A Matheuristics for the Single-period Lot Scheduling with Component Availability Constraints in a Partially Closed Manufacturing/Remanufacturing System

Davide Giglio and Massimo Paolucci
DIBRIS, University of Genova, Via Opera Pia, 13, 16145, Genova, Italy

Keywords: Remanufacturing, Production Scheduling, Mixed-integer Programming Modelling, Matheuristics.

Abstract: An integrated manufacturing/remanufacturing system is considered in this paper with the aim of scheduling the operations of the manufacturing plant. The system is partially closed in the sense that the raw materials, necessary for assembling the final products, can be obtained both from an internal remanufacturing plant (which disassembles returned products) and from external suppliers. The manufacturing system is modelled as a flexible flow shop whose stages represent the different assembly phases leading to the final products. In this paper, an original event-based mixed integer programming (MIP) formulation is presented, whose objective consists of minimizing, as primary objective, the weighted number of tardy jobs and, as secondary ones, the fixed and variable purchase costs of raw materials possibly acquired from external suppliers. Due to the complexity of the problem, the MIP formulation can be used to solve only small instances. For this reason, a matheuristics is proposed, which consists of three interoperating mathematical programming models: the first model assigns the jobs to the machines; the second model sequences the jobs on the machines; the third model defines the external supplies, taking into account the component availability constraints. A preliminary computational analysis shows the effectiveness of the proposed algorithm.

1 INTRODUCTION

In recent years the research community has increasingly dedicated efforts in dealing with environmental issues. Among these, the ones relevant to supply chains, specifically reverse logistics, represent the overall framework for the work presented in this paper. Classical supply chain research focuses on the management and optimization of the flow of materials and products within a network connecting suppliers, producers, distributors and customers so that the product demand can be satisfied in the right times, quantities and locations by optimizing the use of the available resources and the operational costs. To account for environmental aspects, this structure has been extended by incorporating reverse flows of finished products at their end-of-life that are returned by customers (Fleischmann et al., 2001; Shah, 2005). Therefore, reverse logistics includes all the logistic activities allowing to transform the used products, discarded and then returned by customers, into new products to be delivered in the market.

Several classes of reverse logistics are distinguished according to the way used to recover the returned products (Thierry et al., 1995), i.e., reuse, recycling, and remanufacturing. In reuse, the returned product (e.g., a pallet) is directly used after some cleaning or reprocessing; recycling does not maintain the product but aims at recovering the materials composing it (e.g., plastic); remanufacturing denotes the industrial process through which the returned products are restored to like-new condition: products are first disassembled, then usable parts are cleaned, refurbished and stored into parts inventory; finally, the new finished products are reassembled using the restored parts and possibly new parts (Lund, 1998). In this way, the need for new materials is diminished, so increasing the environmental sustainability of the production processes and in general improving the survivability of companies with respect to the new environmental regulations and the environmental awareness of the market.

In this paper the problem of defining the production schedule in an integrated manufacturing/remanufacturing system is considered. In particular, the focus is on optimizing the scheduling of the orders (jobs) in the manufacturing facility taking into account the constraints due to the use of shared com-

Giglio, D. and Paolucci, M.
A Matheuristics for the Single-period Lot Scheduling with Component Availability Constraints in a Partially Closed Manufacturing/Remanufacturing System.
DOI: 10.5220/0006005501100120
In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2016) - Volume 1*, pages 110-120
ISBN: 978-989-758-198-4

ponents that are produced by the remanufacturing facility and made available through a component storage. The main scheduling objective consists in minimizing the number of orders completed after the due date, weighted according to the priority of the orders. In addition, as secondary objective, the possible acquisitions of lots of new components from external suppliers must be optimized. Indeed, the replenishment plan for the shared components from the remanufacturing facility is assumed given and additional new components that may be needed should be ordered to the external suppliers. Such further supplies should be needed as late as possible to increase the possibility of receiving the new components on time, and they should consist of lots as large as possible to reduce the fixed costs of supplies.

In the literature the presence of component availability constraints is considered in both resource-constrained project scheduling (RCPS) and machine scheduling, even if few papers can be found in the latter case. Among them, in (Grigoriev et al., 2005) the complexity of single machine scheduling with raw material constraints with the objective of minimizing the number of tardy jobs or the makespan is analyzed, stating that some variants of this problem are strongly NP-hard even with unit processing times, although polynomially-solvable cases exist. Several types of mathematical programming models are introduced in the literature for scheduling problems with component availability constraints. The case of assembly scheduling is considered in (Kolisch and Hess, 2000) and (Kolisch, 2000) where RCPS-based mixed-integer programming (MIP) models, which use time-indexed variables, are proposed. In general the MIP models provided in the literature differ for the kind of time representations, as analysed in (Pinto and Grossmann, 1995), (Li et al., 2010), and (Mouret et al., 2011). In particular, in (Li et al., 2010) three main classes of models are distinguished, i.e., slot-based, event-based and sequence-based. Among them, event-based MIP models have been recently defined to solve RCPS problems (Zapata et al., 2008; Koné et al., 2011; Artigues et al., 2013).

This paper is organized as follows. After having formally defined the considered scheduling problem, in Section 2 an original event-based MIP model is proposed for its solution. Due to the problem complexity, this formulation can solve only small instances within acceptable computation times. Therefore, in Section 3 a matheuristics, which exploits three interoperating mathematical programming models, based on both event-based and sequence-based MIP formulations, is proposed: the first sequence-based model assigns the jobs to the machines; the second sequence-based model identifies the overall priority order among the jobs; the third event-based model defines the external supplies, taking into account the component availability constraints. In Section 5 a preliminary computational analysis showing the effectiveness of the proposed algorithm is reported and, finally, conclusions are drawn in Section 6.

