formulation relying on the hypotheses of identical machines. This feature makes our approach suitable for large-scale applications (with hundreds of machines) arising in the textile and fiberglass industries. Moreover, our procedures are shown to provide a feasible solution for any feasible instance. Comparisons with lower bounds provided by a truncated branch-and-bound show that the gap between the best heuristic solution and the lower bound never exceeds 3%.

Keywords: Lot-sizing and scheduling; Production planning; Rolling-horizon; Fix-and-relax

1. Introduction

Lot-sizing and scheduling problems have been an area of active research starting from the seminal paper of Wagner and Whitin [1]. Since then there has been a considerable amount of investigation in order to incorporate other important features such as backlogging, capacities, multiple items, multiple machines, multiple stages, etc. See Drexl and Kimms [2] and Staggemeier and Clark [3] for two recent surveys, and Pochet [4] for an up-to-date tutorial. In spite of this research effort, the body of knowledge on almost all practical problems is still unsatisfactory.

This paper deals with the lot-sizing and scheduling problem with identical parallel machines (LSPIPM) and sequence-dependent set-up costs, whose applications arise in the textile and fiberglass industries (see, e.g., [5]).

In literature, references have been mainly focused on the *heterogeneous* machine problem, which is a more general case with respect to the parallel machine problem. Clark [6] has developed a very fast myopic rule-based heuristic for rolling-horizon (RH) lot-sizing and sequencing on a set of parallel heterogeneous machines with sequence-dependent set-up times. In this paper, a single set-up is assumed at the beginning of each period while in Clark and Clark [7]

* Corresponding author.
E-mail addresses: beraldi@unical.it (P. Beraldi), gianpaolo.ghiani@unile.it (G. Ghiani), antonio.grieco@unile.it (A. Grieco), emanuela.guerriero@unile.it (E. Guerriero).

0305-0548/$ - see front matter © 2007 Published by Elsevier Ltd.
doi:10.1016/j.cor.2007.04.003

multiple set-ups per planning period are allowed. Meyr [8] has developed a model for simultaneous lot-sizing and scheduling on heterogeneous parallel machines with sequence-dependent set-up times and no backlogging. For each machine the planning horizon is divided into a predetermined number of periods which contain at most one set-up. By extending previous findings on the single machine lot-sizing and scheduling problem [9,10], this paper describes a local search over the set-up sequences, coupled with dual re-optimization of lot sizes. Instances with up to 19 products, 2 machines and 8 periods are considered. Finally, Staggemeier et al. [11] describe a genetic algorithm in which optimal lot sizes are determined through linear programming.

The lot-sizing and scheduling problem is also referred to in the literature as *parallel machine scheduling problem with splitting jobs* [12–14]. Indeed, a job can be split into an arbitrary number of parts and each part processed on a different machine. This problem arises, for example, in the textile industry where a job represents a batch (for example, a batch of 1500 socks) and the job splitting property allows decomposing each batch into sub-batches.

The algorithms described in the literature have been developed and tested on applications characterized by a small number of machines (up to five) [15]. On the contrary, in this paper we have developed algorithms suitable for large-scale applications (with hundreds of machines) arising in the textile and fiberglass industries. Our objective is to minimize the total set-up cost. Indeed, in some applications, set-up costs are directly proportional to set-up times and, consequently, schedules optimal with respect to set-up times are also optimal with respect to set-up costs. In many other cases, particularly when set-up operations require high-skilled labor force, set-up costs are relatively high while set-up times are negligible. Firstly, we have demonstrated the equivalence between the given formulation and the ILP model proposed in Sumischrast and Frendewey [15]. Then we have developed new decomposition heuristics based on RH and fix-and-relax (FR) approaches. In Sumischrast and Frendewey [15], the authors decomposed the original problem in a set of smaller subproblems, each of them solved by applying the heuristic procedure MINSET. The heuristic methods proposed in this paper are based on a decomposition approach too. However, our solution procedures outperform the MINSET method, since they adopt an exact method to solve each subproblem.

The paper is organized as follows. Section 2 presents a compact formulation of the identical parallel machine case. In Sections 3 and 4 RH and FR heuristics are introduced and shown to provide a feasible solution for any feasible instance. The performances of these approaches have been numerically tested on a variety of medium and large size problems. The results have been reported in Section 5. Finally, conclusions and future research directions follow in Section 6.

2. Formulation

In the LSPIPM, we are given m parallel identical machines and n products (or part types) to be manufactured over a discrete planning horizon $\{1, \ldots, T\}$. As customary in the textile and fiberglass applications which motivated our work, we assume that machine changeovers are associated with sequence-dependent set-up costs while set-up idle times are negligible. This is often the case whenever the cost for switching between certain products is relatively high (e.g., because of high-skilled labor requirement) even though the switching time is relatively less [16]. Let c_{ij} be the cost to switch from product i to j ($c_{ii} = 0$, $i = 1, \ldots, n$) and let m_i be the number of machines initially set up to produce product i ($\sum_{i=1,\ldots,n} m_i = m$). Moreover, demands and production capacity are both assumed to be known and no backlog is allowed. Let d_{it} be the demand for product i in period t and let a_i be the production rate of product i. We formulate the LSPIPM as follows:

$$\text{Minimize} \quad \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T} c_{ij}\, y_{ijt} \tag{1}$$

s.t.

$$x_{i1} - \sum_{\substack{j=1\\ j\neq i}}^{n} y_{ji1} + \sum_{\substack{j=1\\ j\neq i}}^{n} y_{ij1} = m_i, \quad i = 1, \ldots, n, \tag{2}$$

$$x_{it} - \sum_{\substack{j=1\\ j\neq i}}^{n} y_{jit} + \sum_{\substack{j=1\\ j\neq i}}^{n} y_{ijt} = x_{it-1}, \quad i = 1, \ldots, n; \quad t = 2, \ldots, T, \tag{3}$$

$$\sum_{t=1}^{\ell} x_{it} \geqslant \sum_{t=1}^{\ell} D_{it}, \quad i=1,\dots,n; \quad \ell=1,\dots,T, \tag{4}$$

$$x_{it} \geqslant 0, \text{ integer}, \quad i=1,\dots,n; \quad t=1,\dots,T, \tag{5}$$

$$y_{ijt} \geqslant 0, \quad i, \; j=1,\dots,n; \;\; t=1,\dots,T, \tag{6}$$

where $D_{it} = \lceil d_{it}/a_i \rceil$ is the number of time periods required to produce d_{it} on a single machine; x_{it} is the number of machines producing part type i in period t; y_{ijt} is the number of machines switched from part type i to part type j at the beginning of period t. Constraints (2) and (3) express the relation between x_{it} and y_{ijt} variables. Constraints (4) state that the number of machines assigned to part type i in periods $1,\dots,\ell$ is sufficient. Constraints (5) and (6) are the usual non-negativity and integrality constraints.

