



Research  
Intelligent Manufacturing—Article

# A Novel MILP Model Based on the Topology of a Network Graph for Process Planning in an Intelligent Manufacturing System

Qihao Liu, Xinyu Li\*, Liang Gao

State Key Laboratory of Digital Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China



## ARTICLE INFO

### Article history:

Received 29 October 2019  
Revised 21 March 2021  
Accepted 27 April 2021  
Available online 5 May 2021

### Keywords:

Process planning  
Network  
Mixed-integer linear programming  
CPLEX

## ABSTRACT

Intelligent process planning (PP) is one of the most important components in an intelligent manufacturing system and acts as a bridge between product designing and practical manufacturing. PP is a nondeterministic polynomial-time (NP)-hard problem and, as existing mathematical models are not formulated in linear forms, they cannot be solved well to achieve exact solutions for PP problems. This paper proposes a novel mixed-integer linear programming (MILP) mathematical model by considering the network topology structure and the OR nodes that represent a type of OR logic inside the network. Precedence relationships between operations are discussed by raising three types of precedence relationship matrices. Furthermore, the proposed model can be programmed in commonly-used mathematical programming solvers, such as CPLEX, Gurobi, and so forth, to search for optimal solutions for most open problems. To verify the effectiveness and generality of the proposed model, five groups of numerical experiments are conducted on well-known benchmarks. The results show that the proposed model can solve PP problems effectively and can obtain better solutions than those obtained by the state-of-the-art algorithms.

© 2021 THE AUTHORS. Published by Elsevier LTD on behalf of Chinese Academy of Engineering and Higher Education Press Limited Company. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

## 1. Introduction

Intelligent manufacturing involves intelligent manufacturing technology and intelligent manufacturing systems [1–3]. Intelligent process planning (PP) is one of the most important intelligent manufacturing technologies and plays a crucial role in intelligent manufacturing systems [4,5]. PP can effectively shorten production cycle and improve product quality, and further, reduce the consumption for resources and energy [6]. Hence, PP has gained a considerable amount of attention in industrial applications. Nevertheless, due to the extensive internal flexibility within PP, it is very difficult to solve for an optimal process plan in practical production scenarios [7].

Solving a traditional PP problem usually involves three steps: process selection, resource allocation, and operation sequencing [5]. The first step is to fix the process selection, afterwards, the length of a process route can be determined because the number of operations varies from different process approaches. Next, the allocation of resources, including machines and tools, should be

configured for all the operations. Sequencing the operations is the last and most critical step for obtaining a feasible process route that obeys the precedence constraints [8]. For example, milling and grinding must be arranged before fine milling and fine grinding operations, and tapping for a hole thread must be carried out after the operation of drilling a hole [9].

PP problems have been proved to be nondeterministic polynomial-time (NP)-hard [6,10] meaning that it is difficult to solve by only relying on the traditional gradient descent methods, graph theory methods, or simulation-based methods. Therefore, most researchers attempt to introduce meta-heuristics [11] to study PP problems. The main research approaches for PP problems include genetic algorithm (GA) [7,12], tabu search (TS) [13], particle swarm optimization (PSO) [8], ant colony optimization (ACO) [14], and honey-bees mating optimization (HBMO) [10], which aim to find high-quality solutions within less computational cost, and attach more importance to the efficiency, rather than the optimality.

In this background, optimal solutions for many open PP problems [8,15] have not been found yet. One important reason is that recent studies have paid more attention to improving the performance of intelligent algorithms than to the modification of current

\* Corresponding author.

E-mail address: [lixinyu@mail.hust.edu.cn](mailto:lixinyu@mail.hust.edu.cn) (X. Li).

mathematical models, especially mixed-integer linear programming (MILP) mathematical models [14]. An effective MILP model can be solved by a mathematical programming solver, such as CPLEX, Gurobi, and so forth, and the optimal solution of a PP problem may be stably obtained under certain conditions. However, the solving effect of an MILP model is highly related to the problem scale. Once the scale becomes larger, the solving effect of the model deteriorates rapidly, and the model may even fail to find a feasible solution within a long computation time [16]. Therefore, current research work for PP does not focus on MILP models. In fact, the research status of MILP models is in a state of “compromise” because almost all reported PP models are feature-based [8,14,17]. These models convert an original network graph into a tabular form to simplify process representations. This kind of conversion, to some extent, changes the original process information of the network. (Discussions on this situation are presented in Section 3.3.) In addition, although there are other modeling methods that can avoid the abovementioned situation, they require complicated preprocessing, such as the generation of combinations of all possible processing operations, in order to eliminate and substitute OR nodes that represent a type of OR logic inside the network [18]. This preprocessing procedure is complicated, in particular, the presence of too many OR nodes or a complex topological structure will greatly increase the complexity and computational time of a PP problem.

In order to fill in the gap of recently reported research work, a novel MILP model is proposed based on the PP network topological structure. The main contributions of this paper are as follows:

(1) This paper proposes a new OR-node-based MILP model for PP problems with no need for any feature conversion or other preprocessing.

(2) The precedence relationships of the operations are discussed in detail, and three types of precedence relationship matrices are presented to illustrate the precedence relationship constraints.

(3) Coded in the general algebraic modeling system (GAMS)/CPLEX solver, this MILP model successfully finds new optimal solutions for open problems in the literature [8,15,16].

The rest of this paper is organized as follows. Section 2 introduces the related work, while Section 3 proposes the model and discusses the related analysis mentioned above in detail. Section 4 presents several groups of comparative experiments to verify the advantage of the proposed model. Section 5 provides conclusions and outlines some future work.

## 2. Related work

The related work on PP can be divided into two categories: algorithms and mathematical models. Xu et al. [6] and Leo Kumar [4] have both provided good reviews of PP problems. Intelligent algorithms such as GA, simulated annealing (SA) algorithms, TS algorithms, and PSO algorithms have shown sufficient advantages and have been widely applied in PP problems [6]. To solve PP problems with complex prismatic parts, