Integration of Hybrid Additive/Subtractive Manufacturing Planning and Scheduling by Metaheuristics

Abstract

Reviewed solutions to model additive manufacturing process planning do not include all types of flexibility required by hybrid additive/subtractive operations and antecedence. In the proposed system the features are extracted from CAD by a rule-oriented system denoting accessibility, surface accuracy, volume merging by functional relations, including deleted datum; the process planner identifies alternative modes to process the operation (operation flexibility) and allows swaps between operations in the routing sequence (sequencing flexibility); the system core generates a precedence graph of manufacturing features introducing flexibility by both conjunctive and disjunctive arcs and nodes, which can enter any position of a subsequence and ranks them by an antecedence analysis. Parametrized relation allows the precedence graph to be visited by a constructive metaheuristic. An Ant Colony Optimization (ACO) has been validated on properly designed instances and literature benchmarks varying job number, alternative operations and disjunctive arcs of routing. Current model reduces the node redundancy of AND/OR graphs and copes with the increased complexity of integrated process planning and job shop scheduling.

Keywords

Additive manufacturing; nonlinear routing; antecedence rules; swarm optimization

1. Introduction

Most Additive Manufacturing (AM) processes are still not net-shape or near net-shape and subtractive operation is necessary. The proposed approach shows how to integrate subtractive operations at the process planning stage, and how to optimize production in a hybrid manufacturing system. Process planning establishes technological requirements and alternative sequences of operations to convert a good from an initial shape to a finished form. Over the past years, process planners have evolved following the development of manufacturing systems, and the aim of productivity is gradually combined with that of sustainability. Product development cost depends on a variety of facilities integrated in the CAD systems, as accessibility analysis for tools and fixturing, functional relations between form features, tolerance specifications, and orientations and datum sequences. Unlike the subtraction process, where a quantity of material is discarded in the form of chips, the additive process usually uses only the material it needs. This potentially affects the sustainability of the production process, as it significantly limits the amount of waste material.

Compared to conventional manufacturing processes, AM processes carry potential competitive advantages, such as material and resource efficiency, part and production flexibility and complex features accessibility (Thompson et al., 2016). Functional requirements are defined and communicated to the manufacturing system by close engineering tolerances. For example, high degree of accuracy and precision is needed on each controlled feature in energy and automotive assembly plant. Although research in the sector is constantly evolving, precision, surface accuracy and mechanical properties obtained by layer deposition cannot yet replace conventional finishing operations to satisfy the functional requirements in high-precision mechanics. Hybrid additive/subtractive manufacturing (Hybrid AM) combines the best features of additive and subtractive manufacturing techniques (Paris and Mandil, 2017).

Layered processes allow packing parts within the working area in parallel batch production. The key role of the datum reference for the operation in feature-based computer aided process planning becomes the part orientation in the working area (Wang et al., 2006; Pande and Kumar, 2008). The set of part orientations identifies the operations to be carried out on a machine, each with his mode of execution. The mode considers the tool/nozzle, the fixture/support elements and the reference datum of the part feature in terms of surfaces/volumes produced by the related operation. According to Oh et al., 2018 and Chen et al., 2018, the accessibility to an undercut is not guaranteed in models based on fused deposition modeling (FDM) and direct energy deposition (DED). Consequently, part reorientations are required.

Figure 1 shows an example where two orientations allow producing the additive part without support structures. The processing times can be drastically reduced by producing the support structure only once and integrating it into the machine setup by the multi-axis hybrid AM machine considered. The tight dimensional tolerance specification must be obtained by finishing cutting.

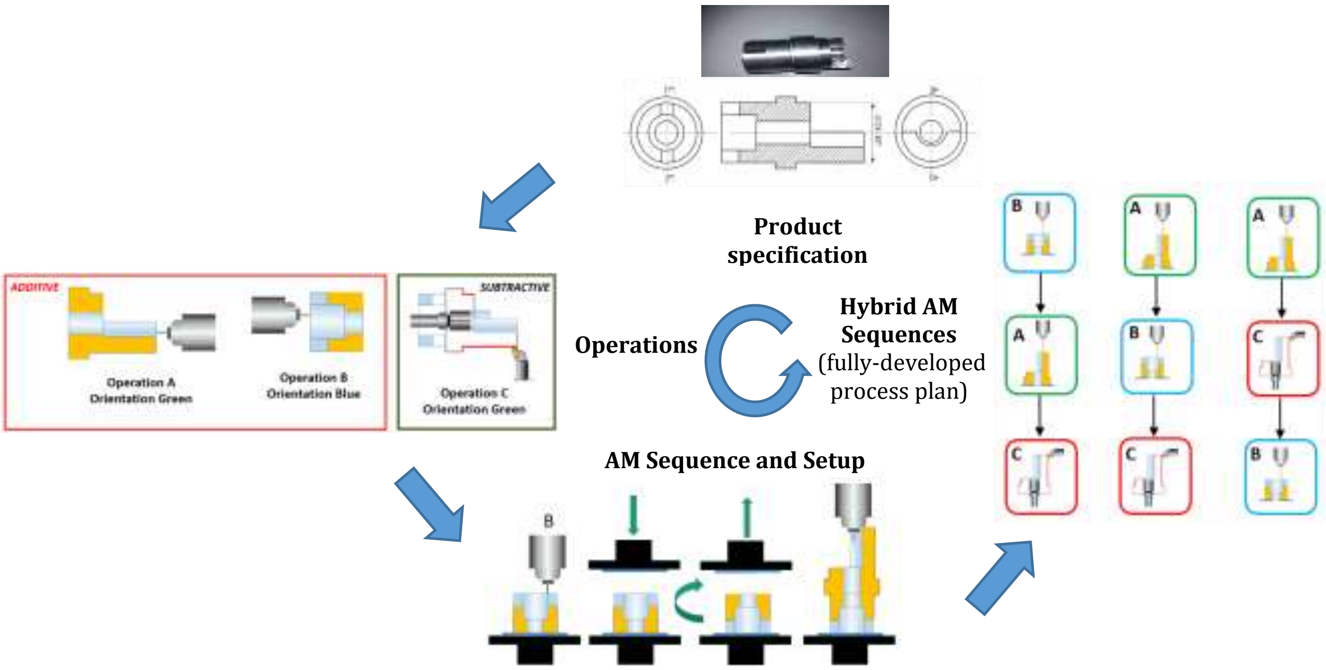


Figure 1. Hybrid AM process planning system integrated with production scheduling (Hybrid AM-IPS): from product specification to manufacturing sequences, by layered manufacturing and turning operations.

To keep up with the evolution of flexible manufacturing systems, a large and continuously increasing number of researches focus on the flexibility of the job shop scheduling problem. Sequencing flexibility is the ability to give alternative operations sequences for the same job and operation flexibility is the ability to select alternative machines and modes (Li et al., 2017; Kim et al., 2017; Birgin et al. 2015). The productivity of a Hybrid AM system depends on the process planner in terms of how much potential of flexibilities he can combine. They are mandatory for the integration between planning and scheduling (IPS) in which the sequence of operations is chosen from a precedence graph to optimize some relevant criteria as minimum time or minimum production costs. The precedence graph covers the *fully-developed* process plan of each job, where the main alternative modes are considered (Halevi and Weill, 2012). IPS includes jobs with many alternative sequences, in addition to alternative machines for operation (Kumar, 2017; Zhang and Wang, 2018; Yang & Hu, 2018). Therefore, the IPS problem to be addressed in a hybrid AM system (Hybrid AM-IPS) considers alternative operations permutable in a sequence of part re-orientations.

Compared to purely AM sequence in Figure 1, new relations must be defined by introducing priority constraints between additive and subtractive operations. They produce constraints typical of the flow shop type of production. However, these constraints must also allow reorder locally the unconstrained operations within the sequence. For example, the fully-developed process plan in Figure 1 shows a *wildcard* operation where the additive operation B (orientation Blue) can be processed in any part of the sequence A-C, i.e. its precedence relation is *immaterial* (Rossi, Soldani and Lanzetta, 2015).

Sequencing flexibility plays a key role. The main advantage is to tackle bottlenecks in critical machines. The operations can be postponed by reordering the job sequence, by opportunely directing their immaterial constraints. The impact of job idle time over the production horizon is reduced. Sequencing flexibility also characterizes flow and job shop-type approaches: different jobs which use the same or different sequences.

In Figure 1, the operation B can be processed after C, the finishing cutting of the feature A (the right sequence), to balance subtractive and additive workstations in the production of two (or more) parts. Figure 2 shows the Gantt chart of 2 parts obtained by using only the central branch (flow-type, diagram at the top) and that one obtained by using also the right branch one (job-type, diagram at the bottom). The job-type of shop scheduling leads to the minimization of the makespan, because even the last scheduled operations (1-B and 2-C) can be processed in parallel.

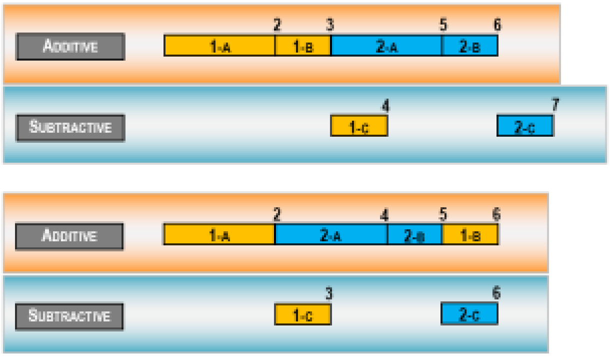


Figure 2. Scheduling of two parts having the process plan presented in Figure 1 on one machine per type. Top case: scheduling by using the entral branch for both parts (flow-type). Bottom case: scheduling by using the right branch for part A and the last OR branch for part C (job-type).

Operation flexibility can be combined to the sequencing flexibility to further reduce production times. Figure 3 shows the impact of a hybrid machine combined with the sequencing flexibility by the Gantt diagram for the jobs in Figure 2, with the subtractive machine replaced by a hybrid machine. The sharp makespan reduction (from 6 to 4 time units) allowed by the introduction of the hybrid machine would have not be achieved if the two subtractive operations were carried out at the end of the cycle, i.e. according to a flow-type planning.

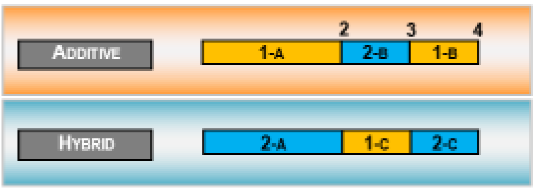


Figure 3. Replacement of a subtractive machine with a more flexible (hybrid) machine. The selection of a job-type planning of job 1 allows to prepend the subtractive operation that can be done only on the hybrid machine, balancing the machine loading.

The Hybrid AM-IPS problem grows with more complexity degrees than the classical job shop scheduling problem as sequencing and operation flexibility must be considered. The simplest job shop scheduling problem is non-polynomial hard (NP-hard). Thus, the IPS problem, from which it derives, is one of the hardest among NP hard optimization problems (Blazewicz et al., 2013). This paper is motivated by the fact that nodes replication occurs in the different branches where the operations are not constrained in the prefixed sequence. AND/OR graphs can reorder operations in subsequences by mean of several alternative OR paths to cover all possibilities. The proposed *P-digraph* uses both conjunctive and disjunctive arcs in the graph model to remove nodes redundancy, implicitly visiting a large amount of alternative sequences.

The non-polynomial complexity problems are effectively approached by metaheuristics. Here, the scheduling problem is approached by a constructive algorithm, the ant colony optimization (ACO), and the effectiveness of the method is evaluated through modified literature benchmarks to include sequencing and operation flexibility at the same time.

This paper is structured as follows. Section 2 includes the literature on process planning systems considering operation and sequencing flexibility. Consideration has been given to hybrid additive/subtractive planning systems. Section 3 defines the problem considered and the properties that characterize it. Section 4 addresses the proposed approach: background and algorithms, which link the planning (CAD features extraction and precedence graph design) and the scheduling (metaheuristics by an ACO algorithm). It describes the rules of conversion between hybrid additive subtractive features and their precedence graph. This graph is also compared to the AND/OR representation. In Section 5, we describe the application to a real case and evaluate the performance of an algorithm based on ant colony on benchmarks from the literature. Section 6 discusses limitations and addresses future research along with conclusions.

2. State-of-the-Art

Process planning problems are formulated as special case of the most representative class of project scheduling (Blazewicz et al., 2013). An important element consists in the precedence constraints among tasks (or jobs). Emerging in different areas, the constraints are used to represent, manufacturing sequences, activity precedence (project scheduling) or parallel and sequential parts in computer programs.

Blazewicz and Kobler (2002) revise the properties of different precedence graphs for scheduling problems based on the *directed graph* representation. A directed graph considers only relations denoted by the symbol “$\prec$”, where $a \prec b$ means that task *a* may be assigned to a processor only after task *b* is finished. The task-on-node depicts the precedence constraints of a directed graph, where nodes correspond to tasks and arcs reflect precedence constraints. This representation is used to a wider extent in manufacturing and computer scheduling problems, where precedence relations are considered (Sinnen, 2014; Bensmaine et al., 2014). AND/OR graph is the conventional model met in an extensive literature (Bentaha et al., 2014; Zhang and Wong, 2016; Wang et al., 2014; Doh et al., 2013; Shao et al.,2009). More authors depict the precedence graph of precedence for multiple process plans in manufacturing, their alternative operations and sequences, and the alternative machines for each operation. AND/OR graphs do not properly consider the sequencing flexibility, unless to explicit all the possible sequences, thus introducing an amount of redundancy in the model (Zhang and Wong, 2016).