The formulation (1)–(6) can be obtained from the ILP model proposed in [15], by relaxing the integrality constraints on Y's variables. The validity of formulation (1)–(6) stands on the following proposition.

Proposition 1. *For a given set of x_{it} variables satisfying constraints* (4) *and* (5), *there exists an optimal solution of formulation* (1)–(3), (6) *in which y_{ijt} variables are integers.*

Proof. This result follows from the observation that the coefficient matrix associated with (2) and (3) is totally unimodular and the right-hand sides m_i and x_{it} are integers. □

As a result, no integrality constraints need to be imposed explicitly on y_{ijt} variables. Compared to formulations for the heterogeneous machine case [7,8], model (1)–(6) contains much less integer variables. Indeed the proposed formulation is characterized by $O(nT)$ general integer variables against the $O(n^2mT)$ binary variables in (Meyer, 2002). This feature makes the linear programming lower bounds provided by (1)–(6) much tighter than those reported in the literature for the heterogeneous machine case.

Unfortunately, when fed into a commercial mixed integer programming (MIP) solver, formulation (1)–(6) allows solving only small-sized instances (e.g., instances with up to $n=10$ and $T=10$). In the next three sections, new RH and FR heuristics, specifically tailored for large-scale identical parallel machine instances (like those arising in the textile and fiberglass industries) are described and computationally assessed.

3. The RH heuristic

RH heuristics are usually used in dynamic lot-sizing and scheduling problems, where demands are gradually revealed during the planning horizon and part types have to be allocated to machines in an on-going fashion as new orders arrive (see, e.g., [17,18]). On the other hand, a RH approach is still suitable when product demand is perfectly known. In this case, a large multi-stage problem is decomposed into a number of smaller subproblems. The limited size of these subproblems allows using exact methods for their solution, which would be impossible for the overall problem. The number and the size of the subproblems define the computational burden and the solution quality of the heuristic procedures.

More formally, the RH methodology partitions the planning horizon into k time intervals $[t_r, T_r]$ $(r=1,\dots,k)$ such that $t_1=1$, $t_r=T_{r-1}+1$ $(r=2,\dots,k)$ and $T_k=T$. For each time interval $[t_r, T_r]$, we solve a subproblem $LSPIPM^r_{RH}$ consisting of formulation (1)–(6) over $[t_r, T_r]$, subject to an additional set of constraints C^r on the x_{it} variables. Such constraints are generated on the basis of the solution values of subproblems $LSPIPM^1_{RH},\dots,LSPIPM^r_{RH}$.

Our implementations of the RH concept are designed in such a way that subproblems $LSPIPM^r_{RH}$ $(r=1,\dots,k)$ are always feasible if the LSPIPM instance is feasible, as proved subsequently.

Given the solution values $\hat{x}^{r-1}_{it}$ and $\hat{y}^{r-1}_{ijt}$ of subproblem $LSPIPM^{r-1}_{RH}$, let $\hat{D}^r_{it}$ be the demand for product i due on period t, unsatisfied at time t_r $(t \geqslant t_r)$:

$$\hat{D}^1_{it} = \sum_{\ell=1}^{t} D_{i\ell}, \quad i=1,\dots,n; \;\; t=1,\dots,T_1, \tag{7}$$

$$\hat{D}_{it}^r = \sum_{\ell=1}^{t} D_{i\ell} - \sum_{\ell=1}^{T_{r-1}} \hat{x}_{i\ell}^{r-1}, \quad r = 2, \ldots, k; \;\; i = 1, \ldots, n; \;\; t = t_r, \ldots, T_r. \tag{8}$$

Therefore, in subproblem $LSPIPM_{RH}^r$ constraints (3) are replaced by

$$x_{it_r} - \sum_{\substack{j=1 \\ j \neq i}}^{n} y_{jit_r} + \sum_{\substack{j=1 \\ j \neq i}}^{n} y_{ijt_r} = \hat{x}_{iT_{r-1}}, \quad r > 2; \;\; i = 1, \ldots, n, \tag{9}$$

and inequalities (4) are replaced by

$$\sum_{t=t_r}^{\ell} x_{it} \geqslant \sum_{t=t_r}^{\ell} \hat{D}_{it}^r, \quad 1 \leqslant r \leqslant k; \;\; i = 1, \ldots, n; \;\; \ell = t_r, \ldots, T_r. \tag{10}$$

We have developed two distinct RH procedures, named RH1 and RH2 in the following. The two heuristics differ in the definition of the set of additional constraints C^r on the x_{it} variables.

3.1. RH1 heuristic

Let L_{ir} be the set of *due-dates* ℓ for product i in time interval $[t_{r+1}, T]$:

$$L_{ir} = \{\ell \in [t_{r+1}, T] : \hat{D}_{i\ell}^r > 0\}, \quad i = 1, \ldots, n; \;\; r < k. \tag{11}$$

C^r consists of the following constraints:

$$\sum_{t=t_r}^{T_r} x_{it} + \sum_{\substack{t \in L_{ir} \\ t \leqslant \ell}} u_{it} \geqslant \sum_{t=t_{r+1}}^{\ell} \hat{D}_{it}^r, \quad i = 1, \ldots, n; \;\; \ell \in L_{ir}; \;\; r < k, \tag{12}$$

$$\sum_{i=1}^{N} \sum_{\substack{t \in L_{ir} \\ t \leqslant \ell}} u_{i\ell} \leqslant m(\ell - t_{r+1} + 1), \quad i = 1, \ldots, n; \;\; \ell \in L_{ir}; \;\; r < k, \tag{13}$$

where $u_{i\ell}$ are additional non-negative integer variables defined for $i = 1, \ldots, n$, $\ell \in L_{ir}$ and $r < k$. For each product i, $u_{i\ell}$ variables represents a feasible value of the unsatisfied demand due within ℓ. Constraints (12) state the relationship between the x and u variables. Constraints (13) state that an unsatisfied demand due within ℓ cannot exceed production capacity in time interval $[t_{r+1}, T]$.

3.2. RH2 heuristic

The second RH heuristic is made up of two steps.