2 THE SCHEDULING PROBLEM

An integrated manufacturing/remanufacturing ("man/reman") system is considered, in which final products are produced in a manufacturing plant by assembling raw components supplied both by an internal remanufacturing facility (which disassembles returned products) and by external suppliers. A representation of such class of systems is given in Figure 1. The manufacturing system is modelled as a flexible flow shop whose stages represent the different assembly phases which lead to the final products. Let M^{a}_{ks} be the k-th machine at stage s, with $s \in \{1,\ldots,S\}$ and $k \in \{1,\ldots,N^{\mathrm{a}}_s\}$, being S the number of stages and N^{a}_s the number of parallel machines which are present at stage s. The remanufacturing system consists of a disassembling subsystem which extracts the components to be recovered from the returned products, and of a refurbishing subsystem which processes the components bringing them back to an as-good-as-new state. The disassembling subsystem is modelled as a flow shop of N^{d} machines, whereas the refurbishing subsystem consists of a set of N^{r} parallel machines which perform operations. Let M^{d}_k be the k-th machine in the flow-shop of the disassembling subsystem, with $k \in \{1,\ldots,N^{\mathrm{d}}\}$, and M^{r}_k be the k-th parallel machine in the refurbishing subsystem, with $k \in \{1,\ldots,N^{\mathrm{r}}\}$.

In this paper, the scheduling of manufacturing activities is considered, assuming that the disassembling and refurbishing operations in the remanufacturing plant have been already suitably planned and scheduled. For this reason, the flow of raw components from the remanufacturing system to the manufacturing system is assumed given, both for what concerns times and quantities.

The manufacturing system has to produce a set of orders that are due at the end of the considered planning horizon, conventionally set to $[0,T]$, being T the optimization horizon for the scheduling problem. This corresponds to process a set of jobs, consisting of operations that transform raw components into final products, and to adopt a common due-date model, since all the jobs should be completed before the end of the planning period.

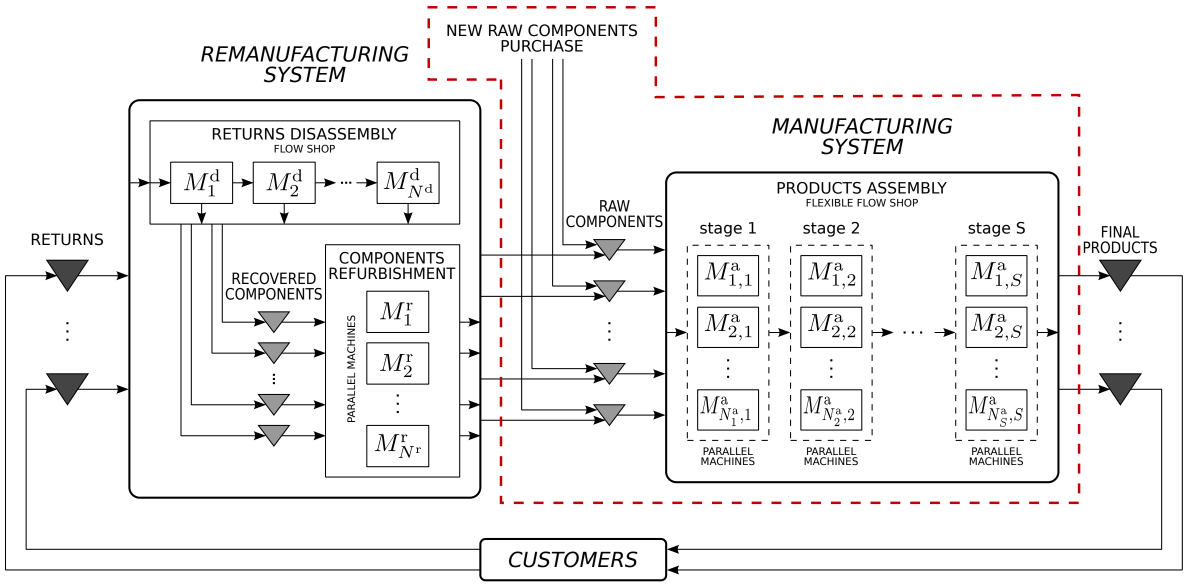

Figure 1: The integrated manufacturing/remanufacturing (“man/reman”) system.

Let J be the set of jobs and let i, j be two indexes used to denote a job. A job i, $i \in J$, is characterized by the lot size L_i and by a priority coefficient W_i (weight) which, in this paper, is assumed to be a fractional number proportional to the size of the lot, that is, $W_i = L_i / \sum_{j \in J} L_j$. The processing of a job consists of operations (one per stage) which consume raw components in order to assemble the lot of final products. Let R be the set of raw components and let:

- Q_{isr}, with $i \in J$, $s \in \{1, \ldots, S\}$, $r \in R$, be the number of raw components of type r which are required to execute job i at stage s; Q_{isr} is defined as $L_i \cdot Q^{\mathrm{U}}_{isr}$, being Q^{U}_{isr} the (given) number of components r that are required to produce one unit of job i at stage s (unitary component requirement);
- P_{isk}, with $i \in J$, $s \in \{1, \ldots, S\}$, $k \in \{1, \ldots, N^{\mathrm{a}}_s\}$, is the processing time of job i at stage s when assigned to machine k; P_{isk} is defined as $L_i \cdot P^{\mathrm{U}}_{isk}$, being P^{U}_{isk} the (given) time to produce one unit of job i on machine k at stage s (unitary processing time).

The remanufacturing plant carries out some disassembling and refurbishing operations aimed at recovering raw components from returned products. As previously discussed, the activities of the remanufacturing system are independently planned and scheduled. Therefore, it is here assumed that the arrival times of raw components from the remanufacturing plant are known in advance. Let H be the planned number of arrivals of lots of raw components, and let:

- D_h, with $h \in \{1, \ldots, H\}$, be the time instant of the h-th arrival (from the remanufacturing system) of lots of raw components;
- M_{rh}, with $r \in R$, $h \in \{1, \ldots, H\}$, be the number of raw components of type r for the h-th arrival.

Raw components are stored in inventories, one per each kind of components; the value I^0_r, with $r \in R$, denotes the initial inventory of raw components of type r. It is worth observing that such inventories contain both components recovered by the internal remanufacturing plant and the ones acquired from external suppliers when the recovered components are not sufficient to satisfy the production requirements of the manufacturing plant.

The considered scheduling problem has the objective of minimizing the sum of the following cost terms:

1. weighted number of tardy jobs;
2. variable purchase costs;
3. fixed purchase costs.