Some authors have focused on how to overcome the problem. Leung et al. (2010) propose a complete disjunctive graph to integrate process planning and shop scheduling system. Alternative paths are evaluated by a constructive metaheuristic with the objective to minimize the makespan. Blum and Sampels (2004) use an ant colony to approach the group shop scheduling problems, a mixture of job and open shop scheduling problems. In the group shop scheduling problems, set of operations connected with immaterial precedence constraints are processed in a sequence with the aim of minimizing the makespan of the entire schedule. Aloulou and Artigues (2010) and Amin-Naseri and Afshari (2012) modeled some mixed precedence relations, particularly useful in some cases of stage-based production planning. These mixed relations have been extended to composite relations in the high fashion Made-in-Italy manufacturing (Rossi, Soldani and Lanzetta, 2015). Conversely to the AND/OR approaches, these mixed shop models produce alternative paths simply reordering operations connected by immaterial constraints.

Disjunctive arcs have been recently introduced to approach the problem of reorder the operations in the scheduling phase. They tackle the node replication which becomes intractable in a network completely connected or dominate by immaterial precedence relations.

In general, the level of abstraction considered in the above work makes the operations incomplete or too approximate from the point of view of the manufacturing constraints. So far, process planning models for machining or assembly process have been presented. The information model underlying is consistent in additive machining, where each operation of a job is associated to the subtractive model through the reference datum of a feature-based computer aided process planning. Three elements identify an operation:

1) part orientation (i.e. the datum reference), 2) support structure (i.e. the clamping elements), and 3) nozzle tip (i.e. the tool) (Ahsan et al., 2015; Zhang et al., 2017). Other features of the operation are the slicing and the toolpath (Sparks et al., 2008), which are considered based on colour scaling, graded deposition and surface accuracy (Hofmann et al., 2014). But these features do not directly involve the integration of the process planning with the job shop scheduling, which only affect the processing times. They are considered as attributes or parameters of the operation.

Concerning Hybrid AM, and the feature extraction from a 3D (CAD) model, several authors consider process planning as sequential separate or interchangeable processes (Newman et l., 2015; Frank et al., 2017; Hao et al., 2018). This sequential structure is typical of the flow-shop type of production, where finishing operations by conventional machine tools are postponed to the additive ones. At the other hand, hybrid additive machines become profitable, by offer a sharp of reduction in setup and transportation times between different machines. Recently, however, hybrid machining platforms have been introduced, particularly machining centres integrating the FDM technology (Hansel et al., 2016, Flynn et al., 2016; Chen et al., 2018), which reduce setup times and eliminate transportation times.

It seems that Chen et al. (2018) are the only authors, which propose a process planning for additive and subtractive operations on a hybrid machine. They consider the accessibility problem to free-form surfaces for a part with slender or curvy features and narrow cavities in between, where collision is very likely to occur. They alternate an amount of additive and subtractive operations to tackle the problem. This can be very expensive for the sculptured inner surfaces because it requires a new calibration and certain pre-processing for each usage of additive nozzle or the cutting tool. Accessibility analysis should lead to merge volumes that can be deposited, or surfaces that must be processed with the same reference datum.

Based on the examined literature, operations are not yet modelled as required in Hybrid AM: they are either too simplified to be easily manipulated by the algorithms for integration with the scheduling phase, or they are too fragmented after the CAD extraction phase to prevent seamless integration with the scheduling phase. A gap on the necessary standardization and synthesis of the process to be integrated with a scheduling algorithm emerges. A new approach versus traditional process planning is required.

In this work we operate a standardization of the operation concept: the item of the problem dataset is a unique entity along the whole system, from the CAD extraction to the scheduling phases. The item consistes into the production of one single element of the part, be it a surface or a volume, included in the form feature. Together with the other items which contribute to produce the form feature, it identifies the operation (manufacturing feature). The phase of extraction from the CAD allows datum identification, volume/surface merging and extended accessibility, which allows to consider, for example, a modification of a base surface or a support volume, which will no longer be able to perform its function. The operations are immediately available within the precedence disjunctive graph; if permitted, additive operations may be postponed. The precedence graph is then used by the ant colony optimization algorithm in the scheduling phase.

3. The Integration of Hybrid AM Planning and Scheduling problem

This section presents the problem under consideration and the conditions of feasebility of its solutions.

3.1 Objective function

$$\boldsymbol{min}_{s\in\pi_i}(\boldsymbol{max}_{i=1,..,n}\left\{t_{o_{r_i}}\right\})$$

3.2 Notation

i	job/part, $i\in\{1,..,n\}$
k	resource/machine, $k\in\{1,..,m\}$
j	operation index, j=1,..,R_i (r_i) of the job i
h=(T_h, F_h)	mode $h\in\{1,..,q\}$, adopted by the machine k to process operations with tool/nozzle T_h and fixture/support element F_h

3.2.1 Notation related to the i^{th} job

R_i	number of candidate operations

r_i number of selected operations

$\sigma_i=\{o_{ijkh}\}$ set (collection) of R_i candidate operations (items) $j=1,..,R_i$

P_i precedence graph (σ_i, C, D, E), where σ_i is the set of nodes and C, D and E are the set of arcs

3.2.2 *Notation related to the jth operation of the ith job*

o_{ijkh}, o_j candidate operations; o_{ijkh} explains the mode / machine that uses

S_j form feature produced (surface machined/volume added by FDM)

CS_j datum surface for the mode h ($h \in H_k$) adopted by the machine k

Q_j required surface accuracy or quality

L_j antecedence in the set σ_i

t_{ijkh} , t_j processing time; t_{ijkh} makes explicit mode / machine

H_{jk} set of alternative modes to process the operation j by the machine k

Y_{hk} set of the candidate operations to be processed by the mode h of the machine k

$SU_{hh'}$ setup time to replace tool $T_{h'}$ and fixture $F_{h'}$ of the mode h' with the ones of the mode h

$TT_{kk'}$ transportation time from the machine k' to the machine k

$T\{o_j, (o_{j'}, o_{j''})\}$, triangular relation; two disjunctive arcs $[o_j, o_{j'}]$ and $[o_j, o_{j''}]$ connecting the wildcard operation o_j enclosed by one conjunctive arc $(o_{j'}, o_{j''})$.

Ω dataset; set of data collection, $\Omega = \bigcup_{i=1}^{n}\{\sigma_i\}$

3.2.3 *Notation related to the schedule*

s, s^* feasible and optimal schedule

$\Sigma_i=(o_{ijkh})$ sequence of the (selected) operations of i inclused in s

l_j position of the operation o_{ijkh} inclused in the feasible sequences Σ_i

$st(o_{ijkh})$ starting time of the operation o_{ijkh} inclused in the feasible sequences Σ_i

Π_i , Π_i^1, Πi^2 sets of the feasible sequences of the job i

Σ_i* the sequence of operations of i included in the optimal schedule $s*$

In the Hybrid AM-IPS problem, a set of n jobs, each with his collection σ_i of items forming the dataset Ω of R_i candidate operations o_j, structured as a precedence graph P_i:

$$P_i = \{o_1,..,a_{L_a},b_{L_b},..,o_{R_i} \mid \; L_a \langle L_b, a, b \in \sigma_i\} \quad (1)$$

must be processed on a set of k resources by using a mode $h \in Y_{hk}$ from the set of alternative mode H_{jk}, to find the sequence Σ_i* for each job i of r_i operations:

$$\Sigma_{i*} = \{o_1,..,a_{l_a},b_{l_b},..,o_{r_i} \mid r_i \leq R_i \wedge l_a \leq l_b, \quad a,b \in \sigma_i \} \quad (2)$$

each of them with his processing times t_j, such that the makespan is minimized.

The expression (2) is subjected to the constraints:

1) $r_i \leq R_i$. Considering the alternative modes for operation ($h \in H_{jk}$), the operation $o_{i\,j\,h\,k}$ is a *candidate* for the insertion in Σ_i.
2) $l_a \leq l_b$: two operations $a,b \in \sigma_i$ are ordered in the sequence Σ_i* according to the precedence relation $L_a \langle L_b$, where the *antecedences* L_a (L_b) of the candidate operation a (b) is collected in the dataset Ω.

The relation "$\langle$" extends the directed graph model of Blazewicz and Kobler (2002). L_a and L_b stands for one of the following types of relation, defined in Table 1:

i. $L_a < L_b$: anteriority depicted by the type of arc, C (*conjunctive*, $(a,b) \in C$) or E (*exclusive*, $<a,b> \in E$), which stands for $st(b) \geq st(a)+t_a$, between the starting time of b and the completion time of a
ii. $L_a = L_b$: anteriority depicted by the type of arc, D (*disjunctive*, $[a,b] \in D$), which stands for the immaterial relation, $st(b) \lesseqgtr st(a)+t_a$.

The anteriority $<a,b> \in E$ is used for alternative selection of arcs, similar to the OR branch (Table 1).

Case	***Arc notation***	***Precedence graph***	***Alternative routes***

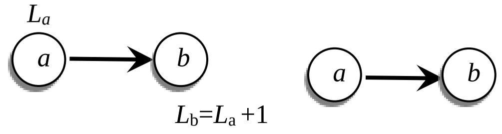
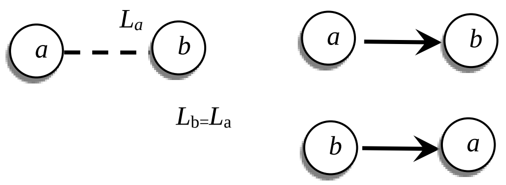
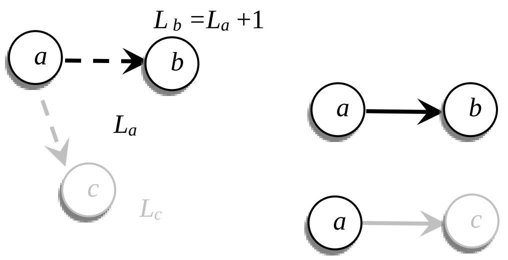
#1	*C*	(a,b)	
#2	*D*	[a,b]	
#3	*E*	{a,b}	

Table 1. Basic precedence relations and their antecedence rules.

According to the antecedences, a precedence graph $P_i = (\sigma_i, C, D, E)$ is embedded in the database **Ω**. The anteriority $(a,b)\in C$ is a close precedence constraint while $[a,b]\in D$ can be understood as one of the two alternatives, $st(b) \geq st(a)+t_a$ or $st(a) \geq st(b)+t_b$, or also its relaxation, where the operations *a* and *b* can be process in parallel. In this paper is adopted the first case: if $st(b) \geq st(a)+t_a$, the operation *a* precedes *b* in the sequence Σ_i, i.e. $l_a < l_b$, where l_a and l_b are the positions of node *a* and *b* in the path Σ_i.

It results $l_a < l_b$, if and only if, *a* and *b* have a close precedence constraint, $(a,b)\in C$. In the case of relations depicted by *D*-arc ($L_a=L_b$), it results $l_a < l_b$ or $l_b < l_a$.

The sequence Σ_i obtained by directing some disjunctive arcs of the precedence graph P_i is *feasible* if all the relations $L_a \langle L_b$ of P_i are satisfied. We denote the set of the feasible sequence with Π_i, in particular, $\Sigma_i^* \in \Pi_i$.

A constraint $L_a \langle L_b$ is violated, if $l_a > l_b$. This happens when some disjunctive arcs have been directing in an arbitrary way and a path which starts from *b* and ends at *a* is generated. Figure 4 shows examples of both anteriority violation and incomplete visit.

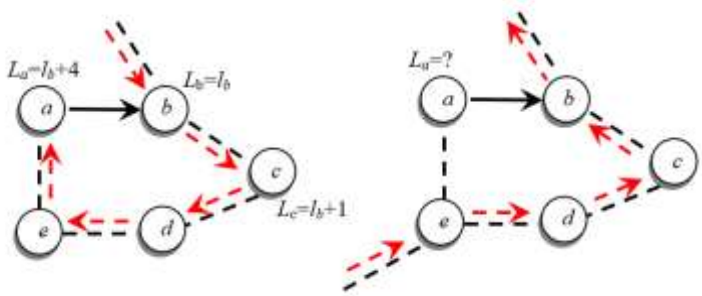


Figure 4. Conjunctive anteriority violation and incomplete node selection by directing in an arbitrary way some disjunctive arcs.

For this reason, the problem is unambiguously represented by the AND/OR graph, which includes conjunctive arcs only. Deadlock-free operations systems can also be obtained by coupling the digraph to a finite-state automata or Petri net [Du et al., 2019].

Figure 5 shows the AND/OR graph of the part of Figure 1, with alternative plans in each OR branches. The part has been obtained by layer deposition with two orientations. Each operation is a node where alternative machines and processing times are indicated. The setup elements are also indicated (on the arc connecting each node). The production results in a flow-type of shop scheduling in the first two branch. The additive operation B (A) is processed after inserting his proper support element on the feature A (B) and overturning the part by 180°. On the contrary, the part is obtained in the last branch by additive manufacturing in two steps because the subtractive operation C (finishing cutting of A) is processed before the feature B. It does not prescribe a linear flow of the part through additive and subtractive machines, resulting in a job-type shop scheduling.

Figure 5 shows a wildcard operation, B. It is interposed in any position of the sequence compelled by the 2 conjunctive arcs. The relation with the connected operations is immaterial as in the *open shop scheduling problem* [Gonzales and Sahni, 1976]. On the contrary, a strong constraint represented by a conjunctive relation, (A, C)$\in C$ is present between the additive and subtractive operations.