Step 1. Solve a MIP made up of the *linear relaxation* of (1)–(6) and of the additional inequalities:

$$\sum_{\ell=t_r}^{T_r} x_{i\ell} = q_{ir}, \quad i = 1, \ldots, n; \;\; r < k, \tag{14}$$

$$q_{ir} \geqslant 0, \;\text{integer}, \quad i = 1, \ldots, n; \;\; r < k. \tag{15}$$

In this auxiliary problem, only q_{ir} variables are constrained to be integer. Let $\hat{q}_{ir}$ be their optimal values.

Step 2. Run an RH heuristic in which C^r consists of constraints

$$\sum_{\ell=t_r}^{T_r} x_{i\ell} \geqslant \hat{q}_{ir}, \quad i = 1, \ldots, n; \;\; r < k. \tag{16}$$

Each $\hat{q}_{ir}$ value represents a dummy product demand to be due within T_r. Constraints (16) are the corresponding due-date constraints. In Proposition 3 it will be proved that the due-date constraints (16) guarantee the feasibility of the heuristic approach.

It is worth noting that both RH1 and RH2 procedures result in a feasible schedule for the overall problem as stated by the following two propositions.

Proposition 2. *If an LSPIPM instance is feasible then subproblems* $LSPIPM^r_{RH}$ $(r = 1, \ldots, k)$ *of RH1 are feasible too.*

Proof. First, we observe that an instance of LSPIPM is feasible if and only if:

$$\sum_{i=1}^{n} \sum_{\ell=1}^{t} D_{i\ell} \leqslant mt, \quad t \in [1, T]. \tag{17}$$

Then we prove by induction that if an LSPIPM instance is feasible then all subproblems $LSPIPM^r_{RH}$ $(r = 1, \ldots, k)$ of RH1 are feasible too. Let $\overline{x}_{it}$ $(i = 1, \ldots, n; t = 1, \ldots, T)$ and $\overline{y}_{ijt}$ $(i, j = 1, \ldots, n; t = 1, \ldots, T)$ be a feasible solution to the LSPIPM and let $[t_r, T_r]$ $(r = 1, \ldots, k)$ be a partition of the planning horizon $[1, T]$. Since $\overline{x}_{it}$ and $\overline{y}_{ijt}$ values constitute a feasible solution to the original problem, then the following relations hold:

$$\sum_{t=1}^{T_1} \overline{x}_{it} + \sum_{t=t_2}^{\ell} \overline{x}_{it} \geqslant \sum_{t=t_2}^{\ell} \hat{D}^1_{it}, \quad i = 1, \ldots, n; \quad \ell = t_2, \ldots, T, \tag{18}$$

$$\sum_{i=1}^{n} \overline{x}_{it} = m, \quad t = 1, \ldots, T_1. \tag{19}$$

Conditions (18) and (19) imply

$$\sum_{t=t_r}^{T_r} \overline{x}_{it} + \sum_{\substack{t \in L_{i1} \\ t \leqslant \ell}} \overline{u}_{it} \geqslant \sum_{t=t_2}^{\ell} \hat{D}^1_{it}, \quad i = 1, \ldots, n; \quad \ell \in L_{i1}, \tag{20}$$

$$\sum_{i=1}^{N} \sum_{\substack{t \in L_{i1} \\ t \leqslant \ell}} \overline{u}_{i\ell} = m(\ell - t_2 + 1), \quad i = 1, \ldots, n; \quad \ell \in L_{i1}, \tag{21}$$

where

$$\overline{u}_{i\ell} = \sum_{t=t_r+1}^{\ell} \overline{x}_{it}, \quad i = 1, \ldots, n; \quad \ell \in L_{i1}. \tag{22}$$

As a result, $\overline{x}_{it}, \overline{y}_{ijt}, \overline{u}_{i\ell}$ $(i, j = 1, \ldots, n; \ell \in L_{i1}, t = 1, \ldots, T_1)$ determine an $LSPIPM^1_{RH}$ feasible solution. For any $r > 1$, suppose subproblems $LSPIPM^1_{RH}, \ldots, LSPIPM^{r-1}_{RH}$ are feasible. Let $LSPIPM^{r^*}$ be the LSPIPM defined over the planning horizon $[t_r, T]$ in which demands and initial product–machine assignments have been set on the basis of the solution values of $LSPIPM^1_{RH}, \ldots, LSPIPM^{r-1}_{RH}$. We prove that if $LSPIPM^{r^*}$ is feasible, then the feasibility of $LSPIPM^r_{RH}$ can be determined as illustrated for $LSPIPM^1_{RH}$. The following relations are verified:

$$\sum_{t=t_r}^{\ell} x_{it} \geqslant \sum_{t=t_r}^{\ell} \overline{D}_{it}, \quad i = 1, \ldots, n; \quad \ell = t_r, \ldots, T, \tag{23}$$

$$\overline{D}_{it} = \sum_{\ell=1}^{t} D_{i\ell} - \sum_{\ell=1}^{t} \hat{x}_{i\ell}, \quad i = 1, \ldots, n; \quad t = t_r, \ldots, T. \tag{24}$$

In addition, since

$$\sum_{i=1}^{N} \hat{x}_{iT_{r-1}} = m, \tag{25}$$

the capacity available in $[t_r, \ell] \subseteq [t_r, T]$ is equal to $m(\ell - T_{r-1})$. Then,

$$\sum_{i=1}^{n}\sum_{t=t_r}^{\ell} \overline{D}_{it} = \sum_{i=1}^{n}\left(\sum_{t=1}^{\ell} D_{it} - \sum_{t=1}^{T_{r-1}} \hat{x}_{it}\right) \leqslant \sum_{i=1}^{n}\sum_{t=1}^{\ell} D_{it} - mT_{r-1}. \tag{26}$$

Finally, since $LSPIPM \equiv LSPIPM^{1^*}$ is feasible then

$$\sum_{i=1}^{n}\sum_{t=t_r}^{\ell} \overline{D}_{it} \leqslant \sum_{i=1}^{n}\sum_{t=1}^{\ell} D_{it} - mT_{r-1} \leqslant m(l - T_{r-1}), \tag{27}$$

which shows that $LSPIPM^{r^*}$ is feasible. □

Proposition 3. *If an LSPIPM instance is feasible then subproblems* $LSPIPM^r_{RH}$ $(r = 1, \ldots, k)$ *of RH2 are feasible too.*

Proof. Let $\tilde{x}_{it}$ $(i = 1, \ldots, n; t = 1, \ldots, T)$ be the optimal solution values of x_{it} variables in the initialization step. The following relations hold:

$$\sum_{t=t_r}^{T_r} \tilde{x}_{it} = \hat{q}_{ir}, \quad i = 1, \ldots, n; \quad r = 1, \ldots, k, \tag{28}$$

$$\sum_{\ell=1}^{T_r} \tilde{x}_{i\ell} \geqslant \sum_{\ell=1}^{T_r} D_{i\ell}, \quad i = 1, \ldots, n; \quad r = 1, \ldots, k. \tag{29}$$