The second cost term is weighted by the unitary cost of raw materials and also by a coefficient which is inversely proportional to the time instant at which the supply is needed. This is justified by the fact that the possible acquisitions of new components from external suppliers are better planned if such components are needed as late as possible in the planning period $[0, T]$); in this way, indeed, the additional supply orders have a greater chance of being delivered on time since they are issued at the beginning of the planning period. The third cost term is proportional to the number of orders from the external suppliers and it is justified by the convenience of grouping the acquisition of the different kinds of raw components. The first cost term is the primary objective and therefore it is weighted by a parameter, denoted as G, whose value is chosen "sufficiently large" to give to this term a lexicographic priority with respect to the other terms. In addition, the second and the third term are respectively weighted by coefficients h_r, $r \in R$, and H: the former represents the cost of a unit of raw material of type r purchased from external suppliers, whereas the latter represents the fixed cost to be paid for any distinct purchase of raw materials from external suppliers.

Such a problem can be modelled as a mixed integer mathematical programming problem as described in the following subsection.

2.1 The Mathematical Programming Formulation

The system under concern can be viewed as a discrete-event system whose state changes at some discrete instants (of the continuous-time axis) corresponding to the beginning of operations of jobs on the various machines of the flexible flow shop. In this regard, the state is basically represented by the list of jobs that have started their execution (one list per each stage of the system), and by the inventory level for each class of raw materials; the system state is updated at each of the discrete instants, by adding the job that starts its processing (at a certain stage) to the list of started jobs, and by updating the inventory levels according to the number of raw materials required by the starting job, the size of the lots of refurbished raw components that are possibly provided by the remanufacturing plant, and the amount of new raw components that are purchased from the external suppliers if necessary. Thus, a continuous-time event-based approach is here proposed to solve the scheduling problem. The number of events is $N = S \cdot |J|$, being $|J|$ the number of jobs to be processed in the flow shop.

The decision variables are the following:

- $s_{is} \in \mathbb{R}$, $s_{is} \geq 0$, with $i \in J$, $s \in \{1,\ldots,S\}$, represents the start time of job i at stage s;
- $x_{ijsk} \in \{0,1\}$, with $i,j \in J$, $i \neq j$, $s \in \{1,\ldots,S\}$, $k \in \{1,\ldots,N_s^{\mathrm{a}}\}$, is a binary sequencing variable such that $x_{ijsk} = 1$ if both job i and job j are processed at stage s by machine k and job i is sequenced before job j, namely

$$x_{ijsk} = \begin{cases} 1, & \text{if } i,j \text{ assigned to } k \text{ and } s_{is} < s_{js} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

- $w_{isk} \in \{0,1\}$, with $i \in J$, $s \in \{1,\ldots,S\}$, $k \in \{1,\ldots,N_s^{\mathrm{a}}\}$, is a binary assignment variable such that $w_{isk} = 1$ if job i is processed by machine k at stage s;
- $u_i \in \{0,1\}$, with $i \in J$, is a binary variable such that $u_i = 1$ if job i completes after its due date (tardy job), namely

$$u_i = \begin{cases} 1, & \text{if } s_{iS} + \sum_{k=1}^{N_S^{\mathrm{a}}} P_{iSk}\, w_{iSk} > T \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

- $y_{ise} \in \{0,1\}$, with $i \in J$, $s \in \{1,\ldots,S\}$, $e \in \{1,\ldots,N\}$, is a binary variable associating events with jobs; $y_{ise} = 1$ if the processing of job i at stage s starts in correspondence of event e;
- $t_e \in \mathbb{R}$, $t_e \geq 0$, with $e \in \{1,\ldots,N\}$, represents the time of occurrence of event e;
- $z_{he} \in \{0,1\}$, with $h \in \{1,\ldots,H\}$, $e \in \{1,\ldots,N\}$, is a binary variable such that $z_{he} = 1$ if the time of occurrence of event e does not precede the h-th arrival of raw components from the remanufacturing system, namely

$$z_{he} = \begin{cases} 1, & \text{if } t_e \geq D_h \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

- $a_{re} \in \mathbb{R}$, $a_{re} \geq 0$, with $r \in R$, $e \in \{1,\ldots,N\}$, represents the number of new raw components of type r, purchased from the external suppliers, arriving at instant t_e (since needed to execution of the job associated with event e);
- $\alpha_e \in \{0,1\}$, with $e \in \{1,\ldots,N\}$, is a binary variable such that $\alpha_e = 1$ if any new component from the external suppliers arrives at instant t_e, namely

$$\alpha_e = \begin{cases} 1, & \text{if } \sum_{r \in R} a_{re} > 0 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

- $i_{re} \in \mathbb{R}$, $i_{re} \geq 0$, with $r \in R$, $e \in \{1,\ldots,N\}$, represents the number of raw components of type r in the inventory after the occurrence of event e at instant t_e (inventory level).

The mathematical programming model is the following.

$$\min \left\{ G \sum_{i \in J} W_i u_i + \sum_{e=1}^{N} \left[(N-e) \sum_{r \in R} h_r\, a_{re} \right] + H \sum_{e=1}^{N} \alpha_e \right\} \tag{5}$$

subject to:

$$a_{re} \leq V \alpha_e\,, \quad \forall r \in R, \forall e \in \{1,\ldots,N\} \tag{6}$$

$$s_{iS} + \sum_{k=1}^{N_S^{\mathrm{a}}} P_{iSk}\, w_{iSk} - B u_i \leq T\,, \quad \forall i \in J \tag{7}$$

$$s_{is} \geq s_{i(s-1)} + \sum_{k=1}^{N^{\mathrm{a}}_{(s-1)}} P_{i(s-1)k} w_{i(s-1)k} , \quad \forall i \in J , \forall s \in \{2, \ldots, S\} \quad (8)$$

$$s_{js} \geq s_{is} + P_{isk} w_{isk} - B(1 - x_{ijsk}) , \quad \forall i, j \in J , i \neq j , \forall s \in \{1, \ldots, S\} , \quad \forall k \in \{1, \ldots, N^{\mathrm{a}}_s\} \quad (9)$$

$$\sum_{k=1}^{N^{\mathrm{a}}_s} w_{isk} = 1 , \quad \forall i \in J , \forall s \in \{1, \ldots, S\} \quad (10)$$