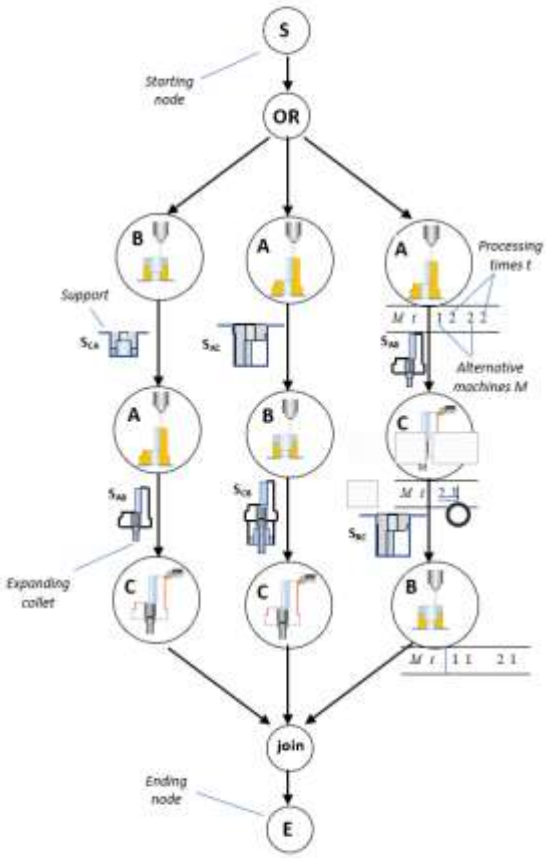


Figure 5. Hybrid AM planning by AND/OR graph. Additive operations A and B (in yellow); subtractive operations C (in red); $S_{j1,j2}$, stands for setup of the operation j_1 after the operation j_2.

Sequencing and operation flexibility lead to a mixed open-job shop scheduling problem. Therefore, the Hybrid AM-IPS problem can be reduced to the flexible job shop scheduling (FJS, Maccarthy and Liu, 1993), where the n fully-develop process plans $\{o_{i\,1\,k\,h},..,o_{i\,R_i\,k\,h}\}$ (i=1,..,n) are non-linear sequences. In Graham notation, it is defined by $J_K|nonlinear(D)|C_{max}$. In FJS, alternative modes to process operations are associated with alternative machines (*partial flexibility*); conversely, if each machine can process all the modes there is a *total flexibility* (Kacem et al., 2002). In the latter case, n jobs are mutually connected with all the q arcs; therefore, the number of disjunctive arcs in the digraph is $O(mqn^2)$ (Rossi, 2014). In Figure 5, partial flexibility is the result of two machines or modes which process operations A (both with processing time of 2) and B (both with processing time of 1).

4. Current work

The proposed system is shown in Figure 6.

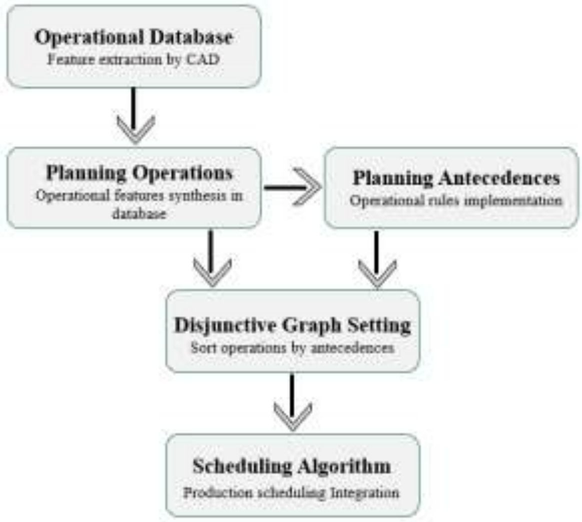


Figure 6. The proposed system

Formally, all modules are linked by the disjunctive graph of precedence defined by the following expression:

$$P\text{-}digraph = \{\bigcup_{i=1}^{n} P_i , Y_{hk}\} = (N, C, D, E, Y_{hk}) \qquad (3)$$

N is the set of operations in Ω plus the dummy start and finishing operations, 0 and *, and the dummy forking nodes; C is the set of conjunctive arcs, and between every last operation on a routing and *; D is the set of disjunctive arcs; E is the set of exclusive arcs between 0 and o_j, with L_j=1, and between o_j and *, with $L_j = R_i$; Y_{hk} is the set of disjunctive arcs between pairs of operations, o_{ijkh}, that have to be processed in some mode $h \in H_{jk}$ on the same resource k; it also includes disjunctive arcs between 0 and o_{ijk*h}, and between o_{ijk*h} and * for all Y_{hk}.

Figure 7 shows an example of the Hybrid AM-IPS instance whose precedence graph is defined by the expression (3). The Gantt char is the same of the one shown in Figure 5, where an operation o_j of the job i is abbreviated by i-j.

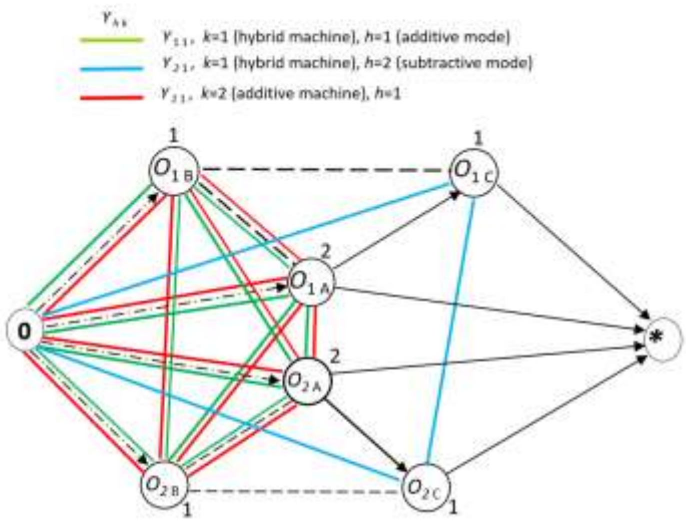


Figure 7. The resulting *P-digraph* for two jobs with the planning sequence shown in Figure 5.

Even if the two alternative representational models produce the same Gantt char, the precedence graph in Figure 7 has less nodes than the AND/OR graph in Figure 5. The AND/OR graph depicts the precedence between nodes by using only conjunctive arcs. Therefore, the nodes are duplicated in the alternative OR-branches with a total number of 9 nodes and 9 arcs. The *P-digraph* has exactly the minimum number of arcs which connect 3 nodes: one conjunctive (O_{1A}, O_{1B}), and two disjunctive arcs, [O_{1A}, O_{1C}] and [O_{1C}, O_{1B}], where O_{1C} is the wildcard operation. This is a *triangular relation* (*T*): three operations connected to one conjunctive arc and two disjunctive arcs joined by the wildcard operation.

The proposed *P-digraph* removes the redundancy by the sequencing flexibility offered by the wildcard operation, which is not constrained in the sequence. Three possible paths obtained by interposing the operation O_{1C} in the sequence compelled by the two conjunctive arcs can be obtained: i) (O_{1C}, O_{1A}) (O_{1A}, O_{1B}); ii) (O_{1A}, O_{1C}) (O_{1C}, O_{1B}); iii) (O_{1A}, O_{1B}) (O_{1B}, O_{1C}).

In *P-digraph*, the machine data (modes and processing times) are allocated on the arcs rather than on the nodes. They are shown by the disjunctive arcs of type Y_{hk}: Y_{11} in green, and Y_{21} in blue; Y_{12} in red. *Theorem 1* guarantees that no relation is violated in a sequence Σ_i obtained by directing some disjunctive arcs [*a*,*b*] of *D* starting from the root.

Theorem 1. Supposing that, $\Sigma_{i\ partial} = \{ o_1,..,a, z_1,..,z_q\}$ is the sequence of the already scheduled operations. An arc $[a,b] \in D$ can be directed according to $(a,b) \in C$ if and only if $l_{z1} = l_{zq} = l_a$.

□ If $\{o_1,..,a, z_1,..,z_q\}$ is the set of already scheduled operations, all the relations, $L_a \langle L_{z1},.., L_a \langle L_{zq}$, are already satisfied between them: $l_a \leq l_{z1} = l_{zq}$.

Introducing $(a,b) \in C$, it results: $\{o_1,..,a, z_1,..,z_q, b\}$. Therefore, all the relations $L_{z1} \langle L_b,.., L_{zq} \langle L_b$ must be satisfied: $l_{z1} = l_{zq} \leq l_b$. As, $l_b = l_a+1$, then $l_{z1} = l_{zq} \leq l_a+1$. Therefore, $l_a \leq l_{z1} = l_{zq} \leq l_a+1$.

Vice versa, if $l_{z1} = l_{zq} = l_a$, the operation b can be added to $\Sigma_{i\ partial}$, only if $l_b \geq l_a+1$. Therefore, the disjunctive $[a,b]$ must necessarily be directed from a to b ■

Let Σ_i be the sequence just obtained when all the operations are included in the schedule. From *Theorem 1* follows that Σ_i is a feasible schedule, i.e. it belongs to Π_i. $L_a \langle L_b$ are met, where a precedes b in the sequence, i.e. the operations of Σ_i are sorted according to their antecedences l_i:

$$\Sigma_i = \{o_1,..,a,b,..,o_{l_i} |\ l_a \leq l_b, a, b \in \sigma_i\} \quad (4)$$

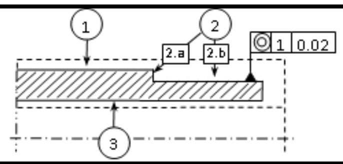
Direct the disjunctive arcs of D according to expression (4) leads to a feasible solution of the same Hybrid AM-IPS problem, defined in Section 3. We denote with Π_i^1 the set of the feasible sequences Σ_i in expression (4).

4.1 Operations and constraints

An operation can be produced by different manufacturing strategies or modes. A mode is defined by a set of attributes, which is comprehensive of large production environments as assembly/disassembly, inspection, coating, painting, forming, welding, washing, sterilizing etc. It includes additive and subtractive operations. In the fully-developed process plan of the job i, the main alternative modes for each operation are inserted in the dataset σ_i.

In this section, we consider items $o_{i\,j\,h\,k}$ of data collection σ_i related to the production of one single element of the part *i*, be it an area or volume, included in the form feature S_{ij}. Together with other items which have the same mode in the set H_{jk}, it contributes to the (candidate) operation o_j to produce the same form feature S_{ij}.

Table 2 introduces an example of the data collection for the part showed at the top. The list of surfaces includes two outer and one inner cylindrical surface, and their modes expressed by a mode description, i.e. tool, fixture and motion of machine axis: rotational (turning) or translational (machining center, FDM).



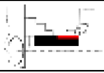
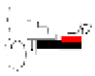
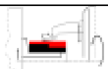
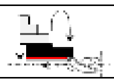
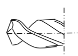
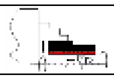
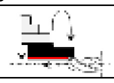
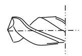
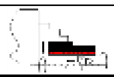
Operation (Surface)	Modes Id (*h*)	Mode Description: Rotational	Mode Description: Translational	Tool (T_h)	Fixture (F_h)	Clamping surface (CS_h)
1	A	External turning 			Chuck	2
2	C	External turning 			Chuck	1
1-2	BD		External turning 		Expanding collet	3
3	E		Spade drilling 		Chuck	1
	F	Internal turning 			Chuck	1
	G		Spade drilling 		Chuck	1
	H	Internal turning 			Chuck	1

Table 2. An example of dataset σ of the part showed at the top.

There are two modes to obtain both surface 1 (A and B) and surface 2 (C and D); there are four modes to obtain the surface 3 (from E to H). The modes B and D are merged in one item because the form feature 1-2

Is processed with the same tool and fixture. To simplify, are omitted the modes B and D taken individually, while it shows the entire form feature BD.

Formally, an item $o_{i\,j\,h\,k}$ related to the production of a surface/volume S_{ij} with his h^{th} alternative mode is defined by the following item:

$$o_{i\,j\,h\,k} = \{\overbrace{h, T_h, F_h, CS_{ij}, t_{i\,j\,h}}^{\text{mode}}\ \overbrace{S_{ij}, Q_{ij}, L_{ij}}^{\text{anteriority}}\} \qquad (5)$$

It is subdivided in two parts: the *mode specification*, at the left side which defines the mode to process it, and the *anteriority specification*, at the right side which allows to explain his precedence relations, with the aim to establish the *P-digraph*.

According Ahsan et al., 2015 and Zhang et al., 2017, three elements identify an operation: $T_{h,}$ CS_{ij} and S_{ij}. A support structure F_h of the machine setup is also considered (see Figure 1). Table 5 shows the dataset σ_i of the problem instance in Table 2.

i	1							
j	1		2		3			
anteced.	L_1		L_2		L_3			
$H_{j\,k}$	A	B	C	D	E	F	G	H
k t_{ijk}	1 24	2 31	1 18	2 9	1 15	2 16	1 16	2 17

Table 3. The dataset σ_i of the instance of the considered problem (in Table 1). The operations *j* of the job *i* are exploded on their modes $H_{j\,k}$. Please note that the antecedents have been identified but not yet been evaluated.

Two or more items of the dataset σ_i which have the same mode specification, have null setup time. They only differ for the surfaces to be machined: the accuracy is the same because it obtained with the same tool. Therefore, they can be merged in a *form feature*, which becomes a single item of the dataset. A *subjob* is the union of single surfaces and form features which have the same clamping surface. It designs one sub-phase of the job.

The starting time of an operation *o* is evaluated by considering the following lag times: i) the job transportation time $TT_{kk'}$ from the machine *k'* to *k* which processed, respectively, the previous operation *o'* and the current operation *o*, and ii) the sequence-dependent setup $SU_{hh'}$ to setup the machine *k* by replacing the previous mode *h'* with the mode *h*. Therefore, the starting time depends on the longest time between the transportation and setup phase:

$$st(o) = max\{t_{o'} + TT_{k\,k'},\ to'' + SU_{h\,h'}\} \qquad (6)$$

Table 4 shows an example of setup timetable for the fully-developed process planning in Table 2. It is compiled according to four multipliers of the time unit τ, which increase based on the difficulty of the setup activity: i) tool/nozzle, ii) clamping surface/orientation change (a new subjob), iii) moving part from machine 1 to 2 (and vice versa), and iv) fixturing/support change required.