Constraints (29) dominate constraint (4) for every i and for $t = T_r$. Therefore, $LSPIPM^r_{RH}$ can be reformulated as an LSPIPM over $[t_r, T_r]$ in which the right-hand side of (4) is set equal to $\hat{q}_{ir}$ for $t = T_r$. If (8) is satisfied, then

$$\sum_{i=1}^{n}\sum_{t=1}^{\ell} \hat{D}^r_{it} \leqslant m(\ell - t_r + 1), \quad \ell \in [t_r, T_r]; \quad r = 1, \ldots, k, \tag{30}$$

$$\sum_{i=1}^{n} \hat{q}_{ir} \leqslant m(T_r - t_r + 1), \quad r = 1, \ldots, k. \tag{31}$$

Inequality (31) is verified because of (28). As far as (30) is concerned, it should be noted that $\sum_{i=1}^{N} \hat{x}_{it} = m$ $(t = 1, \ldots, T_{r-1})$. Hence,

$$\sum_{i=1}^{n}\sum_{t=1}^{\ell} D_{it} \leqslant ml, \quad \ell = 1, \ldots, T_1, \tag{32}$$

$$\sum_{i=1}^{n}\sum_{t=1}^{\ell} D_{it} - \sum_{i=1}^{n}\sum_{t=1}^{T_{r-1}} \hat{x}_{it} \leqslant m(l - t_r + 1), \quad \ell = t_r, \ldots, T_r; \quad r > 2. \tag{33}$$

As the left-hand sides of (32) and (33) are equal to $\hat{D}^r_{it}$ the thesis is proved. □

Proposition 4. *If subproblems are formulated according to either RH1 or RH2 it results that: if all subproblems* $LSPIPM^r_{RH}$ $(r = 1, \ldots, k)$ *are feasible then the original LSPIPM is feasible too.*

Proof. Let the values $\hat{x}^r_{it}$ and $\hat{y}^r_{ijt}$ denote the solution values of x and y variables for subproblem $LSPIPM^r_{RH}$, $r = 1, \ldots, k$. By hypothesis, the values $\hat{x}^r_{it}$ and $\hat{y}^r_{ijt}$, with $i = 1, \ldots, N$, $j = 1, \ldots, N$ and $t = t_r, \ldots, T_r$ for $r = 1, \ldots, k$, satisfy constraints (2)–(3). Due to their network structure, the following equality is satisfied for each period:

$$\sum_{i=1}^{N} \hat{x}^r_{it} = M, \quad t = t_r, \ldots, T_r. \tag{34}$$

Moreover, $\hat{x}^r_{it}$ satisfy the due-date constraints (4) as follows:

$$\sum_{t=t_r}^{\ell} \hat{x}^r_{it} \geqslant \sum_{t=t_r}^{\ell} \hat{D}^r_{it}, \quad i = 1, \ldots, N; \;\; \ell = t_r, \ldots, T_r; \;\; r = 1, \ldots, k. \tag{35}$$

Since the parameter $\hat{D}^r_{it}$ is defined as reported in (8) it results that

$$\sum_{i=1}^{N} \sum_{t=1}^{\ell} D_{it} \leqslant \sum_{i=1}^{N} \sum_{t=1}^{\ell} \hat{x}^r_{it} \leqslant M^* t, \quad t = 1, \ldots, T. \tag{36}$$

As stated preliminary in proof of Proposition 2, the thesis is proved. □

Given the above Proposition it results that a solution to the LSPIPM can be obtained as follows:

$$x_{it} = \hat{x}^r_{it} \quad \forall i = 1, \ldots, N; \;\; t \in [t_r, T_r]; \;\; r = 1, \ldots, k, \tag{37}$$

$$y_{ijt} = \hat{y}^r_{ijt} \quad \forall i; \;\; j = 1, \ldots, N; \;\; t \in [t_r, T_r]; \;\; r = 1, \ldots, k. \tag{38}$$

4. The FR heuristic

The FR methodology [19] decomposes a large-scale MIP problem into a number of smaller MIP subproblems. The limited size of these subproblems allows using exact methods for their solution, which would be impossible for the entire problem (see, e.g., [20,21]). As in the RH approach, the number and the size of the subproblems define the computational burden and the solution quality of the heuristic procedures.

When the FR approach is applied to formulation (1)–(6), integer variables x_{it} ($i = 1, \ldots, n$; $t = 1, \ldots, T$) are partitioned into k sets ($\{x_{it} : (i, t) \in X_h\}, h = 1, \ldots, k$) and the rth subproblem $LSPIPM^r_{FR}$ amounts solving (1)–(6) with the following constraints:

$$x_{it} = \hat{x}_{it}, \quad (i, t) \in \bigcup_{h=1,\ldots,r-1} X_h, \tag{39}$$

$$x_{it} \geqslant 0 \text{ and integer}, \quad (i, t) \in X_r, \tag{40}$$

$$x_{it} \geqslant 0, \quad (i, t) \in \bigcup_{h=r+1,\ldots,k} X_h, \tag{41}$$

where $\hat{x}_{it}$ are the solution values of the integer variables in subproblems $LSPIPM^1_{FR}, \ldots, LSPIPM^{r-1}_{FR}$. Since only a reduced subset of variables are kept integer at each stage r, $LSPIPM^r_{FR}$ is expected to be solved relatively easily.

The FR methodology may provide a lower bound (associated to subproblem 1 solution) as well as an upper bound. It is worth noting [22] that a generic FR algorithm may fail to identify a feasible solution even if the problem is feasible. However, in the following we demonstrate that if the LSPIPM problem is feasible, then the four variable partition policies we propose always result in feasible subproblems.

Time-based variable partitioning. This strategy partitions the planning horizon into k time intervals $[t_h, T_h]$ ($h = 1, \ldots, k$) such that $t_1 = 1$, $t_h = T_{h-1} + 1$ ($h = 2, \ldots, k$) and $T_k = T$ subproblems are solved, one for each time period:

$$X_h = \{(i, t) : i = 1, \ldots, n; t \in [t_h, T_h]\}, \quad h = 1, \ldots, k. \tag{42}$$

Thus, the resulting FR heuristic (FR1) determines first the values of those integer variables associated with decisions that occur early in time.