$$2(x_{ijsk} + x_{jisk}) \leq w_{isk} + w_{jsk} , \quad \forall i, j \in J , i \neq j , \forall s \in \{1, \ldots, S\} , \quad \forall k \in \{1, \ldots, N^{\mathrm{a}}_s\} \quad (11)$$

$$x_{ijsk} + x_{jisk} \geq w_{isk} + w_{jsk} - 1 , \quad \forall i, j \in J , i \neq j , \forall s \in \{1, \ldots, S\} , \quad \forall k \in \{1, \ldots, N^{\mathrm{a}}_s\} \quad (12)$$

$$\sum_{e=1}^{N} y_{ise} = 1 , \quad \forall i \in J , \forall s \in \{1, \ldots, S\} \quad (13)$$

$$\sum_{i \in J} \sum_{s=1}^{S} y_{ise} = 1 , \quad \forall e \in \{1, \ldots, N\} \quad (14)$$

$$s_{is} \geq t_e - B(1 - y_{ise}) , \quad \forall i \in J , \forall s = 1, \ldots, S , \forall e \in \{1, \ldots, N\} \quad (15)$$

$$t_e \geq s_{is} - B(1 - y_{ise}) , \quad \forall i \in J , \forall s = 1, \ldots, S , \forall e \in \{1, \ldots, N\} \quad (16)$$

$$t_e \geq D_h - B(1 - z_{he}) , \quad \forall h \in \{1, \ldots, H\} , \forall e \in \{1, \ldots, N\} \quad (17)$$

$$t_e \leq D_h + B z_{he} , \quad \forall h \in \{1, \ldots, H\} , \forall e \in \{1, \ldots, N\} \quad (18)$$

$$z_{he} \geq z_{h(e-1)} , \quad \forall h \in \{1, \ldots, H\} , \forall e \in \{2, \ldots, N\} \quad (19)$$

$$t_e \geq t_{e-1} , \quad \forall e \in \{2, \ldots, N\} \quad (20)$$

$$i_{re} = I^0_r + \sum_{h=1}^{H} M_{rh} z_{he} + \sum_{f=1}^{e} a_{rf} - \sum_{f=1}^{e} \sum_{i \in J} \sum_{s=1}^{S} Q_{isr} y_{isf} , \quad \forall r \in R, \forall e \in \{1, \ldots, N\} \quad (21)$$

The first term of the objective function (5) corresponds to the weighted number of tardy jobs; the second term is the acquisition cost from external suppliers of new components, where the multiplier $N-e$ penalizes the acquisitions needed early in the planning period; the third term is the fixed acquisition cost. The coefficient G is fixed much greater than parameters h_r, $r \in R$, and H such that the weighted number of tardy jobs is the primary objective. Constraints (6) allow determining the number of purchases from the external suppliers, whereas constraints (7) establish which jobs are tardy. Constraints (8) impose that the start time of any job i at a stage s must be greater than the sum of the start time and the processing time of job i at the previous stage. Constraints (9) define the sequence of the jobs on the machines at each stage; given two distinct jobs i and j, if i precedes j on machine k at stage s then $x_{ijsk} = 1$ and (9) becomes $s_{js} \geq s_{is} + P_{isk} w_{isk}$, imposing that job j starts after the completion of job i; otherwise, if $x_{ijsk} = 0$, the relevant constraint is always satisfied. Constraints (10) guarantee that each job at each stage is assigned to one and only one machine. The joint role of constraints (11) and (12) is that of forcing a sequence order among pairs of jobs that are assigned to the same machine; indeed, if two jobs i and j are both assigned to machine k, the variables w_{isk} and w_{jsk} are equal to 1 and the constraints (11) and (12) are equivalent to $x_{ijsk} + x_{jisk} = 1$, then either "i precedes j" or "j precedes i" and machine k. Constraints (13) and (14) state that only one event can be associated with the start time of a job and viceversa. Constraints (15) and (16) associate the timee of occurrence of the events with the start times of the jobs (when $y_{ise} = 1$ such constraints reduce to $t_e = s_{is}$). Similarly, constraints (17) and (18), together with (19), associate the arrivals of components from the remanufacturing systems to events. Constraints (20) impose the time sequencing of events and finally, constraints (21) compute the inventory levels for the components in correspondence of each event.

3 THE PROPOSED MATHEURISTICS

The MIP model presented in the previous section can be used to solve the considered class of problems only in the case of small instances. This problem indeed is NP-hard since it generalizes the problem of minimizing the number of tardy jobs on parallel machines proved to be NP-hard in (Garey and Johnson, 1990). For this reason, in this paper an heuristic algorithm to solve the problem even in the case of

STEP 1	
Goal	Job assignment
Objective (min)	Makespan
Fixed variables	none
Solution (main)	w_{isk}, $\forall i \in J$, $\forall s \in \{1,\dots,S\}$, and $\forall k \in \{1,\dots,N_s^a\}$

STEP 2	
Goal	Job sequencing
Objective (min)	Weighted number of tardy jobs
Fixed variables	w_{isk}, $\forall i \in J$, $\forall s \in \{1,\dots,S\}$, and $\forall k \in \{1,\dots,N_s^a\}$
Solution (main)	x_{ijsk}, $\forall i, j \in J$, $\forall s \in \{1,\dots,S\}$, and $\forall k \in \{1,\dots,N_s^a\}$

STEP 3	
Goal	Components acquisition
Objective (min)	Weighted number of tardy jobs and (variable and fixed) purchase costs
Fixed variables	w_{isk} and y_{ise}, $\forall i \in J$, $\forall s \in \{1,\dots,S\}$, $\forall k \in \{1,\dots,N_s^a\}$, and $\forall e \in N$
Solution (main)	a_{re}, $\forall r \in R$ and $\forall e \in \{1,\dots,N\}$

Figure 2: The structure of the proposed matheuristics.

medium and large instances is proposed. In particular, a matheuristic algorithm based on a decomposition heuristics implemented through the interoperation of different mathematical programming models is introduced in the following.