Modes			**A**	**BD**	**C**	**E**	**F**	**G**	**H**
	Motion		**FD**	**CT+FD**	**FD**	**FD**	**FD**	**CT+FD**	**CT+FD**
		Clamping / Tool							
A	**CT**			∞	2	2	2	4	4
BD	**-**		∞		∞	6	6	6	6
C	**CT**		2	∞		1	1	4	4
E	CT		2	6	1		∞	∞	∞
F	CT		2	6	1	∞		∞	∞
G			4	6	4	∞	∞		∞
H			4	6	4	∞	∞	∞	

Table 4. Setup times (in τ units of time). Clamping motion facilities associate modes to machine: CT stands for cutting motion, FD for feeding motion; 1 stands for turning center, 2 stands for machining center. Infinite (∞) setup times are used between two alternative modes.

Figure 8 shows the antecedences, L_{ij}, for the part in the Table 1. The concentricity tolerance leads to two possibilities for the process plan: i) surface 3 is the clamping surface for both surfaces 1 and 2 (i.e. the form feature 1-2); ii) the reference surface 1 is the clamping surface of surface 2, i.e. must be machined by finishing turning before clamping the raw surface 2. Considering the first case, there are two conjunctive arcs: (3,1) and (1,2); from the latter case, there are two conjunctive arcs: (1,2) and (1,3). Therefore, the following arcs exist: i) one arc of D, denoted by (1,3), and its reverse arc (3,1) and, ii) one arc of C, (1,2). Besides, the hole 3 can be machined as the last, introducing the D-arc [2,3]. Therefore, the surface 3 is a wildcard operation. Figure 8 shows the related triangular relation T{3, (1, 2)}.

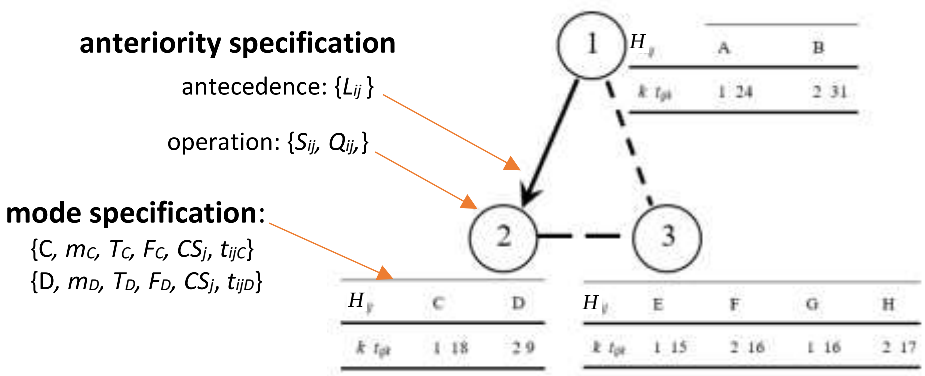


Figure 8. A precedence graph P_i. The nodes are operations, the arcs are antecedences and the label on nodes are the mode specification H_{jk} of the node j on the machine k, which includes the processing time t_{ijjk}

The anteriority specification does not depend on the mode h. Therefore, all the items which have equal right side can be grouped in a node and this node is connected to as many labels as many items there are, each obtained by varying the left side (i.e. the mode specification). The mode specification is the label on node (one for each mode) which lists all the alternative pairs of mode / machine and their processing times.

Therefore, the *P-digraph* graph includes antecedences on the arcs and mode specification by labels on the nodes. The nodes are operations unambiguously identified by the pair of surfaces, S_{ij}, and its required accuracy, Q_{ij}.

4.2 Antecedence rules

To establish relations from two operations *a* and *b* with antecedences, L_a and L_b, the process planner acts on the parameters which identifies the operation (i.e. surface and its accuracy) and on its mode specification. The following basic rules are considered.

i. *Functional relation* (rule *a*)

We consider the plane and axial symmetrical part features which derive from the functional requirements of the mechanical part designed by geometrical tolerance specification (ISO/TC 213). The considered features have a well-defined direction of accessibility that allows to standardize and optimize the hybrid process.

The surfaces which have a functional relation between them, must be included in the same subjob and, therefore if possible, they must have the same clamping surface. Disjunctive arcs of type *D* connect these operations. Alternatively, the datum is used as the clamping surface for the subjob which includes all the surfaces linked in the tolerance specification. Conjunctive arcs connect datum with all the linked surfaces.

For example, the surface associated with a geometric tolerance and the datum surface must be included in the same subjob. If not, we refer to the Accessibility rule (c): the subjob which include the datum must be machined before the subjob which include all the surfaces referred to that datum.

Figure 9 shows an example of this rule obtained by the part in Table 2, where a tolerance of concentricity between the surfaces S_1 and S_2 is required. Therefore, S_2 and its datum S_1 must be included in the same subjob obtained by clamping the part on the surface S_3.

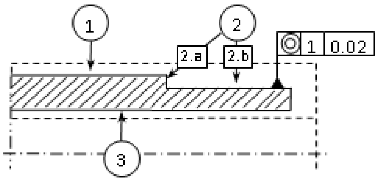


Figure 9. An example of functional relation rule. The concentricity tolerance with datum the surface S_1 generates a subjob denoted by $S_1 \cup S_2$.

ii. *Merging* (rule *b*)

This rule is performed in two consecutive steps: *form feature recognition* and *node merging*.

i. *form feature recognition*: all the items of σ_i which differ only for the surface/volume S_{ij} and the surfaces/volumes are contiguous, are grouped in a node of the digraph. They will be considered as an operation, i.e. an unique item in the dataset (Figure 10). Notice that these items have equal mode specification.

$$a = \{ h, T_h, F_h, CS_a, t_{a\,m}, S_a, Q_a, L_a \}$$

$$= \qquad \neq \qquad = \qquad ab = \{ h, T_h, F_h, CS_a, t_{a\,m} + t_{b\,m}, S_a \cup S_b, Q_a, L_a \}$$

$$b = \{ h, T_h, F_h, CS_b, t_{b\,m}, S_b, Q_b, L_b \}$$

Figure 10. Merging rule for form features

ii. *node merging*: all the items of σ_i which have the same anteriority specification are grouped in a node of the digraph. According to *P-digraph* definition, they are connected to as many disjunctive arcs of Y_{hk} as many modes per resources are, which cover the different items H_{jk}, j=1,..,R_i (see Section 4.3).

After applied this rule, all the remaining items of the dataset are form features. In addition, they are subjobs because of the equal mode specification of the original items, particularly, the clamping surface.

At this conceptual stage the subjob is the operation. If the unique applied rule is the Merging rule (*b*), only nodes disconnected or connected with disjunctive arcs which covers their mode specifications are present. They are the arcs which connect operations of the set H_{ih} with each other. Notice that no routing arcs of *C* are included by rule (*b*).

iii. *Accessibility* (rule *c*)

Each form feature must be accessible through the tool and equipment to allow the associated operation to be processed according to the planned mode. Figure 11 shows an operation (S_b) must be accessible by the tool ($T_{h'}$) and its clamping surface (CS_b) must be accessible by the fixture ($F_{h'}$).

Each clamping surface CS_j makes accessible the operation *j* which belongs to, by gripping CS_j for the processing. A conjunctive arc is directed from the operation related to the clamping surface to the operation *j*. It is possible that this relation can be reversed when the operation *a* produces a volume/surface which is a base/clamping surface of the operation *b*: $CS_b = S_a$. In this case the two opposite *C*-arcs are replaced by a *D*-arc.

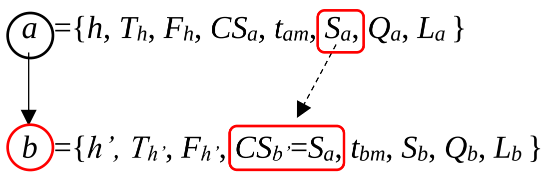


Figure 11. Accessibility rule. The first operation *a* makes the accessibility of both the tool $T_{h'}$, and the fixture $F_{h'}$.

The accessibility by the tool is basic for inner surfaces or holemaking: a sequence of operations using gradually less bulky tools must to be performed to meet design specification. A path of conjunctive arcs connects these operations.

For example, the motor of a refrigerator can be extracted only after the cover has been removed. In machining operations, facing makes holemaking accessible, spot drilling must be processed before twist drilling etc. Also, cylindrical surfaces obtained by turning (cutting motion possessed by the part) makes milling operations accessible on such surfaces (cutting motion possessed by the tool), to avoid interrupted cutting, and so on. These are examples of conjunctive arcs, *C*, where the completion time of the first operation precedes the starting time of second operation.

The accessibility by the tool results also for surface accuracy: the value of Q_a makes accessible a more accurate operation *b*, i.e. $Q_a < Q_b$. Consistent with the correct processing sequence, a conjunctive arc is directed from the operation with worse surface accuracy to the operation with better surface accuracy for the same surface (Figure 12).

$$a = \{ h, T_h, F_h, CS_a, t_{a\,m}, S_a, Q_a, L_a \}$$

$$\downarrow \qquad\qquad <$$

$$b = \{ h', T_{h'}, F_{h'}, CS_b, t_{b\,m}, S_b, Q_b, L_b \}$$

Figure 12. Relation by accuracy rule, $Q_a < Q_b$.

Other examples are: i) the finishing cutting operation will be processed after the roughness operation of the same surface is completed: an arc *C* starts from the roughness operation and ends to the finishing operation; ii) in a sterilization process, an arc *C* starts from the washing operation and ends to the sterilizing operation.

At the current stage, all the nodes are connected from them by construction. In fact:

- the clamping surface nodes are interconnected due to the accessibility by clamping surface;
- at least one path connects the other nodes due to the accessibility by tool.

Therefore, it exists a precedence graph that connects all nodes. Figure 8 shows an example of the graph obtained at the current stage, i.e. after application of the rules from *a* to *c*.

iv. *Clamping removing* (rule *d*)

A clamping surface CS^* can be removed, if and only if, at least one of the following two conditions is verified:

a. At least an alternative clamping surface must be present for the all the operations related to the presence of CS^*; alternatives clamping surfaces are connected by disjunctive arcs, *D*;
b. If exists an operation o_j which has not alternative clamping surfaces except CS^*, it must be processed before CS^* is removed. The rule *d* prescribes an arc of type *C*, (o_j, o_{j^*}) in the *P-digraph*, where o_{j^*} is the operation which processes CS^*, i.e. $\boldsymbol{o_{j^*}} = \{h, T_h, F_h, CS_{oj^*}, t_{ijh}, \boldsymbol{CS^*}, Q_{ij}, L_{ij}\}$.

v. *E-arcs* (rule *e*)

The dummy source node is connected by *E*-arcs to an operation such that no arc of *C* ends to it. The antecedences *L* assigned to them has the value of 1, considering l_O=0; they are also termed as the *first operations*.

The antecedence rules applied in sequential manner, from *a* to *e*, lead to a precedence graph, i.e. the *P-digraph* is completely defined.

For as an example, it is considered the case in Table 2. It leads to the *P-digraph* in Figure 13. The following rules are applied:

- Functional relation rule (a). As mentioned in Section 4.1, the tolerance of concentricity leads to the triangular relation, $T\{3,(1,2)\}$, in Figure 13 (at the left side).

- Merging rule (b). According to this rule, the form features are obtained by the surfaces which have the same mode specification. For example, Table 2 shows the form feature 1-2 obtained by the surfaces S_1 and S_2 which have the same mode specification $m_{BD}=(T_{BD}, F_{BD}, S_3)$.
- Accessibility rule (c). According to this rule, the tool T_1 can access to the surface S_1 by clamping on the surface S_2 (i.e. $CS_1= S_2$). Similarly, the setup for the surface $S2$ is the clamping surface S_1 (i.e. $CS_2= S_1$). Therefore, the operations 1 and 2 (which produce the two alternative clamping surfaces) would connect them by a *D*-arc, if they were not already connected by a stronger relation, i.e. an arc of *C*.
- E-arcs rule (e). The operation 1 is no reached by arcs of type *C*; the operation 3 is only connected to arcs of type *D*. Therefore, the operations 1 and 3 are connected to the source node by *E*-arcs.

Figure 13 also shows the equivalent AND/OR graph (right side). The part can be processed by five different routing represented by five branches, where two branches are nested in the left (1-2-3) and in the central (3-1-2) OR branches. Each of these two has two alternative routing: i) the operations 1 and 2 routed by the antecedence ($L_1 \wedge L_2$), in the right sub-branch, and ii) the form feature1-2 routed by the antecedence $\neg(L_1 \wedge L_2)$, in the left sub-branch. The last branch interposes the operation 3 between 1 and 2.

Figure 13 also shows the node 1-2, his mode m_{BD} in a node label, and the arcs of alternative routing, (3,1-2) and (1-2, 3).