Product-based variable partitioning. The second strategy which is based upon a partitioning of products, partitions the planning horizon into k time intervals $[i_h, n_h]$ $(h = 1, \ldots, k)$ such that $i_1 = 1, i_r = n_{r-1} + 1$ $(h = 2, \ldots, k)$ and $n_k = n$ Then,

$$X_h = \{(i, t) : i \in [i_h, n_h]; t = 1, \ldots, T\}, \quad h = 1, \ldots, n. \tag{43}$$

Thus, the resulting FR heuristic (FR2) determines first the values of those integer variables associated with high overall demands.

Time–product hybrid variable partitioning. The third strategy (FR3) partitions the planning horizon into a number of intervals and applies a product partitioning in each interval.

Time–product hybrid variable partitioning. In the fourth strategy, products are partitioned first and then a time-partitioning procedure is used.

We now prove that the above four FR heuristics always provide a feasible schedule if LSPIPM is feasible.

Proposition 5. *If any of the above FR procedures is used, all subproblems are feasible whenever LSPIPM is feasible.*

Proof. Let $\hat{y}_{ijt}$ $(i, j = 1, \ldots, n; t = 1, \ldots, T)$ be the optimal solution values of y_{ijt} variables in LSPIPM. We prove the proposition by induction. $LSPIPM^1_{FR}$ is a relaxation of the original LSPIPM. Since LSPIPM is feasible by hypothesis, $LSPIPM^1_{FR}$ is feasible too. For $r > 1$, the demonstration has to be specialized for each of the proposed FR heuristics.

Procedure FR1. Let $LSPIPM^r_{FR}$ be the rth subproblem solved by a FR procedure and let $LSPIPM'^r_{FR}$ be a subproblem (1)–(6) defined over a planning horizon $[t_r, T]$. In $LSPIPM'^r_{FR}$, product demands $\bar{D}_{it}$ $(i = 1, \ldots, n; t = t_r, \ldots, T)$ are determined on the basis of the optimal solution values of $LSPIPM^1_{FR}, \ldots, LSPIPM^{r-1}_{FR}$. Relations (17) imply that $LSPIPM'^r_{FR}$ is feasible. Moreover, $LSPIPM'^r_{FR}$ is equivalent to $LSPIPM^r_{FR}$ if $r = k$, while $LSPIPM^r_{FR}$ is a relaxation of $LSPIPM'^r_{FR}$ if $1 < r < k$. In both cases the feasibility of $LSPIPM'^r_{FR}$ implies the feasibility of $LSPIPM^r_{FR}$.

Procedure FR2. Let $LSPIPM'^r_{FR}$ be a subproblem (1)–(6) with a restricted set of products $\{i_r, \ldots, n\}$ subject to the following constraints:

$$\sum_{i=i_r}^{n} x_{it} = m_t, \quad t = 1, \ldots, T, \tag{44}$$

where

$$m_t = m - \sum_{i=1}^{n_r - 1} \hat{x}_{it}, \quad t = 1, \ldots, T, \tag{45}$$

and $\hat{x}_{it}$ are the optimal solution values of variables x_{it} in subproblems $LSPIPM^1_{FR}, \ldots, LSPIPM^{r-1}_{FR}$. As demands in $LSPIPM'^r_{FR}$ are the same as in LSPIPM, the following inequalities hold:

$$\sum_{t=1}^{\ell} \hat{x}_{it} \geqslant \sum_{t=1}^{\ell} D_{it}, \quad \ell = 1, \ldots, T; \quad i = 1, \ldots, n_{r-1}. \tag{46}$$

Conditions (17) (which are verified since LSPIPM is feasible by hypothesis), can be rewritten as follows:

$$\sum_{t=1}^{\ell} \sum_{i=i_r}^{n} D_{it} \leqslant m\ell - \sum_{t=1}^{\ell} \sum_{i=1}^{n_{r-1}} D_{it}, \quad \ell = 1, \ldots, T. \tag{47}$$

Then, on the basis of (44) and (46), the total product demand does not exceed the available production capacity

$$\sum_{t=1}^{\ell} \sum_{i=i_r}^{N} D_{it} \leqslant \sum_{t=1}^{\ell} m_t, \quad \ell = 1, \ldots, T. \tag{48}$$

Hence, $LSPIPM'^{r}_{FR}$ is feasible. As in the FR1 case, if $r = k$, subproblem $LSPIPM^{r}$ is equivalent to $LSPIPM'^{r}_{FR}$ while it is a relaxation of $LSPIPM'^{r}_{FR}$ if $1 < r < k$. In both cases the feasibility of $LSPIPM'^{r}_{FR}$ implies the feasibility of $LSPIPM^{r}_{FR}$.

Procedures FR3 and FR4. In these cases, the proof is a trivial extension of those illustrated for FR1 and FR2. □

Following the idea underlying the proof of Proposition 4 it can be demonstrated what follows.

Proposition 6. *If subproblems are formulated according to a FR approach it results that: if all subproblems* $LSPIPM^{r}_{FR}$ $(r = 1, \ldots, k)$ *are feasible then the original LSPIPM is feasible too.*

The determination of a solution to the original problem LSPIPM can be recovered as already illustrated in (37) and (38).

5. Computational results

Both RH and FR procedures have been implemented in C + + and linked to CPLEX 7.0 in order to solve MIP subproblems. The resulting codes have been run on a PC with a 800 MHz Pentium processor with 256 MB of RAM. In particular, for the set-up cost parameter three classes of problems, i.e., S1, S2 and S3, have been generated and values have been taken from uniform distributions defined in the intervals [1,10], [1,100] and [1,1000], respectively. Moreover for the due-date parameter, two classes of problems have been considered, each of them characterized by either identical due-dates (D1) or due-dates uniformly distributed over the planning horizon (D2). For each combination of these parameters (i.e., set-up and due-date), a set of test problems has been generated. All tests were characterized by 100 machines (i.e., $m = 100$): a typical value for textile and fiberglass industries. In all the tests, the solution provided by each heuristic has been compared with a lower bound provided by a truncated branch-and-bound algorithm (which has been allowed to run for 3600 s). This parameter has been used to determine the solution quality. The column headings of Tables 1–4 are as follows:

- HEUOBJ: percentage deviation of the solution value provided by heuristic HEU from the truncated branch-and-bound lower bound;
- HEUSEC: computing time in seconds for heuristic HEU.