The basic idea is to decompose the problem decisions in a series of sub-problems. First of all, the decisions on the assignment of the jobs to the available machines are separated from the ones relevant to the sequencing of the jobs on the machines; secondly, the timing decisions are taken considering the component availability constraints only after having assigned and scheduled the jobs at the various stages of the flexible flow shop. Such an approach appears promising having adopted a common due-date model and being the manufacturing/remanufacturing system partially closed. As a matter of fact, a strategy for minimizing the number of tardy jobs in case of a common due-date can be that of balancing the utilization of the machines through a suitable job assignment and then scheduling the jobs trying to minimize the makespan. Moreover, since it is assumed always possible to acquire the needed additional new components from the external suppliers and being the penalization of such additional supplies a secondary objective, then it is reasonable to consider the component availability constraints only in the final timing subproblem.

More specifically, the proposed matheuristics consists of three steps:

1. the jobs are assigned to the machines at each stage of the flexible flow shop with the objective of minimizing the makespan; this problem is solved in a short computation time by a very simple MIP model, presented in Section 3.1, which assigns the jobs on the basis of their processing times P_{isk}; the solution of this step is used to fix the values of the binary assignment variables, i.e., w_{isk}, $\forall i \in J$, $\forall s \in \{1,\dots,S\}$, and $\forall k \in \{1,\dots,N_s^a\}$;
2. the jobs are scheduled on the machines, at each stage of the flexible flow shop, without considering the component availability constraints; to this end a simplified version of the MIP model shown in Section 2.1, as reported in section 3.2, is considered: since the component availability constraints are not taken into account, only the weighted number of tardy jobs is minimized and constraints (6) and (13)÷(21) become not necessary; in addition, w_{isk} are no longer decision variables since they are fixed to the values obtained by step 1; the solution of such a model provides the values of the binary sequencing variables, i.e., x_{ijsk}, $\forall i, j \in J$, $\forall s \in \{1,\dots,S\}$, and $\forall k \in \{1,\dots,N_s^a\}$;
3. the possible additional component acquisitions from the external suppliers are determined on the basis of the component availability constraints; at this step the MIP model presented in section 2.1 is solved, having fixed the values of the binary assignment variables w_{isk}, $\forall i \in J$, $\forall s \in \{1,\dots,S\}$, and $\forall k \in \{1,\dots,N_s^a\}$, obtained at step 1 and also the values of the variables y_{ise}, $\forall i \in J$, $\forall s \in \{1,\dots,S\}$, and $\forall e \in \{1,\dots,N\}$, computed according to the overall order among all the job operations determined by the values of the start time variables s_{is}, $\forall i \in J$, $\forall s \in \{1,\dots,S\}$, found at step 2.

Such an approach, which is depicted in Figure 2, does not guarantee to find an optimal solution; however, the preliminary experiments discussed in Section 5 show that the proposed heuristics is able to find in minutes solutions better than the ones yielded by the MIP solver in one hour.

3.1 The Model for Step 1

The MIP model used at the first step of the proposed matheuristics to determine the assignment of the job operations to the machines for each stage of the flexible flow shop is the following.

$$\min C_{\max} \tag{22}$$

subject to:

$$C_{\max} \geq c_{sk}\,, \quad \forall s \in \{1,\dots,S\}\,, \forall k \in \{1,\dots,N_s^{\mathrm{a}}\} \tag{23}$$

$$c_{sk} = \sum_{i\in J} P_{isk}\, w_{isk}\,, \quad \forall s \in \{1,\dots,S\}\,, \forall k \in \{1,\dots,N_s^{\mathrm{a}}\} \tag{24}$$

$$\sum_{k=1}^{N_s^{\mathrm{a}}} w_{isk} = 1\,, \quad \forall i \in J\,, \forall s \in \{1,\dots,S\} \tag{25}$$

In this model, $C_{\max}$ denotes the makespan, whereas $c_{sk} \in \mathbb{R}$, $c_{sk} \geq 0$, with $s \in \{1,\dots,S\}$, $k \in \{1,\dots,N_s^{\mathrm{a}}\}$, is a decision variable representing the completion time of the last operation on machine k at stage s. The objective function (22) corresponds to the minimization of the makespan, which is defined through constraints (23). Constraints (24) compute, for each stage of the flexible flow shop, the sum of the processing times of all operations assigned to a certain machine, i.e., the completion time of the last operation on such machine. Finally, constraints (25) impose that each job is assigned, at each stage, to one and only one machine.

3.2 The Model for Step 2

The MIP model exploited at the second step for sequencing the operations of jobs on the machines, at each stage of the flexible flow shop, disregarding the component availability constraints, is illustrated in the following.

$$\min \left\{ G \sum_{i\in J} W_i\, u_i \right\} \tag{26}$$

subject to (7)÷(12).

The objective function (26) minimizes the weighted number of tardy jobs. This model corresponds to a relaxation of the MIP model in Section 2.1, where only the variables relevant to sequencing decisions are used, having fixed the assignment variables and neglecting the ones used to model the component availability constraints that link the job operations to the events.

3.3 The Model for Step 3

The MIP model introduced in section 2.1 is employed at the last step of the matheuristics to determines both the timing of the job operations and the acquisitions from external suppliers of the different types of components possibly needed to process the job operations, on the basis of the availability and the usage of the different components.

As previously pointed out, this model is solved at step 3 having fixed the variables w_{isk} and y_{ise} according to the solutions obtained at steps 1 and 2, respectively. In particular, the values of the variables y_{ise} are determined first sorting the values of the N variables s_{is} in non decreasing order (ties are broken arbitrarily) and then assigning the events $e \in \{1,\dots N\}$ to the job operations identified by the pairs (i,s) according to the obtained order. Finally, if an event e is assigned to the job operation (i,s), then it is fixed $y_{ise} = 1$, otherwise $y_{ise} = 0$.

4 EXAMPLE

A small instance of the scheduling problem is considered in this section with the aim of showing the application of the proposed matheuristics.

Let the manufacturing system consists of three stages ($S = 3$) and two machines per stage ($N_s^{\mathrm{a}} = 2$ $\forall s \in \{1,\dots,S\}$). The system processes two types of raw components ($R = \{1,2\}$) and the remanufacturing plant provides two lots of refurbished "as-good-as-new" raw components ($H = 2$); the lot sizes M_{rh} and the arrival times D_h are reported in table 1 (which also includes the initial inventory of raw components).