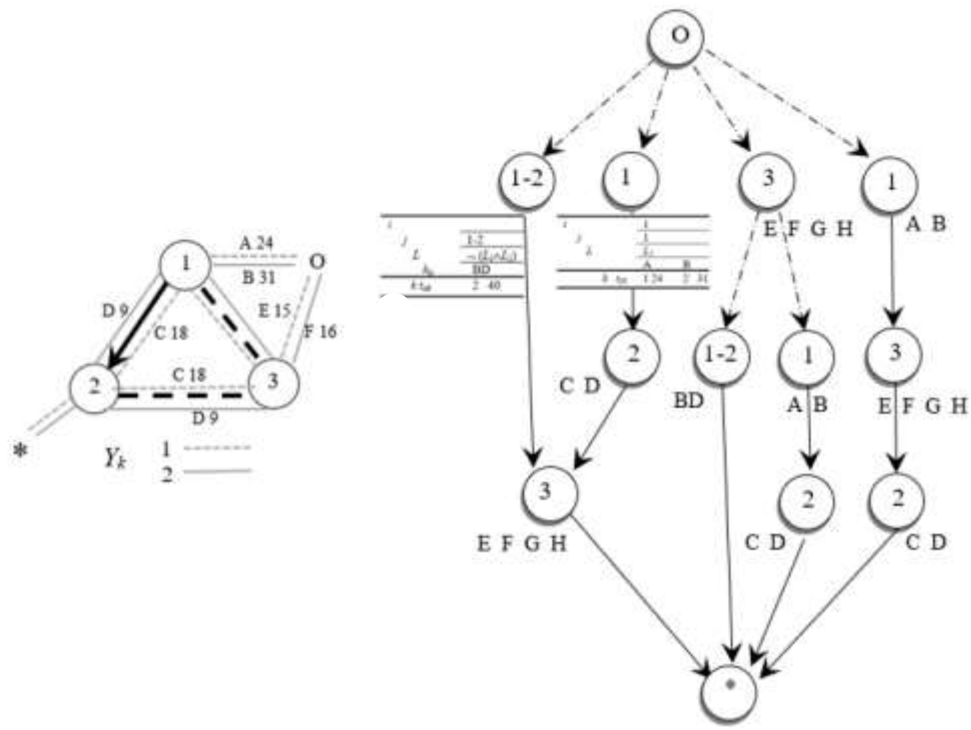


Figure 13. Redundancy saving in nonlinear job routing for the part in Table 2: *P-digraph* (left hand) and his alternative representation AND/OR graph (right hand). A wildcard operation 3 appears at each position of the sequence in the OR branches. The modes are labels on the nodes in the AND/OR graph, and they are showed only for the nodes 1 and 1-2.

4.3 Disjunctive graph setting

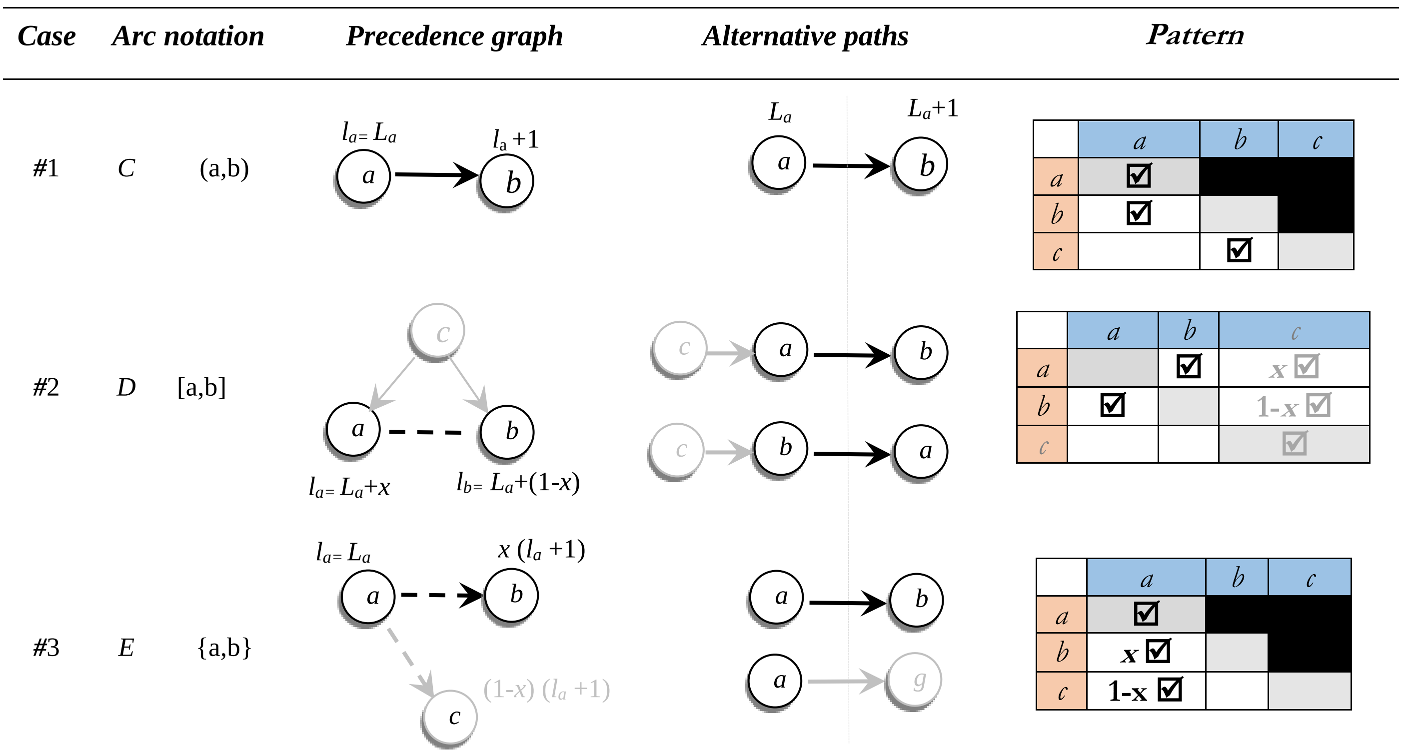
According to expression (4), this section describes the approach to produce the feasible sequences Σ_i sorted according to their antecedences l_i. A constructive algorithm *LS** is proposed that sets the antecedences values, L_j, on the *P-digraph*. It is a list scheduling algorithm which works with elements included in the cells of the square matrix, $\sigma_i \times \sigma_i$, of side of the dataset. This matrix maps the relations of the *P-digraph*. The list scheduling algorithm, originally proposed by Giffler and Thompson (1960) for the flow shop scheduling, starts from an empty solution and generates a feasible solution in $O(r_i)$.

The pseudo-code is shown by Algorithm 1. It looks for *parametrized* antecedences on the nodes of triangular relations or belong to single disjunctive arcs. Table 5 shows the types of antecedence and their *Alternative paths*. Compared with Table 1, two new cases #4 and #5 are reported. Case #4 represents the triangular relation with *b* as the wildcard operation. Case #5 ensures the selection of a random sequence of priorities among three nodes; if the number of operations connected by immaterial relations is n>3, in a time O(n) their permutation must be selected.

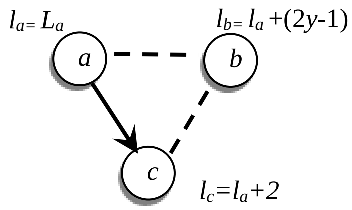
Pattern column maps the three-nodes relations on the cells of a 3x3 matrix, which reflect the parametrized antecedence of types: i) $l=x+L$, and ii) $l=L+(2y-1)$, where:

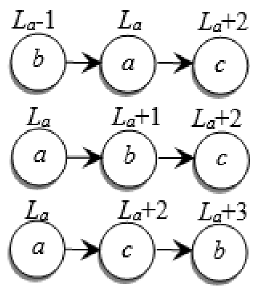
- $x\in\{0,1\}$ is a binary digit to direct arcs of types D (or to select arcs of type E), knowing that the disjunctive arcs can be run in both directions;
- $y\in\{0,1,2\}$ is a ternary digit to establish a routing in a triangular relations of type T_1 and T_2.

The pattern mapping is used to associate special cell places to the parametrized antecedences of the nodes in the *P-digraph*. In the *Pattern* column, the operations in the main diagonal precede the others. They are operations connected with E-arcs to the root O. Their antecedence l=1 implies maximum priority of selection by LS^*. Besides, a check mark in the cell (w,j) stands for the possibility to process the operation in the row w after the operation in the column j by a mode $h\in H_{jk}$ for the operation o_{iwhk}.



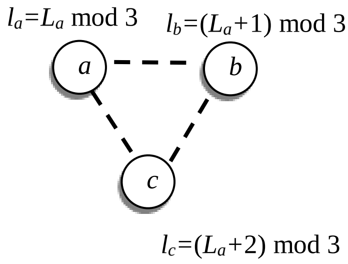
#4 T_1 {b,(a,c)}

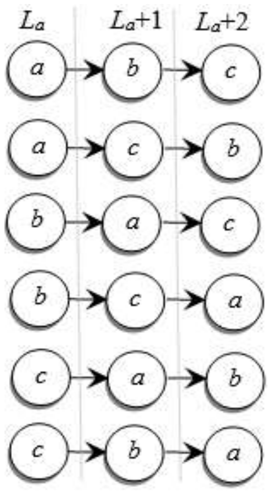




	a	*b*	*c*
a	***x*** ☑		☑
b	☑		***x*** ☑
c	☑	☑	1-***x*** ☑

#5 T_2 [a,b,c]





	a	*b*	*c*
a	☑	☑	☑
b	☑	☑	☑
c	☑	☑	☑

Table 5. Basic and triangular precedence relations and their antecedence rules. $L_a \geq 1$, $x\{0,1\}$, $y \in \{0,1,2\}$

Table 6 shows an example of the square matrix, $\sigma_i \times \sigma_i$, of the problem instance originally proposed in the Table 2.

i		1							
	j	1		2		3			
	anteced.	L_{11}		L_{12}		L_{13}			
	H_{jk}	A	B	C	D	E	F	G	H
j	1	A	B			E	F	G	H
	2	C	D			E	F	G	H
	3	E	F	G	H	E	F	G	H
o_{ij}	H_{jk}	A,	B,	G,	H,	E,	F,	G,	H,
	l_{ij}	1		3		1+(2*y*-1)			

Table 6. A complete instance of the considered problem introduced in Table 2 obtained by exploding the columns of the dataset σ_i on the rows; $y \in \{0,1,2\}$.

The algorithm *LS** is described by the following pseudocode, where the array of triangular relations is y_T.

Algorithm 1. *LS* algorithm* for Hybrid AM-IPS (set the constructive metaheuristic for FJS with sequencing flexibility)

Input: $\{\bigcup_{i=1}^{n}(P_i \backslash \{L_J, o_j \in \sigma_i\}), Y_{hk}\}$

1. Apply the Rule E: $l_O \leftarrow 0$ and $l_{ij} \leftarrow 1$ for each < O, *j*> $\in E$
2. T ← 1 // ternary digit counter
3. **for each** *i* =1 **to** *n* **do** {
4.   **for each** *j* =1 **to** r_i **do**
5.   **if** σ_i (*j*,*j*)≠NULL {
6. $W \leftarrow \{\}$
7.   **for each** *w* =*j*+1 **to** r_i **do**
8.   **if** (σ_i (*w*,*j*)≠NULL {
9.   insert *w* in the set *W*
10.   **if** σ_i (*j*,*w*)=NULL : $l_{iw} \leftarrow \max\{l_{iw}, l_{ij}+1\}$ // (*j*,*w*)∈ *C*, for essential submatrix at the case #1
11.   **else** {
12. $l_{iw} \leftarrow l_{ij}$ // [*j*,*w*]∈ *D*, for essential submatrix at the case #2
13.   // search for a wildcard by pattern of case #4
14.   **for each** *v*=*w* **to** r_i **do** // search for a wildcard node pattern
15.   **if** (*v*,*w*) ≠NULL **and** (*w*,*v*) ≠NULL {
16.   *T* ←T+1
17. $l_{iw} \leftarrow l_{ij}+(2\,y_T-1)$
18. $l_{iv} \leftarrow \max\{l_{iw}, l_{ij}+1\}$
19.   **reset cells** (*w*,*v*) ←NULL; (*v*,*w*) ←NULL; (*j*,*v*) =NULL; (*v*,*j*)=NULL }
20.   }
21. σ_i (*j*,*w*) ←NULL and σ_i (*w*,*j*) ←NULL // reset the row and column corresponding to (*j*,*j*)
22.   }
23.   }
24.   **for each** $w \in W$ **do** {
25.   *j* ←w
26. $W = W \backslash \{w\}$
27.   **repeat** the cycle **from 6 to 21**
28.   }
29. }

Output: *P-digraph* $=\{\bigcup_{i=1}^{n} P_i, Y_{hk}\}$

At the beginning, *LS** selects an operation in the main diagonal whose cell is not empty and set l_j =1 (by rule *e*). Then *LS** scans on the column of the main diagonal to search for a mark. If it is found on a row *w*, i.e. in the cell (*w*,*j*), it removes the mark, looks for the symmetrical cell in the matrix, and set accordingly l_w according to: i) $l_w = l_j+1$, if (*j*,*w*) is empty, or ii) $l_w = l_j$, otherwise. In the case i) and ii) is found, respectively, a conjunctive arc (o_j, o_w) and a disjunctive arc [o_j, o_w]. If the case ii) occurs, *LS** looks for the presence of a triangular relation of precedence according to steps from 14 to 19. Figure 14 shows this mechanism to detect the wildcard node. Finally, if *LS** finds a triangular relation of precedence, it increments the parametrized antecedence of the wildcard of (2*y*-1).

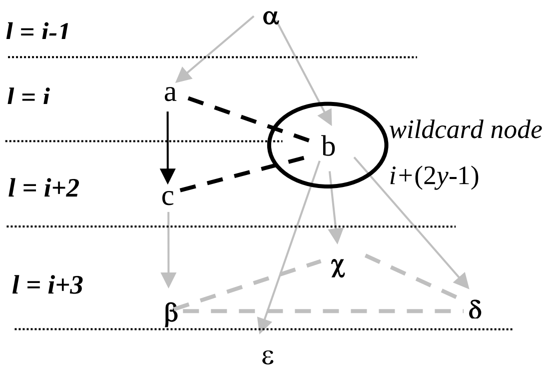


Figure 14. Parametrized antecedence assigned to a wildcard operation which break two consecutive values of *l*.

It can be shown that the sequences obtained by *LS** are feasible, i.e. it belongs to the set Σ_i defined by the expression (4).