Tables 1 and 2 report computational results for the FR procedures FR1, FR2, FR3 and FR4 on 60 instances with $n = 24$ and $T = 30$. Both the solution quality and the efficiency of the FR procedures depend on the number and size of the subproblems. In order to determine the best trade off between efficiency and solution quality, a preliminary computational campaign was carried out. For each FR procedure, we have determined the number of subproblems as the maximum k value providing an optimality gap not exceeding 10%. The k value was equal to 3 and 6 for FR1, FR2 and FR3, FR4, respectively. As reported in Table 2, the most efficient FR procedures were FR3 and FR4, characterized by subproblems with approximately 100 integer variables. FR1 and FR2 were the less efficient procedures since the number of subproblems was halved, but the size of each subproblem was doubled. The boxplots in Fig. 1 report the solution quality produced by the four FR procedures. On the basis of these boxplots, we may state that the FR2 procedure outperforms the other ones. To prove with statistical evidence that the highest quality solution is provided by a pure product-partitioning policy (i.e., by the FR2 heuristic), the *analysis of the variance* and the *Tukey's pairwise comparison* were applied to the values reported in Table 1 (with family error rate $= 0.05$). For each problem class, except D2–S3, the two procedures demonstrated the superiority of the FR2 procedure. For this reason, the RH algorithms were then compared only with the FR2 procedure (see Tables 3 and 4). The RH approaches were capable of solving instances with up to 30 time periods and 12 products (i.e., $n = 12$, $T = 30$). As mentioned for the FR approach, the k value ($k = 3$) was determined by preliminary tests. A boxplot representation of the results have been shown in Figs. 2 and 3. Moreover, the median, the minimum, the maximum, the first and the third quartiles of the deviations from the lower bound have been reported in Table 5. On the basis of these statistics, it is possible to conclude that FR2 outperforms RH procedures. In particular, the median value is significantly lower for FR2 than for RH procedures, and the third quartile for FR2 is significantly lower than the first quartile for the best RH procedure.

Table 1
FR1, FR2, FR3 and FR4 solution quality ($n = 24$, $T = 30$)

Instance	D1–S1				D2–S1				D1–S2			
	FR1OBJ	FR2OBJ	FR3OBJ	FR4OBJ	FR1OBJ	FR2OBJ	FR3OBJ	FR4OBJ	FR1OBJ	FR2OBJ	FR3OBJ	FR4OBJ
1	1.75	0.83	2.5	2.25	1.30	0.62	1.64	1.37	1.69	0.5	0.9	0.78
2	1.66	0.65	1.29	1.85	1.40	1.03	0.58	0.34	0.46	0.7	2	1.35
3	1.66	0.68	2.15	2.35	1.20	0.76	1.44	1.81	1.06	0.82	1.56	1.11
4	1.48	1.69	2.86	2.01	1.70	0.51	1.52	1.52	1.75	1.7	0.94	1.11
5	1.95	1.69	2.86	2.73	1.90	0.42	1.54	1.46	1.29	1.4	0.94	1.1
6	2.13	0.88	2.75	2.38	1.5	0.8	1.3	1.36	1.53	1.65	0.9	1.07
7	3.83	2.24	3.83	3.67	1.32	0.6	0.88	0.79	1.03	1.27	1.36	0.95
8	1.41	1.03	1.88	1.97	1.54	0.59	1	1	1.03	1.2	0.99	1.19
9	2.07	1.27	3.11	2.65	1.60	1.47	1.32	1.57	1.35	1.45	1.42	0.92
10	2.64	0.97	3.33	3.06	1.82	0.64	1.33	1.5	1.06	1.3	1.41	0.93
Instance	D2–S2				D1–S3				D2–S3			
	FR1OBJ	FR2OBJ	FR3OBJ	FR4OBJ	FR1OBJ	FR2OBJ	FR3OBJ	FR4OBJ	FR1OBJ	FR2OBJ	FR3OBJ	FR4OBJ
1	1.46	0.91	1.1	1.21	1.14	1.06	0.9	0.93	1.64	0.81	1.42	1.34
2	0.54	0.58	1.23	2.56	1.59	0.61	2	2.04	0.61	0.68	0.88	1.07
3	1.3	0.65	1.24	1.08	1.26	0.94	1.56	1.55	1.02	0.57	1.05	1.71
4	0.99	1.37	1.85	1.66	1.42	0.64	0.94	0.9	0.79	0.95	1.42	0.98
5	1.46	1.23	0.54	0.68	0.67	0.28	0.94	1.02	1.51	0.61	1.12	0.9
6	1.1	0.71	1.13	1.39	0.74	0.62	0.9	0.88	0.65	0.75	1	1.95
7	0.88	0.42	1.23	0.94	1.08	0.63	1.36	1.48	3.38	1.7	1.59	1.72
8	1.1	0.48	1.21	1.6	1.24	0.6	0.99	1.1	0.89	0.64	1.1	1.45
9	1.24	0.74	1.26	3.13	1.31	1.15	1.42	1.52	0.89	0.72	1.1	1.44
10	1.16	0.83	1.69	1.38	1.32	1.02	1.41	1.49	1.39	0.81	1.02	1.16

Table 2
FR1, FR2, FR3 and FR4 computing times ($n = 24$, $T = 30$)

Instance	D1–S1				D2–S1				D1–S2			
	FR1SEC	FR2SEC	FR3SEC	FR4SEC	FR1SEC	FR2SEC	FR3SEC	FR4SEC	FR1SEC	FR2SEC	FR3SEC	FR4SEC
1	86	164	129	117	178	278	56	68	184	388	50	140
2	76	185	55	101	116	344	46	50	282	806	99	73
3	88	155	91	152	104	271	87	80	140	391	87	93
4	60	259	74	112	191	450	69	81	695	234	72	91
5	93	425	85	215	209	276	73	83	219	184	82	102
6	95	171	102	112	202	191	77	86	116	160	67	75
7	144	390	88	168	81	187	72	81	120	158	70	74
8	79	270	84	187	326	352	79	113	148	379	75	103
9	73	239	97	113	103	221	67	75	126	202	88	205
10	91	195	84	90	111	306	82	120	169	253	73	85
Instance	D2–S2				D1–S3				D2–S3			
	FR1SEC	FR2SEC	FR3SEC	FR4SEC	FR1SEC	FR2SEC	FR3SEC	FR4SEC	FR1SEC	FR2SEC	FR3SEC	FR4SEC
1	103	183	59	102	200	262	55	90	174	369	97	127
2	102	180	92	121	789	274	68	70	346	208	68	107
3	99	166	156	249	716	198	96	93	565	191	96	133
4	86	293	81	127	337	190	115	110	165	260	115	109
5	166	280	77	90	131	165	85	89	207	204	85	94
6	121	223	124	126	233	783	92	90	110	214	92	121
7	288	196	77	93	151	174	102	100	178	176	102	114
8	95	195	104	140	287	490	104	106	754	245	104	100
9	87	153	113	115	220	224	96	98	127	207	96	120
10	108	247	105	98	135	222	93	94	176	595	93	170