Five jobs $J_1,\dots,J_5$ must be carried out by the manufacturing system. Values Q_{isr} and P_{isk}, representing respectively the number of raw components r required to execute job i at stage s and the processing time of job i when processed by machine M_{ks}^{a} at stage s, are reported in tables 2 and 3. Moreover, the weights of jobs in the cost function are $W_1 = 0.169$, $W_1 = 0.262$, $W_1 = 0.123$, $W_1 = 0.2$, and $W_1 = 0.246$. Finally, the common due-date for all jobs is $T = 25$.

This small instance of the problem has been solved through the proposed matheuristics. At the first step, jobs are assigned to machines in all the stages of the flexible flow shop; the solution of this step is reported in table 4. In the second step, jobs are sequenced on the assigned machines without taking into account the component availability constraints; the solution of this step is reported in table 5.

In the second step of the matheuristics, the event assignment is also performed, that is, the assignment

Table 1: Example – Lots of refurbished raw components (D_h and M_{rh}) and initial inventory levels (I_r^0).

	I_r^0	$h=1$		$h=2$	
		D_h	M_{rh}	D_h	M_{rh}
$r=1$	500	17.92	221	4.99	160
$r=2$	200		195		106

Table 2: Example – Number of raw components that are required to execute jobs (Q_{isr}).

	$s=1$		$s=2$		$s=3$	
	$r=1$	$r=2$	$r=1$	$r=2$	$r=1$	$r=2$
J_1	88	66	77	99	55	55
J_2	51	68	153	68	17	51
J_3	56	64	48	56	32	24
J_4	91	91	117	52	65	52
J_5	16	128	112	32	64	80

Table 3: Example – Processing times of jobs (P_{isk}).

	$s=1$		$s=2$		$s=3$	
	$M^a_{1,1}$	$M^a_{2,1}$	$M^a_{1,2}$	$M^a_{2,2}$	$M^a_{1,3}$	$M^a_{2,3}$
J_1	4.35	10.04	6.03	7.33	2.85	5.99
J_2	11.66	5.52	9.98	6.35	9.89	6.24
J_3	5.24	5.82	7.41	6.96	5.72	3.13
J_4	10.44	7.9	4.19	9.11	8.26	11.24
J_5	14.09	10.17	5.03	8.03	15.1	11.47

Table 4: Example – Solution of step 1: job assignment.

	$s=1$	$s=2$	$s=3$
J_1	$M^a_{1,1}$	$M^a_{1,2}$	$M^a_{1,3}$
J_2	$M^a_{2,1}$	$M^a_{2,2}$	$M^a_{2,3}$
J_3	$M^a_{2,1}$	$M^a_{2,2}$	$M^a_{1,3}$
J_4	$M^a_{2,1}$	$M^a_{1,2}$	$M^a_{1,3}$
J_5	$M^a_{1,1}$	$M^a_{1,2}$	$M^a_{2,3}$

Table 5: Example – Solution of step 2: job sequencing.

	$M^a_{1,s}$	$M^a_{2,s}$
$s=1$	$J_1 \to J_5$	$J_2 \to J_3 \to J_4$
$s=2$	$J_1 \to J_4 \to J_5$	$J_2 \to J_3$
$s=3$	$J_1 \to J_3 \to J_4$	$J_2 \to J_5$

of job operations to the ordinal numbers of the event from 1 to 15 (in general, from 1 to $S \cdot |J|$) according to the job operation start times. Such an association "(event #, job start)", which is reported in table 6, is obviously made in accordance with the job sequencing shown in table 5. The solution of step 2 is also represented by the Gantt chart illustrated in figure 3.

The solution in figure 3 can be not feasible with respect to the component availability constraints. At step 3, such constraints are actually considered; however, in order to speed up the determination of the op-

Table 6: Example – Solution of step 2: event assignment.

Event #	Associated job start	
	job	stage
1	J_1	1
2	J_2	1
3	J_1	2
4	J_5	1
5	J_2	2
6	J_3	1
7	J_1	3
8	J_4	1
9	J_3	2
10	J_2	3
11	J_3	3
12	J_4	2
13	J_5	2
14	J_4	3
15	J_5	3

timal solution, the assignment/sequencing of the jobs and the assignments of the events to the job operations are fixed to the values provided by the first two steps of the matheuristics. Therefore, the solution of step 3, which is illustrated through the Gantt chart in figure 4, is very similar to the one obtained after step 2; however, it is easy to notice that some operations are delayed to satisfy the component availability constraints. In addition, such delays are also due to the fact that, when new raw components must be purchased from the external suppliers to satisfy the component availability constraints, it is convenient to delay the purchase as much as possible.

In conclusion, the Gantt chart illustrated in figure 4 represents the final solution of the proposed matheuristics.

5 COMPUTATIONAL EXPERIMENTS

The performances of the proposed matheuristics were evaluated considering a set of 25 problem instances that were randomly generated. Instances from #1 to #10 are relevant to a flow-shop of 3 stages, including 3, 2, and 3 machines for the instances #1-5 and 4, 3, and 4 machines for the instances #6-10. The number of jobs is 10 for the instances #1-5 and 20 for the instances #6-10. The instances from #11 to #25 are relevant to a flow-shop of 4 stages, including 4, 3, 4, and 3 machines for all instances; in this case, the number of jobs is 30 for the instances #11-15, 60 for the instances #16-20, and 90 for the instances #21-25. According to such characteristics the considered

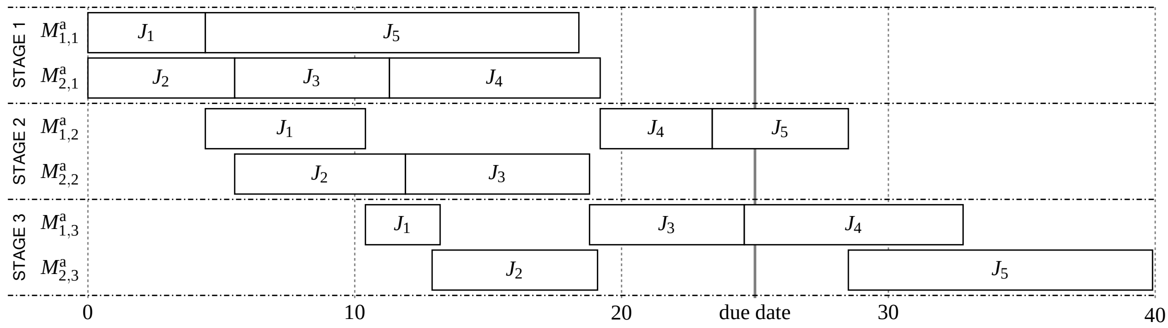

Figure 3: Gantt chart of the solution after step 2.