Proposition 1. The precedences l_a and l_b, defined in Table 5, are in according to the expression (4).

□ In the column *Alternative routes* of the Table 4, all the possible cases are displayed. For the *C* and *E* types of arc, the proposition follows directly by definition (see Table 1). For the type of arc *D*, i.e. [*a*,*b*]:

$$l_a = L_a + x = l_a, \qquad l_b = L_a + (1 - x) = l_a + 1, \;\; x = 0 \qquad (7)$$

$$l_a = L_a + x = l_a + 1, \quad l_b = L_a + (1 - x) = l_a, \quad x = 1 \qquad (8)$$

obtaining, the first route (a,b) by the expression (7) and the second one (b,a) by the expression (8).

For the triangular relation of type T_1:

$$l_a = L_a, \quad l_b = l_a + (2y - 1) = l_a - 1, \quad l_c = l_a + 2, \quad y = 0 \qquad (9)$$

$$l_a = L_a, \quad l_b = l_a + (2y - 1) = l_a + 1, \quad l_c = l_a + 2, \quad y = 1 \qquad (10)$$

$$l_a = L_a, \quad l_b = l_a + (2y - 1) = l_a + 3, \quad l_c = l_a + 2, \quad y = 2 \qquad (11)$$

obtaining, the first route (b,a,c), the second (a,b,c) and the last (a,c,b) by the expressions, respectively, (9), (10) and (11) ■

We denote with Πi^{2} the set of feasible sequences which includes the parametrizing relations from (7) to (11). It results, $\{o_1,..,[a,b,c],..,o_{l_i} \mid a,b,c \in \Omega_i\} \in \Pi i^2$ for each sequence including the operations a, b and c such that $L_a < L_b$, i.e. $\{o_1,..,(c,a,b),..,o_{l_i}\},\{o_1,..,(a,c,b),..,o_{l_i}\},\{o_1,..,(a,b,c),..,o_{l_i}\} \in \Pi i^2$.

Proposition 2. Πi^2 includes Πi^1.

□ From *Theorem 1* follows that the set Πi^1 includes feasible sequences ordered by their antecetences L (see expression (4)). Therefore, Πi^1 includes exclusively the sequence (a, b, c), ordered by $L_a < L_b$ (i.e. $l_a \leq l_b$) and $L_b < L_c$, or the sequence (c, a, b), ordered by $L_c < L_a < L_b$. It can not include both the sequences otherwise no order there is between L_a and L_c (or L_b and L_c). The earliest constraint implies $\{o_1,..,(c,a,b),..,o_{l_i} \mid l_c < l_a\} \notin \Pi i^1$. The latter constraint implies $\{o_1,..,(a,b,c),..,o_{l_i} \mid l_b < l_c\} \notin \Pi i^1$■

From *Theorem 1* and *Proposition 2* it results:

$$\Pi_{i1} \subset \Pi_{i2} \subset \Pi_i \qquad (12)$$

Therefore, there is a chance that the optimal sequence Σ_i^{opt} can be included in Πi^2 but not in Πi^1. Thus, the proposed algorithm *LS** produces antecedences according to *Proposition 1* (the sequences are included in

Πi^2). Namely, the problem under consideration is more accurately defined with triangular relations of precedence than without of these. In theory, there is no limit to the size of complex relations.
The algorithm *LS** can be used as scheduling algorithm to create a feasible schedule. In the next section is proposed the strategy to select operations which has not been yet inserted in the schedule.

4.4 Scheduling of non-linear routing

As all the operations are related to a proper antecedence, the non-linear routing of the job is defined. When all the binary and ternary digits are selected, the feasible sequence $\Sigma_i \in \Pi i^2$, i=1,..,n, is automatically defined and a constructive metaheuristic can be applied. A constructive metaheuristic is a heuristic where an intelligent strategy which generates step-by-step the schedule s, by the direction of a disjunctive arc h of Y_{hk}. The node reached by h is inserted in s.

The proposed strategy is a population-based metaheuristics where each ant of the colony performs a list scheduling to form an optimal path Σ_i on the *P-digraph* in $O(\sum_{i=1}^{n} r_i)$. The path starts from an empty solution and is in accordance with the precedence constraints and parametrized antecedences set by *LS**. At every step, each ant of the colony connects a node to the acyclic conjunctive graph $\Sigma_{i\ \mathrm{partial}}$ (which represents the partial path) by a feasible move. A feasible move is a disjunctive arc of type Y_{hk} and, eventually, of type D that can be directed in the partial conjunctive graph without creating a cycle according to Σ_i (sorting operations by their antecedence) and the *Theorem 1* (sorting routes of parametrized antecedence). When all the paths are created by the colony, the one with minimum makespan updates the pheromone structure. A pheromone evaporation routine is later performed.

The ant path differs from the list scheduling algorithm over the following two steps: i) the set of candidate nodes, because of the non-linear routing (at the step 1); ii) selection of a feasible move (*Move Selection*, at the step 4). Move selection is based on visibility function and pheromone amount on the arcs of Y_{hk}. The pheromone amount changes according to the evaporation routine and the local gain on the best path.

The proposed ACO is in the Appendix. It differs from the ant path generation described in Rossi (2014), in the selection of the disjunctive arcs of routing which there are not in the cited approach. In addition, we use the modified Critical Ratio in reference (Rossi, Pandolfi and Lanzetta, 2014), for the visibility function.

The following issues characterize the ant path generation.

i. From Rule *E* (Algorithm 1, step 1), the nodes connected to the dummy start node 0 have antecedence equal to l=1. Therefore, the set of Candidate Nodes, AL_0, is initialized with the only dummy start node.

ii. A node a is included in the partial schedule by Move Selection step at a given iteration w, if exists a disjunctive arc $(c, a) \in Y_{hk}$, i.e. feasible move. If exists a node b connected to a by a disjunctive arc $[a, b] \in D$, it results $L_a < L_b$. By *Proposition 1*, the disjunctive arc $[a,b] \in D$ can be directed according to $(a,b) \in C$ without corrupts the partial schedule. The node b is inserted in the *Candidate Nodes* set (AL_{w+1}).

iii. Finally, the conjunctive arc in a triangular relation can be removed (from the job routing sequence) when a node connected to the wildcard operation is executed.

5. Experimental Results

This section shows examples of application and compares the proposed system with benchmarks from the literature.

Figure 15 shows the main steps of the proposed system in the case of two hybrid additive/subtractive parts proposed in Figure 1. The process planning has been shown both in Figure 5 (by AND/OR graph) and in Figure 7 (by *P-digraph*). It has two additive FDM operations, A and B, with two alternatives (additive or hybrid machine). The optimal schedule has been shown in the Gantt chart of Figure 3.

The tolerance specification on the external diameter of the cylindrical surface forces the Accessibility rule (c) denoted by the sequence of tools, nozzle→cutting tool. This requires that subtractive operation C be performed after additive operation A. It results L_a=1. The cells (C_1,A_1) and (C_2,A_2) of the matrix $\sigma_i \times \sigma_i$ include the processing time of the operation C on the hybrid machine η, $t_{C\eta}$=1.

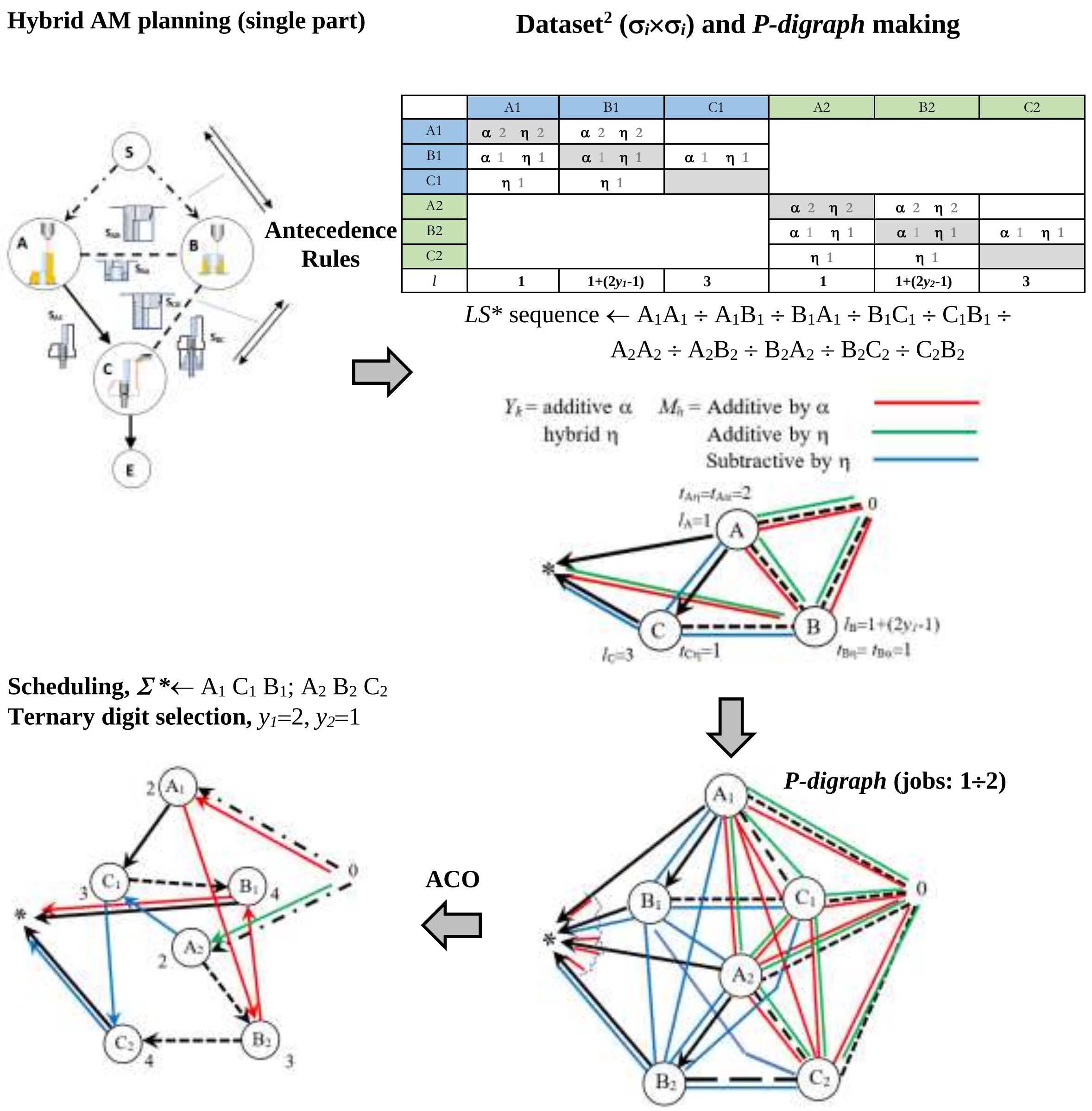


Figure 15 – Example of application of the proposed system to the FJS with sequencing flexibility, $J_m|nonlinear|C_{\max}$ with n=2 and m=2, obtained by the process plan in Figure 1.

Concerning the accessibility by the clamping surface, it can be performed indifferently, first A and then B or vice versa by using the alternative supports of Figure 15 pre-assembled on different tables, one for each part orientation (see also Figure 5). The cells (B$_1$,A$_1$) and (A$_1$,B$_1$) of $\sigma_i \times \sigma_I$ include the processing times $t_A = t_B = 1$, for both the additive and hybrid machines, α and η; accordingly, [A$_2$,B$_2$] of D. It results $L_{A1} = L_{B1} = 1$ and

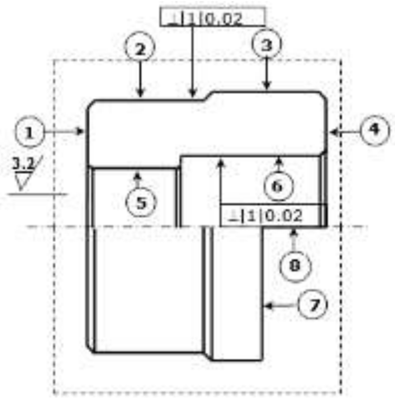
L_{A2}=L_{B2}=1. This forces the cells on the main diagonal (B_i,B_i), i=1,2, to include the modes (for additive and hybrid machine) of the operation B.

In addition to these relations, the *P-digraph* includes the *E-arcs*, which connect the source node to A_1 and B_1, alternatively selected by the binary digit x_1; accordingly, A_2 and B_2, alternatively selected by the binary digit x_2.

*LS** visits the matrix $\sigma_i \times \sigma_i$ and detects the wildcard operation B. It places the parametrized antecedences $(1+(2y_1-1))$ and $(1+(2y_2-1))$, respectively, on the nodes B_1 and B_2. Finally, the Algorithm 2 starts to build constructively the conjunctive graph Σ_i guided by the antecedences L_{ij}.

The following parameters has been used by the ACO in all the experiments: cutting exploration q_0=0.9 with intensification of pheromone amount, *alfa*=1; intensification of visibility function, *beta*= 0.3; evaporation, ρ=0.12. The obtained schedule Σ_i* is an acyclic graph with respect to both machines and jobs routing, whose arc sequences are complete paths from source to destination dummy nodes.

Table 7 shows a more detailed example of application of the proposed system. The part is obtained by external turning, holemaking and open-slot milling. The original items of the dataset shown in the next rows are obtained by CAD software and has been already merged in form features in Table 2; for better reading, we removed the redundancy in the items that had the same mode specification. For the same reason, the two different modes of processing the same operation, machining and turning center, have been merged. Each item $o_{i\,j\,h\,k}$, identified by the column "#", is an operation (identified in the column "Operation Id"). Mode Id keeps track of the original items of the CAD software. The modes 9 and 10 are, respectively, for turning and for machining center. Partial flexibility is considered: not all the operations can be processed by both the machines.