Table 3
FR2, RHP1 and RHP2 solution quality ($n = 12$, $T = 30$)

Instance	D1–S1			D2–S1			D1–S2		
	FR2OBJ	RH1OBJ	RH2OBJ	FR2OBJ	RH1OBJ	RH2OBJ	FR2OBJ	RH1OBJ	RH2OBJ
1	1.79	11.32	91.51	0.3	6.61	154.13	0.06	12.35	124.06
2	1.29	9.89	99.27	0.43	0.87	312.83	2.21	10.63	84.15
3	2.56	16.88	121.88	0.3	4.23	196.62	1.22	10.71	129.25
4	5.88	15.88	124.46	0.47	3.36	365.63	1.99	17.16	146.84
5	3.54	18.37	122.45	0.27	9.85	141.54	1.38	12.93	134.97
6	1.06	9.4	81.2	0.37	7.51	154.46	2.15	11.98	100.91
7	2.97	11.06	111.98	0.5	5.98	247.28	1.68	15.83	124.98
8	2.98	12.24	120.41	0.28	11.59	178.41	2.34	19.72	123.39
9	2.31	14.08	131.92	0.96	4.17	73.57	0.27	9.33	103.26
10	0.76	11.34	127.32	0.33	4.69	178.28	0.66	11.86	152.57

Instance	D2–S2			D1–S3			D2–S3		
	FR2OBJ	RH1OBJ	RH2OBJ	FR2OBJ	RH1OBJ	RH2OBJ	FR2OBJ	RH1OBJ	RH2OBJ
1	0.21	22.52	267.02	2.89	11.91	76.07	0.3	18.3	391.53
2	0.5	19.44	119.39	0.71	13.21	236.84	2.2	7.6	212.97
3	0.43	25.83	238.9	0.63	7.61	80.03	0.04	33.08	74.66
4	0.62	106.26	123.65	0.48	9.38	68.01	0.31	9.07	106.21
5	0.61	17.41	279.63	0.23	5.76	63.19	0.28	40.26	101.07
6	0.52	18.47	201.15	0.96	8.85	72.1	1.85	3.44	216.97
7	0.11	5.49	202.94	0.44	25.04	117.05	0.58	17.95	232.87
8	0.51	6.95	226.44	0.81	17	154.12	0.77	51.13	282.27
9	0.28	5.07	98.07	4.1	20.81	195.28	0.08	93.71	95.67
10	0.47	62.46	357.92	1.87	18.96	147.02	0.22	12.54	152.12

Table 4
FR2, RHP1 and RHP2 computing times ($n = 12$, $T = 30$)

Instance	D1–S1			D2–S1			D1–S2		
	FR2OBJ	RH1OBJ	RH2OBJ	FR2OBJ	RH1OBJ	RH2OBJ	FR2OBJ	RH1OBJ	RH2OBJ
1	8.74	212.5	5.61	3.8	3.29	9.39	11.15	73.49	30
2	35.9	127	5.72	3.9	27.02	15.81	4.24	130	20.43
3	98.32	226.8	9.73	2.27	4.73	29.88	7.5	121.7	30.48
4	22.94	89.97	10.43	2.22	4.83	9.6	2.48	342.8	13.84
5	166.03	1253	15.6	2.18	3.5	10.54	1.67	498.6	10.16
6	20.41	308.2	9.12	1.55	1.92	50.64	19.49	68.1	27.52
7	321.58	1018	16.14	2.28	1.05	20.44	122.64	908.1	21.7
8	106.95	91.89	8.24	1.9	1.6	14.22	12.79	87.49	28.84
9	28.13	186.9	5.11	1.5	2.69	10.27	14.67	56.95	8.18
10	5.77	61.63	6.92	5	21.75	111.66	35.76	1869	13.52

Instance	D2–S2			D1–S3			D2–S3		
	FR2OBJ	RH1OBJ	RH2OBJ	FR2OBJ	RH1OBJ	RH2OBJ	FR2OBJ	RH1OBJ	RH2OBJ
1	24.43	6.54	12.79	9.8	1166	27.8	12.75	56	11.31
2	16.95	6.1	32.62	3.47	100	15.71	5.23	157	27.63
3	13.55	45.92	39.83	2.49	146	37.46	4.06	44	52.73
4	22.87	114.2	9.44	7.31	267	549.03	22.04	6.61	63
5	18.47	43.01	13.09	5.67	484	18.89	10.75	7.31	92.99
6	16.44	47.79	171.8	51.97	3768	28.84	19.47	91.51	31.74
7	6.56	31.46	18.73	6.36	4194	14.34	15.85	57	17.53
8	13.66	150	9.34	18.76	290	67.18	3.46	551	58.93
9	19.37	13.8	19.61	74.23	533	80.25	3.2	2.8	12.91
10	20.85	61	16.37	21.31	2324	19.56	3.9	3.41	19.62

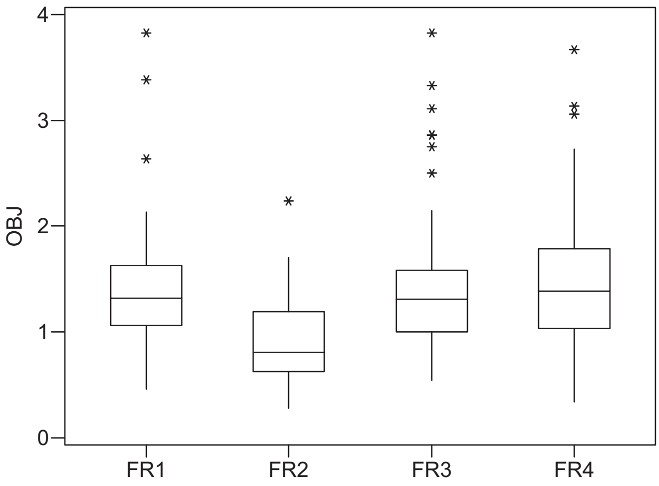

Fig. 1. Solution boxplot for the FR procedures.

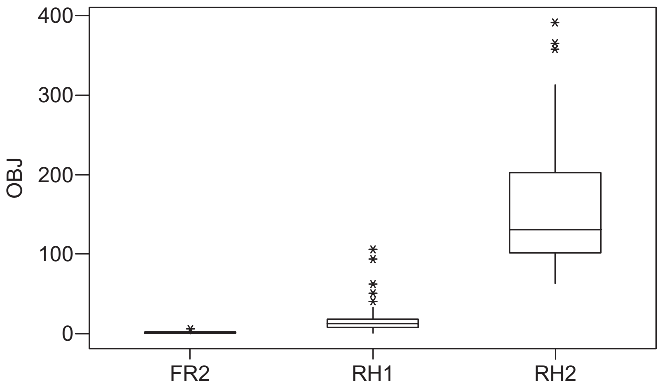

Fig. 2. Solution boxplot for the FR2, RH1 and RH2 procedures.