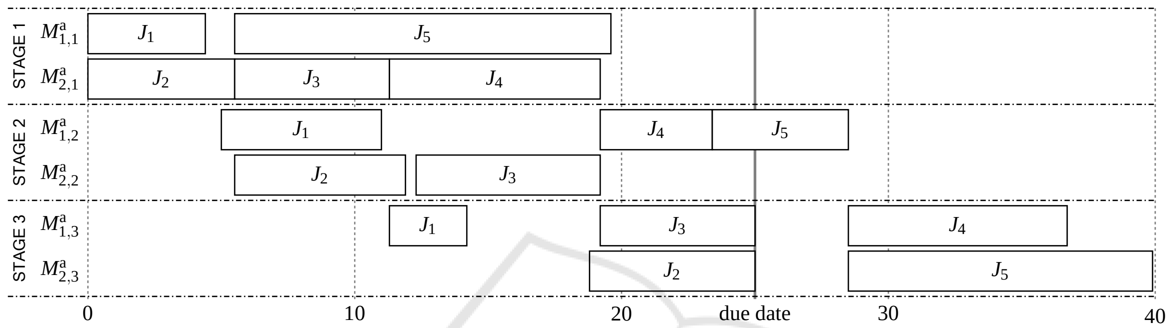

Figure 4: Gantt chart of the solution after step 3 (final solution of the matheuristics).

Table 7: Comparison between Cplex and the proposed matheuristics.

Instance	Njobs	Nstages	Cplex		matheuristics			Comparison
			Opt	CPUOpt	Obj	NumTard	CPU	
1	10	3	92.20	38.6	96.10	9	1.3	+4.2%
2	10	3	99.90	2.7	99.90	10	2.9	0.0%
3	10	3	90.20	61.9	100.20	10	2.0	+11.1%
4	10	3	91.30	762.6	95.60	9	2.9	+4.7%
5	10	3	88.00	345.6	94.40	9	3.8	+7.3%
6	20	3	83.40	3600.1	51.90	10	47.1	−37.8%
7	20	3	79.40	3604.3	40.80	8	43.1	−48.6%
8	20	3	69.80	3600.9	40.00	8	114.3	−42.7%
9	20	3	86.16	3600.4	51.81	10	8.3	−39.9%
10	20	3	91.13	3600.4	45.88	8	11.9	−49.7%
11	30	4	100.87	3601.6	48.79	14	306.9	−51.6%
12	30	4	102.30	3600.9	48.15	11	312.7	−52.9%
13	30	4	94.61	3600.8	52.06	13	308.6	−45.0%
14	30	4	110.20	3599.7	60.72	13	305.7	−44.9%
15	30	4	92.35	3600.6	37.50	8	308.3	−59.4%
16	60	4	152.56	3603.2	95.82	29	607.7	−37.2%
17	60	4	145.46	3603.3	84.48	22	608.2	−41.9%
18	60	4	150.03	3604.6	89.78	27	556.5	−40.2%
19	60	4	145.12	3604.5	85.11	26	559.6	−41.4%
20	60	4	157.49	3603.9	92.53	23	608.9	−41.2%
21	90	4	–	3614.0	169.57	51	627.5	–
22	90	4	–	3614.1	148.80	42	631.9	–
23	90	4	–	3625.6	154.03	45	631.2	–
24	90	4	–	3623.4	136.47	34	631.4	–
25	90	4	–	3623.3	172.11	48	633.5	–

Table 8: Values obtained in each step of the proposed matheuristics.

Instance	Njobs	Nstages	step 1		step 2		step 3	
			Cmax	CPU	WNTJ	CPU	OverallObj	CPU
1	10	3	33.15	0.7	96.10	0.5	96.10	0.0
2	10	3	43.18	1.7	99.90	0.7	99.90	0.5
3	10	3	31.41	0.9	100.20	0.5	100.20	0.7
4	10	3	40.93	0.7	95.60	0.7	95.60	1.5
5	10	3	35.66	2.1	94.40	0.9	94.40	0.7
6	20	3	39.14	0.4	51.90	46.1	51.90	0.6
7	20	3	37.57	0.5	40.80	42.0	40.80	0.6
8	20	3	35.14	0.5	40.00	113.1	40.00	0.7
9	20	3	40.18	0.3	51.60	7.8	51.81	0.2
10	20	3	35.22	0.5	45.00	10.7	45.88	0.6
11	30	4	60.85	2.6	46.90	300.5	48.79	3.8
12	30	4	52.85	3.0	39.70	300.2	48.15	9.4
13	30	4	58.77	2.1	49.40	300.6	52.06	5.9
14	30	4	62.43	1.4	53.50	300.1	60.72	4.2
15	30	4	45.60	0.9	33.40	300.5	37.50	6.9
16	60	4	105.93	1.6	48.40	300.6	95.82	305.6
17	60	4	109.33	2.4	41.80	300.4	84.48	305.5
18	60	4	108.80	1.5	44.40	300.3	89.78	254.7
19	60	4	111.27	2.5	44.80	300.5	85.11	256.6
20	60	4	98.34	3.9	40.10	300.8	92.53	304.2
21	90	4	166.33	1.2	57.30	300.6	169.57	325.8
22	90	4	170.70	8.7	46.70	300.6	148.80	322.6
23	90	4	170.84	5.1	50.90	300.3	154.03	325.8
24	90	4	156.82	5.0	40.00	300.3	136.47	326.1
25	90	4	165.91	4.9	52.10	300.4	172.11	328.2

instances require to schedule a number of operations ranging from 30 to 360. In addition, the MIP model of Section 2.1 needs 2,074 variables (1,830 binary) and 4,560 constraints for the smallest instances and 247,414 variables (244,530 binary) and 602,820 constraints for the largest ones.