#	Operation			Modes Id	Description	Tool	Fixture	Clamping surface
	Id	Surfaces	Accuracy					
0	O	3	Raw		Root surface		Chuck	Raw part
1	A	1,2	Roughing	1,2	Face-Extern turning	1,2	Chuck	3
2	B	1,2	Finishing	3,4	Face-Extern turning	3,4	Chuck	3
3	C	3,4	Roughing	5,6	Face-Extern turning	1,2	Chuck	2
4	D	5,6	Centering	7,8	Centering on 1	5,6	Chuck	3
				9,10	Centering on 4	5,6	Chuck	2
5	E	5,6	Roughing	11,12	Drilling by 1	7,8	Chuck	3
				13,14	Drilling by 4	7,8	Chuck	2
6	F	6	Finishing	15	Back Boring (TurnCenter)	9	Chuck	3
				16,17	Boring	10,11	Chuck	2
7	G	7,8	Roughing	18	Face milling	12	Chuck	2
				19	Slitting	13	Chuck	2

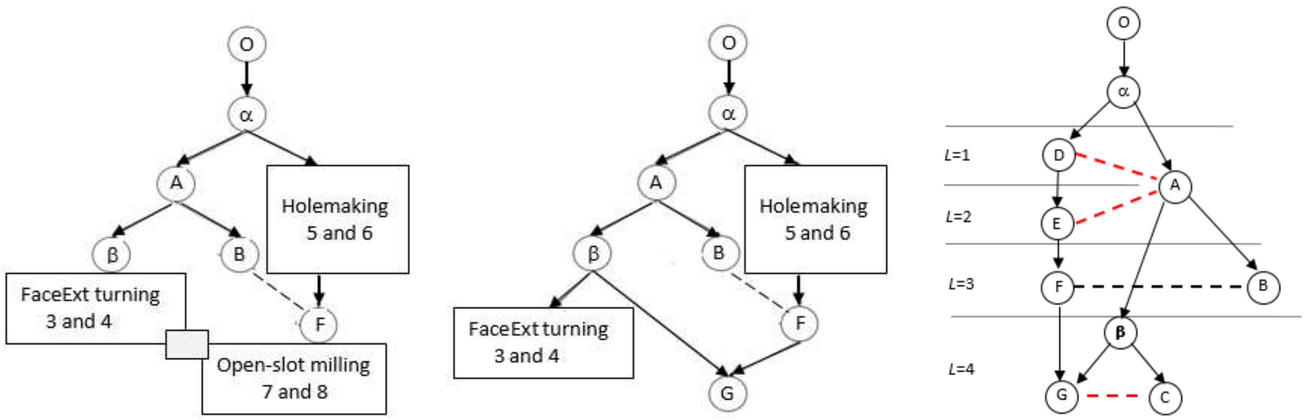
Table 7. Drawing (specifications) for the part displayed and operations.

The algorithm *LS** is driven by two tight restrictions to apply antecedence rules:

i) the presence of tolerance specification of perpendicularity among datum 1 and surfaces 2 and 6;

ii) the removed accessibility to the clamping surface 3 by the open-slot milling operation G.

According to the functional rule (a), there must be a path which starts from the external turning of surface 3 to the following three surfaces: i.1) the external finishing turning of surfaces 2, i.2) the internal finishing turning of surfaces 6, and i.3) the facing of surface 1 (Figure 16.a). The macro-operations which remains are shown in square boxes.



a) b) c)

Figure 16. Making the *P-digraph*: a) reference surfaces for the subjob α and β and restriction i) due to tolerance specification; b) open-slot milling as part of subjob β, subject to the boring F; c) the graph on the right hand as obtained by the algorithm *LS** where the arcs of type *D* are obtained by the sequencing flexibility rule.

Figure 16.a also shows:

- three disjunctive arcs, which connect operations B and F according to the geometric tolerance (Functional rule (a));
- the conjunctive arcs derived from the Accessibility rule (b) to the clamping surface which connect the clamping surface 3 and both operation A and the holemaking;
- the conjunctive arcs derived from the Accessibility rule for quality surface (b) which connect roughing and finishing operations for surfaces 1, 2 and 6.

Accessibility to the clamping surface 3 is corrupted by the slitting operation #7. All the operations which have the clamping surface 3 as the unique clamping surface (i.e. the only operation B), they must be processed before the open-slot milling operation which produces a semi-cylindrical surface. Clamping removing rule (d) prevents the partial surface 3 from being clamped with by a chuck. Figure 16.b shows the resulting effect by the *C*-arc directed from the operation F to the operation G. The conjunctive arcs derived from the Accessibility rule (c), which connect the clamping surface 2 and both G and the external turning of surfaces 3 and 4, are also shown. Also, holemaking operations are solved by the Accessibility rule for the surface accuracy.

Figure 16.c shows the final *P-digraph* produced by the fully-developed process planner by applying the antecedence rules *a÷e* (arcs in black). The sequencing flexibility is shown by red arcs.

The obtained *P-digraph* is the job routing used to formulate four benchmark scheduling problems, P1÷P4, where the number of jobs is 10 (for P1 and P2) or 15 (for P3 and P4). For the experiments are used 2 (P1 and P3) and 4 clusters (P2 and P4) of 2 identical machines each. The benchmark problems, P5÷P8, with the same number of jobs and cluster of parallel machines of P1÷P4, are obtained by the 5 jobs proposed in Rossi, Soldani and Lanzetta (2015).

The lower bound is evaluated by the following expression:

$$LB = \max_{k=1,..,m}\left(\sum_{i=1}^{n}\sum_{j=1}^{r_i}\min_{h\in H_{ik}\wedge h\in Y_{hk}}\left(t_{ijhk} + TT_{kk'}\right)\right) + \mu * \left(\min_{\substack{h,h'=1,..\max\_mode \\ h\neq h'}} SU_{h,h'}\right) \quad (13)$$

where μ is the minimum number of setup necessary to process all the job; *k'* is the machine for the operation o_{j-1}. The transportation time for all the considered problems were calculated with the following formula:

$$TT_{kk'} = \min(|k - k'|, |m + k - k'|) * TT_u \quad (14)$$

considering a ring of *m* machines where the transportation time unit, TT_u, from two contiguous machines is equal to 1.

The problem instances are available in Data in Brief. Each problem fpp_w, *w*=1,..,8, is defined by four files: i) fpp_w-*processtimes*, the processing times matrix (modes x operations); ii) fpp_w-*setuptimes*, the setup times matrix, $S_{h,h'}$ (modes x modes); iii) fpp_w-*routing*, the job routing matrix (operations x operations), where fpp_w-*routing*[O_1,O_2]=1, if (O_2,O_1) belongs to the set of arcs *C*; fpp_w-*routing*[O_1,O_2]=2=fpp_w-*routing*[O_2,O_1], if [O_1,O_2]∈*D*; fpp_w-*routing*[O_1,O_2]=0, otherwise; iv) fpp_w-*machines*, the machine x operation matrix of digits, where fpp_w-*machine*[*O*,*M*]=1 if the machine M can process the operation *O*.

The results of the proposed ACO are shown in Table 8. In all the benchmarks, the proposed system gets a makespan higher that one of *LB*. In fact, *LB* is weak because it summarizes all the minimum setup times of operations that do not match with the minimum processing times of different modes/machines. Increasing the variety of jobs routing from P1÷P4, with a single routing for all the jobs, to P5÷P8, with different routing for all the jobs, the difference between error mean of the sample of 10 runs versus *LB* and the standard deviation increase of about, respectively, 15% and 5%.

Problem	*n*	$\sum_{n} r_i$	*K*	*LB*	*ACO*	ε
P1	10	90	4	79	89	11.2
P2	10	90	8	64	75	14.7
P3	15	135	4	114	128	11.0
P4	15	135	8	91	100	9.0
P5	10	68	4	806	1004	19.7
P6	10	68	8	561	740	29.2
P7	15	102	4	1199	1549	22.6
P8	15	102	8	831	1094	24.0

Table 8. Results on the instances of the general problem. The column *ACO* includes the best makespan found. $\varepsilon = 100 \cdot \frac{Best\ dispatching - ACO}{Best\ dispatching}$

Regarding to the number of jobs, there is no significant difference from P1÷P2 and P5÷P6 (10 jobs) and the respective counterparts, P3÷P4 and P7÷P8 (15 jobs). Regarding to the number of machines, the difference between the error mean and *LB* increases when ranging from P5÷P7 (4 machines) to P6÷P8 (8 machines). This difference does not significantly differ from P1÷P3 (4 machines) to P2÷P4 (8 machines) where the problem instances include only one type of job routing.

The proposed system is affected by the variety of job routings: the mean of the relative errors suffers a sharp increase from P1÷P4 to P5÷P8, where a few arcs of type *D* and some triangular relations of type T_1 and T_2 are introduced. Besides, a high job variety makes less robust the scheduling algorithm.

Table 8 shows a deviation from the mean of the relative errors (16% vs 19%) among the problem instances with odd and even identification numbers, where these last are achieved by increasing the disjunctive arcs (4 vs 8 machines). Therefore, a minor affection of the results concerns with the complexity of the precedence graph.

The performance of the proposed system is also compared with approaches available in the literature similar to the proposed one:

1. $FJS||C_{max}$ proposed by Kacem et al. (2002) (problems from P9 to P12);
2. $FJS|nonlinear|C_{max}$ proposed by Li et al. (2010) (problem P13);
3. $FJS|nonlinear|C_{max}$ three new problems, P14, P15 and P16, obtained by multiplying the number of jobs of P13 respectively by 5, 10 and 20;
4. $FJS|nonlinear|C_{max}$ proposed by Moon et al. (2008) (problem P17);
5. $FJS|nonlinear|C_{max}$ proposed by Amin-Naseri and Afshari (2012) (problems P18 to P20, obtained by multiplying the number of jobs of P17 respectively by 5, 10 and 20);
6. $FJS|nonlinear|C_{max}$ proposed by Lee et al. (2002) (problem P21);
7. $FJS|nonlinear(D)|C_{max}$ proposed by Amin-Naseri and Afshari (2012) (problems P22 to P24, obtained by multiplying the number of jobs of P21 respectively by 5, 10 and 20);
8. $FJS|nonlinear(D), overlapping|C_{max}$ proposed by Rossi, Soldani and Lanzetta (2015) (problems P25 and P26).

The problem instances included in case 1 have total routing and operation flexibility (total flexibility), i.e. a given operation can be processed on one machine selected among all the machines. The problem instances included in the cases 2, 3 and 4 are designed for AND/OR digraphs while those included in the case 5 are based on the relaxed problem formulation (which allows operations overlapping).

For this experiment, the stop condition of $5 \cdot 10^4$ epochs without improvement of the best solution is adopted and the ACO has been run 5 times for all the problem instances. The initial pheromone amount on the arcs of Y_{hk} is $\tau_0=(1/n*m*LB)$. A higher stop condition than those in Table 9 produce the same output for all the executions of the individual benchmark problem. The proposed system optimally solves 12 of the 15 instance problems tested and differs by less than 1.2% in the case it provides the worst performance. The proposed system improves also the results of P26 achieved by a problem formulation, which allows to overlap operations connected by *D*-arcs (see Section 3) by about 11% of the sum of the job completion times.

Problem	***n***	$\sum_n r_i$	***K***	**Exact or Best Known**	**This work**
P9	4	12	5	16	16
P10	10	29	7	15	15
P11	10	30	10	7	7
P12	15	56	10	23	23
P13	6	18	5	27	27
P14	30		5	-	107
P15	60		5	-	215
P16	120		5	-	433
P17	5	13	5	14	14
P18	25	65	5	62	62
P19	50	130	5	124	124
P20	100	260	5	247	249
P21	8	20	6	23	23
P22	40	100	6	105	105
P23	80	200	6	208	209
P24	160	400	6	416	420
P25	5	34	5	546	533
P26	5	34	10	397	392

Table 9. Results on benchmarks from the literature.

6. Conclusions and future work

This work has approached the new problem of optimizing process planning and scheduling where both additive and subtractive operations (including advanced hybrid machines) are present.

Literature review has shown the inadequacy of the operation description available for planning.

The proposed system models and plans hybrid additive/subtractive operations on a batch of different products also considering job shop scheduling.

This model introduces, manages and maximizes all the types of flexibility, affecting a variety of job routings. As the results depend on the complexity of the disjunctive precedence graph, the non-redundant proposed *P-digraph* model strongly reduces the number of nodes and their connections with respect to the AND/OR counterpart. Therefore, the proposed algorithm offers a sharp performance improvement, as the complexity of the precedence graph of antecedences grows, by implicitly visits of the paths in the fully-developed process plan.

Several instances of the problem proposed by different authors, including partial and total flexibility with directed routing of jobs and non-linear job routing with overlapping of operations on different machines, have been effectively dealt with by the ACO implemented.

Current limitations at technical and global levels can be addressed by future research; in particular, operation flexibility is derived from FDM and DED, the proposed model is open to include additional variables and attributes coming from different process (e.g. supports can be created with the part); regarding the metaheuristic algorithm, graph ternary variables can be included in the pheromone structures; also alternative constructive metaheuristics should be explored and compared to improve the system performance (e.g. the computing speed by varying the size of instances); other objective functions can be tested (e.g. weighted tardiness, to consider penalties or scheduling adherence, for just in time).

The implementation of the proposed integrated process planning and scheduling system in industry seems viable today and can provide the best solution among the available options. The generation of alternative options and their objective quantitative evaluation based on different criteria (cost, quality, time etc.), not considered in this work, represent a major exploration area, particularly for AM processes. A concrete output will be a new generation (standard) 3D model file format to be shared among different CAD/CAM systems for Hybrid AM.

References

Ahsan, A. N., Habib, M. A., & Khoda, B. (2015). Resource based process planning for additive manufacturing. *Computer-Aided Design*, *69*, 112-125.