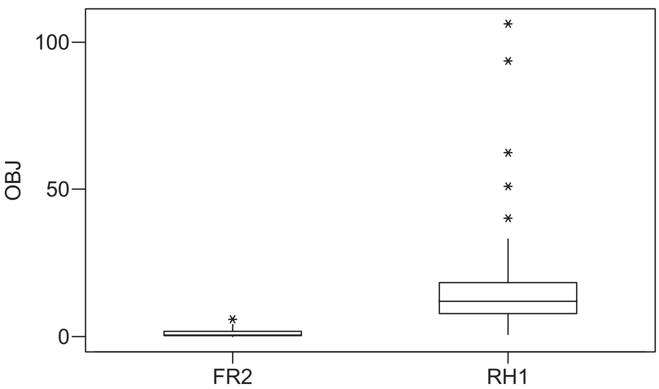

Fig. 3. Solution boxplot for the FR2 and RH1 procedures.

6. Conclusions

In this paper we have developed new RH and FR heuristics for the identical parallel machine lot-sizing and scheduling problem with sequence-dependent set-up costs. Unlike previous papers, our procedures are based on a compact

Table 5
FR2, RHP1 and RHP2 descriptive statistics

Procedure	N	Median	Minimum	Maximum	$Q1$	$Q3$
FR2	60	0.61	0.040	5.88	0.30	1.83
RH1	60	11.95	0.87	106.26	7.92	18.35
RH2	60	130.60	63.2	391.5	101.6	202.5

formulation relying on the hypotheses of identical machines. This feature makes our approach suitable for large-scale applications (with approximately 100 machines) arising in the textile and fiberglass industries. Comparisons with lower bounds provided by a truncated branch-and-bound show that FR procedures outperform RH heuristics. For all FR algorithms, the gap between the heuristic solution and the lower bound never exceeds 6%. In particular, the highest quality solution is almost always obtained by the FR2 procedure in a reasonable amount of time. From qualitative analyses of the solutions, the superiority of the product-partitioning policy FR2 can be explained as follows. For each product due-date, the FR2 solution makes use of as few machines as possible for as many time periods as possible. On the other hand, RH heuristics delay as much as possible product demand satisfaction, worsening subproblem total set-up cost as due-dates become more and more urgent.

Acknowledgments

This research was partially supported by MIUR, Italy, and by the Center of Excellence on High Performance Computing, University of Calabria, Italy. This support is gratefully acknowledged.

References

[1] Wagner HM, Whitin TM. Dynamic version of the economic lot size model. Management Science 1958;5:89–96.
[2] Drexl A, Kimms A. Lot sizing and scheduling, survey and extensions. European Journal of Operational Research 1997;99:221–35.
[3] Staggemeier AT, Clark AR. A survey of lot-sizing and scheduling models. In: Proceedings of the 23rd annual symposium of the Brazilian operational research society. Brazil: Campos do Jordão; 2001.
[4] Pochet Y. Mathematical programming models and formulations for deterministic production planning problems. In: Jünger M, Naddef D, editors. Computational combinatorial optimization. Lecture notes in computer science. Berlin: Springer; 2001. p. 57–111.
[5] Dearingm PM, Henderson RA. Assigning looms in a textile weaving operation with changeover limitation. Production and Inventory Management 1984;25:23–31.
[6] Clark AR. Batch sequencing and sizing with regular varying demand. Production Planning and Control 1998;9(3):260–6.
[7] Clark AR, Clark SJ. Rolling-horizon lot-sizing when set-up times are sequence dependent. International Journal of Production Research 2000;38(10):2287–308.
[8] Meyr H. Simultaneous lot-sizing and scheduling on parallel machines. European Journal of Operational Research 2002;139:277–92.
[9] Meyr H. Simultaneous lot-sizing and scheduling by combining local search with dual optimization. European Journal of Operational Research 2000;120:311–26.
[10] Fleischmann B, Meyr H. The general lot sizing and scheduling problem. OR Spektrum 1997;19(1):11–21.
[11] Staggemeier AT, Clark AR, Aickelin U, Smith J. A hybrid genetic algorithm to solve a lot-sizing and scheduling problem. Report MS-2002-4. Intelligent Computer Systems Centre Group, University of the West of England, Bristol, Great Britain; 2002.
[12] Serafini P. Scheduling jobs on several machines with the job splitting property. Operations Research 1996;44(4):617–28.
[13] Xing W, Zhang J. Parallel machine scheduling with splitting jobs. Discrete Applied Mathematics 2000;103:259–69.
[14] Nait Tahar D, Yalaoui F, Chu C, Amodeo L. A linear programming approach for identical parallel machine scheduling with job splitting and sequence-dependent set-up times. International Journal of Production Economics 2006;99:63–73.
[15] Sumischrast RT, Frendewey JO. Scheduling parallel processors with set-up costs and resource limitations. Decisions Science 1988;19:138–46.
[16] Allahverdi A, Gupta JND, Aldowaisan T. A review of scheduling research involving setup consideration. Omega 1999;27:219–39.
[17] Ovacik M, Uzsoy R. Rolling horizon algorithms for a single machine dynamic scheduling problem with sequence-dependent set-up times. International Journal of Production Research 1995;33(11):3173–92.
[18] Clark AR. Rolling horizon heuristics for production planning and set-up scheduling with backlogs and error-prone demand forecast. Production Planning & Control 2005;16(1):81–97.
[19] Wolsey LA. Integer programming. New York: Wiley; 1998.
[20] Dillenberger C, Escudero LF, Wollensak A, Zhang W. On practical resource allocation for production planning and scheduling with period overlapping setups. European Journal of Operational Research 1994;75:275–86.
[21] Clark AR, MorabitoNeto R, Toso EAV. Multi-period production setup-sequencing and lot-sizing through ATSP subtour elimination and patching. Proceedings of the 25th workshop of the UK planning and scheduling special interest group (PlanSIG 2006). December 14–15, 2006, University of Nottingham; p. 80–87. ISSN 1368-5708.
[22] Escudero LF, Salmeron J. On a fix-and-relax framework for large-scale resource-constrained project scheduling. Annals of Operations Research, 2002, forthcoming.