The IBM ILOG Cplex 12.6.2 MIP Solver was used to solve the MIP model presented in section 2.1 imposing a time limit of 1 hour. The solver was able to optimally solve only the instances #1-5, whereas it reached the time limit for the instances #6-20, and it did not find any feasible solution for the instances #21-25. The results obtained with Cplex are reported in the columns 4-5 of Table 7.

The three MIP models which constitute the matheuristics were also implemented with the solver Cplex 12.6.2. In this case, a time limit of 5 minutes was imposed for solving each step. The results obtained with the proposed heuristic approach are reported in the columns 6-9 of Table 7. As it can be noted in this table, the instances #1-5 in the first group were optimally solved by Cplex, whereas the matheuristics obtained the optimal solution for instance #2 only; however, for the other four instances in the first group the matheuristics found a slightly larger value (with an average percentage deviation, given by the ratio $(matheuristics\_result - solver\_result)/solver\_result$, of 6.8%). Differently, the matheuristics found for the instances #6-20 very better solutions than the ones provided by Cplex in 1 hour. As it can be observed in Table 7, for the instances #6-20 the matheuristics produced solutions whose overall average percentage deviation from the ones yielded by Cplex is -45.0%. No comparison is possible for the instances #21-25 since the MIP solver was unable to find a feasible solution in 1 hour. For what concern the computational burden of the proposed approach, it is worth noting that the solution of the MIP model at step 2 reached the 5 minutes time limit for the instances #11-25, whereas the solution of the MIP model at step 3 reached the same time limit for the instances #16-25. The details about the solutions provided at each step of the matheuristics and the corresponding CPU times are reported in Table 8 (note that the CPU times reported for the step 3 for some of the instances #16-25 are greater than 5 minutes as they include also the time needed to generate the MIP model).

On the basis of these results, it can be concluded that the proposed approach can be an effective method to find, in an acceptable short time, good solutions for the problem under concern.

6 CONCLUSIONS

In this work, an original matheuristics is proposed to solve a scheduling problem in an integrated manufacturing/remanufacturing system. The matheuristics decomposes the decisions related to the assignment of the jobs to the machines, the sequencing of the jobs on the machines, and the determination of the additional acquisitions from external suppliers of needed components into three separate (but interoperating) mathematical programming models. In this way, it is possible to obtain, in short computation times, solutions to medium-large instances of the problem that are much better than the ones yielded within one hour of computation by solving the complete MIP model. Besides, an event-based formulation is proposed for the considered problem as it is suitable to represent the discrete-event dynamic of the system under concern and the interactions among the remanufacturing plant, the external suppliers and the manufacturing system.

The presented experimental analysis, based on 25 randomly generated instances of the problem of different sizes, points out the effectiveness of the proposed algorithm when few minutes are available to generate a solution. A more extensive tests will be performed in the next developments of this research.

For what concerns future research directions, current activities on this topic are mainly related to the scheduling of disassembling and refurbishing activities in the remanufacturing system, taking into account uncertainties which naturally characterize the return of finished products. In addition, a next step of this research will consider the problem of planning the remanufacturing and manufacturing activities in an integrated way, so defining also the schedule of the remanufacturing system to better match the component requirements issued by the manufacturing system to satisfy the customer demand.

REFERENCES

Artigues, C., Brucker, P., Knust, S., Koné, O., Lopez, P., and Mongeau, M. (2013). A note on "event-based {MILP} models for resource-constrained project scheduling problems". *Computers & Operations Research*, 40(4):1060–1063.

Fleischmann, M., Beullens, P., Bloemhof-Ruwaard, J. M., and Van Wassenhove, L. N. (2001). The impact of product recovery on logistics network design. *Production and Operations Management*, 10(2):156–173.

Garey, M. R. and Johnson, D. S. (1990). *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA.

Grigoriev, A., Holthuijsen, M., and van de Klundert, J. (2005). Basic scheduling problems with raw material constraints. *Naval Research Logistics (NRL)*, 52(6):527–535.

Kolisch, R. (2000). Integrated scheduling, assembly area- and part-assignment for large-scale, make-to-order assemblies. *International Journal of Production Economics*, 64(1–3):127 – 141.

Kolisch, R. and Hess, K. (2000). Efficient methods for scheduling make-to-order assemblies under resource, assembly area and part availability constraints. *International Journal of Production Research*, 38(1):207–228.

Koné, O., Artigues, C., Lopez, P., and Mongeau, M. (2011). Event-based {MILP} models for resource-constrained project scheduling problems. *Computers & Operations Research*, 38(1):3–13.

Li, J., Susarla, N., Karimi, I. A., Shaik, M. A., and Floudas, C. A. (2010). An analysis of some unit-specific event-based models for the short-term scheduling of noncontinuous processes. *Industrial & Engineering Chemistry Research*, 49(2):633–647.

Lund, R. (1998). Remanufacturing: An american resource. In *Proceedings og the Fifth International Congress for Environmentally Conscious Design and Manufcaturing*. Rochester Institute of Technology, Rochester, NY.

Mouret, S., Grossmann, I. E., and Pestiaux, P. (2011). Time representations and mathematical models for process scheduling problems. *Computers and Chemical Engineering*, 35(6):1038 – 1063.

Pinto, J. M. and Grossmann, I. E. (1995). A continuous time mixed integer linear programming model for short term scheduling of multistage batch plants. *Industrial & Engineering Chemistry Research*, 34(9):3037–3051.

Shah, N. (2005). Process industry supply chains: Advances and challenges. *Computers and Chemical Engineering*, 29:1225–1235.

Thierry, M., Salomon, M., Van Nunen, J., and Van Wassenhove, L. N. (1995). Strategie issues in product recovery management. *California Management Review*, 37(2):114–135.

Zapata, J. C., Hodge, B. M., and Reklaitis, G. V. (2008). The multimode resource constrained multiproject scheduling problem: Alternative formulations. *AIChE Journal*, 54(8):2101–2119.