Aloulou, M.A., & Artigues, C. (2010). Flexible solutions in disjunctive scheduling: General formulation and study of the flow-shop case. *Computers & Operations Research*, *37*, 890-898.

Amin-Naseri, M. R., & Afshari, A. J. (2012). A hybrid genetic algorithm for integrated process planning and scheduling problem with precedence constraints. *The International Journal of Advanced Manufacturing Technology*, *59*, 273-287.

Bensmaine, A., Dahane, M., & Benyoucef, L. (2014). A new heuristic for integrated process planning and scheduling in reconfigurable manufacturing systems. *International Journal of Production Research*, *52*, 3583-3594.

Bentaha, M. L., Battaïa, O., & Dolgui, A. (2014). A sample average approximation method for disassembly line balancing problem under uncertainty. *Computers & Operations Research*, *51*, 111-122.

Birgin, E. G., Ferreira, J. E., & Ronconi, D. P. (2015). List scheduling and beam search methods for the flexible job shop scheduling problem with sequencing flexibility. *European Journal of Operational Research*, *247*, 421-440. http://dx.doi.org/10.1016/j.ejor.2015.06.023.

Blazewicz, J., & Kobler, D. (2002). Review of properties of different precedence graphs for scheduling problems. *European Journal of Operational Research*, *142*, 435-443.

Błażewicz, J., Ecker, K. H., Pesch, E., Schmidt, G., & Weglarz, J. (2013). *Scheduling computer and manufacturing processes*. Springer Science & Business Media.

Blum, C., & Sampels, M. (2004). An ant colony optimization algorithm for shop scheduling problems. Journal of Mathematical Modelling and Algorithms, *3*, 285-308.

Chen, L., Xu, K., & Tang, K. (2018). Optimized sequence planning for multi-axis hybrid machining of complex geometries. *Computers & Graphics*, *70*, 176-187.

Chen, L., Xu, K., & Tang, K. (2018). Optimized sequence planning for multi-axis hybrid machining of complex geometries. *Computers & Graphics*, *70*, 176-187.

Doh, H. H., Yu, J. M., Kim, J. S., Lee, D. H., & Nam, S. H. (2013). A priority scheduling approach for flexible job shops with multiple process plans. *International Journal of Production Research*, *51*, 3748-3764.

Du, N., Hu, H., & Zhou, M. (2019). A survey on robust deadlock control policies for automated manufacturing systems with unreliable resources. *IEEE Transactions on Automation Science and Engineering*. https://ieeexplore.ieee.org/abstract/document/8827303/

Flynn, J. M., Shokrani, A., Newman, S. T., & Dhokia, V. (2016). Hybrid additive and subtractive machine tools–Research and industrial developments. *International Journal of Machine Tools and Manufacture*, *101*, 79-101.

Frank, M. C., Harrysson, O., Wysk, R. A., Chen, N., Srinivasan, H., Hou, G., & Keough, C. (2017). Direct additive subtractive hybrid manufacturing (DASH)—an out of envelope method. In *28th Annual International Solid Freeform Fabrication Symposium*.

Gonzalez, T.; Sahni, S. (1976). Open shop scheduling to minimize finish time. *Journal of the ACM*, **23** (4): 665–679.

Giffler, B., & Thompson, G. L. (1960). Algorithms for solving production-scheduling problems. *Operations research*, *8*, 487-503.

Hansel, A., Mori, M., Fujishima, M., Oda, Y., Hyatt, G., Lavernia, E., & Delplanque, J. P. (2016). Study on consistently optimum deposition conditions of typical metal material using additive/subtractive hybrid machine tool. *Procedia CIRP*, *46*, 579-582.

Hao, J., Chen, X., Liu, H., & Ye, S. (2018). A novel process planning algorithm for additive and subtractive manufacturing based on skeleton tree matching. *Rapid Prototyping Journal*, *24*(2), 441-462.

Hofmann, D. C., Kolodziejska, J., Roberts, S., Otis, R., Dillon, R. P., Suh, J. O., ... & Borgonia, J. P. (2014). Compositionally graded metals: A new frontier of additive manufacturing. *Journal of Materials Research*, *29*(17), 1899-1910.

Kacem, I., Hammadi, S., & Borne, P. (2002). Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic. *Mathematics and computers in simulation*, *60*(3-5), 245-276.

Kim, H. H., Kim, D. G., Choi, J. Y., & Park, S. C. (2017). Tire mixing process scheduling using particle swarm optimization. *Computers & Industrial Engineering*, *110*, 333-343.

Kumar, S. L. (2017). State of the art-intense review on artificial intelligence systems application in process planning and manufacturing. *Engineering Applications of Artificial Intelligence*, *65*, 294-329..

Lee, Y. H., Jeong, C. S., & Moon, C. (2002). Advanced planning and scheduling with outsourcing in manufacturing supply chain. *Computers & Industrial Engineering*, *43*, 351-374.

Leung, C. W., Wong, T. N., Mak, K. L., & Fung, R. Y. (2010). Integrated process planning and scheduling by an agent-based ant colony optimization. *Computers & Industrial Engineering*, *59*, 166-180.

Li, X., Gao, L., Shao, X., Zhang, C., & Wang, C. (2010). Mathematical modeling and evolutionary algorithm-based approach for integrated process planning and scheduling. *Computers & Operations Research*, *37*, 656-667.

Li, X., Peng, Z., Du, B., Guo, J., Xu, W., & Zhuang, K. (2017). Hybrid artificial bee colony algorithm with a rescheduling strategy for solving flexible job shop scheduling problems. *Computers & Industrial Engineering*, *113*, 10-26.

Maccarthy, B. L., & Liu, J. (1993). Addressing the gap in scheduling research: a review of optimization and heuristic methods in production scheduling. *The International Journal of Production Research*, *31*(1), 59-79.

Moon, C., Lee, Y. H., Jeong, C. S., & Yun, Y. (2008). Integrated process planning and scheduling in a supply chain. *Computers & Industrial Engineering*, *54*, 1048-1061.

Newman, S. T., Zhu, Z., Dhokia, V., & Shokrani, A. (2015). Process planning for additive and subtractive manufacturing technologies. *CIRP Annals*, *64*(1), 467-470.

Oh, Y., Zhou, C., & Behdad, S. (2018). Part decomposition and assembly-based (Re) design for additive manufacturing: A review. *Additive Manufacturing*, *22*, 230-242.

Pande, S., & Kumar, S. (2008). A generative process planning system for parts produced by rapid prototyping. *International Journal of Production Research*, *46*(22), 6431-6460.

Paris, H., & Mandil, G. (2017). Process planning for combined additive and subtractive manufacturing technologies in a remanufacturing context. *Journal of Manufacturing Systems*, *44*, 243-254.

Ren, L., Sparks, T., Ruan, J., & Liou, F. (2008). Process planning strategies for solid freeform fabrication of metal parts. *Journal of Manufacturing Systems*, *27*(4), 158-165.

Rossi, A. (2014). Flexible job shop scheduling with sequence-dependent setup and transportation times by ant colony with reinforced pheromone relationships. *International Journal of Production Economics*, *153*, 253-267.

Rossi, A., Pandolfi, A., & Lanzetta, M. (2014). Dynamic set-up rules for hybrid flow shop scheduling with parallel batching machines. *International Journal of Production Research*, *52*(13), 3842-3857.

Rossi, A., Soldani, S., & Lanzetta, M. (2015). Hybrid stage shop scheduling. Expert Systems with Applications, *42*, 4105-4119. http://dx.doi.org/10.1016/j.eswa.2014.12.050.

Shao, X., Li, X., Gao, L., & Zhang, C. (2009). Integration of process planning and scheduling—a modified genetic algorithm-based approach. *Computers & Operations Research*, *36*, 2082-2096.

Sinnen, O. (2014). Reducing the solution space of optimal task scheduling. *Computers & Operations Research*, *43*, 201-214.

Thompson, M. K., Moroni, G., Vaneker, T., Fadel, G., Campbell, R. I., Gibson, I., ... & Martina, F. (2016). Design for Additive Manufacturing: Trends, opportunities, considerations, and constraints. *CIRP annals*, *65*(2), 737-760.

Wang, J., Fan, X., Zhang, C., & Wan, S. (2014). A graph-based ant colony optimization approach for integrated process planning and scheduling. *Chinese Journal of Chemical Engineering*, *22*, 748-753.

Wang, L., Cai, N., Feng, H. Y., & Liu, Z. (2006). Enriched machining feature-based reasoning for generic machining process sequencing. *International Journal of Production Research*, *44*(8), 1479-1501.

Yang, Y., & Hu, H. (2018). A distributed control approach to automated manufacturing systems with complex routes and operations using petri nets. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*. https://ieeexplore.ieee.org/abstract/document/8574045/

Zhang, L., & Wong, T. N. (2016). Solving integrated process planning and scheduling problem with constructive meta-heuristics. *Information Sciences*, *340*, 1-16.

Zhang, S., & Wang, S. (2018). Flexible assembly job-shop scheduling with sequence-dependent setup times and part sharing in a dynamic environment: Constraint Programming Model, Mixed-Integer Programming Model, and Dispatching Rules. *IEEE Transactions on Engineering Management*, *65*(3), 487-504.

Zhang, Y., Bernard, A., Harik, R., & Karunakaran, K. P. (2017). Build orientation optimization for multi-part production in additive manufacturing. *Journal of Intelligent Manufacturing*, *28*(6), 1393-1407.

Appendix

The following is a pseudo-code description of the proposed algorithm that generates a feasible solution in O(n) with the completion times placed on the related nodes and the weights on the conjunctive arcs.

Algorithm 2. *Ant path generation for Hybrid AM-IPS (FJS with sequencing flexibility)*

Input: an *P-digraph* = (N, C, D, E, Y_k, $W_{i,k}$) with antecedences configured by LS^*

$O \leftarrow \{O_{i\,k\,l} \mid i=1,..,n,\ k=1,..,m,\ l=1,..,r_i\}$

Inizialization of Candidate Nodes: $AL_0 \leftarrow \{O_{i\,k\,l} \in O \mid l_{O_{i,k,l}} = 0\}$ // it results $AL_0 \leftarrow \{0\}$: set formed by the dummy start node

$r \leftarrow l_O + 1$ // the current value of the minimum antecedence in O

for each $w = 1$ **to** $\Sigma_{i=1,..,n}\, r_i$ **do**

1. *Initialization of Candidate Nodes*: mark as a *feasible node* each node $O_{i\,k\,l}$ in a conjunctive arc ($O_{i\,k'\,r}$, $O_{i\,k\,l}$) of C with $l=r+1$ or to a disjunctive arc [$O_{i\,k'\,r}$, $O_{i\,k\,l}$] of D with $l=r$ or $l=r\,(2y_T+1)$:

$AL_w \leftarrow O_{i\,k\,l}$

2. *Inizialization of Feasible Moves*: mark as a *feasible move* each disjunctive arc ($O_{i'\,k\,r}$, $O_{i\,k\,l}$) of Y_k where $O_{i\,k\,l} \in AL_w$ is a feasible node and $O_{i'\,k\,r}$ is the last operation of the sequence of machine k (it creates the possibility for the candidate operation to become the new last operation of Y_k and in $\Sigma_{i\ \text{partial}}$);

3. *Move Selection*: select a feasible move ($O_{i'\,k\,r}$, $O_{i\,k\,l}$) of Y_k according the transition rule by directing the related disjunctive arc; if exists the arc [$O_{i\,k'\,r}$, $O_{i\,k\,l}$]$\in D$, direct this arc;

4. *D-arcs Removal*: remove all the remaining arcs ($O_{i\,*\,*}$, $O_{i\,k'\,r}$) of C and [$O_{i\,*\,*}$, $O_{i\,k'\,r}$] of D connected to $O_{i\,k'\,r}$; if $O_{i\,k'\,r}$ is a wildcard (i.e. has the antecedence of type $r=l\,(2y_T+1)$), remove the remaining arc ($O_{i\,*\,*}$, $O_{i\,k'\,r}$) of C;

5. Y_k *-arcs Removal*: remove all the remaining disjunctive arcs of Y_k connected to $O_{i\,k'\,r}$ (i.e. no other sequence can include the operation);

6. *Computing path length*: the length of the arc ($O_{i\,'j\,r'}$, $O_{i\,j\,r}$), directed at step 4, is the processing times $t_{i\,k}$ of $O_{i\,j\,r}$ is evaluated by:

$$t(O_{i\,k\,l}) = st(O_{i\,k\,l}) + t_{i\,k} = \max\{t(O_{i\,k'\,r}),\ t(O_{i'\,k\,r})\} + t_{i\,k}$$

7. *Updating Structures*: update AL_w by removing operation $O_{i\,k'\,r}$: $AL_w \leftarrow AL_w \setminus O_{i\,k'\,r}$;

8. *Antecedence value update*: scan the set AL_w for the minimum l, $O_{i\,k\,l} \in AL_w$; let l_{min}= min $\leftarrow \{l \mid O_{i\,k\,l} \in O\}$:

$r \leftarrow l_{min}$

end for

9. *Directing the remaining disjunctive arcs*: the arcs are connected to the dummy operation *;

Output: the schedule *S* with the completion times of the operations

These property results from the schedule S:

a) the achieved graph includes all the nodes: the main loop is performed |O| times and initially the candidate list includes all the nodes; at each iteration one and only one node is removed from candidate list (step 7);

b) the achieved graph is conjunctive: for each iteration, the selected feasible move is a conjunctive arc which ends at the node removed from candidate list (step 3); all the disjunctive arcs which starts from the first node of the conjunctive arc are removed (step 4 and 5);

c) the conjunctive graph is acyclic: each feasible move ends to a node which is in the candidate list, i. e. is a not scheduled operation (steps 1 and 2).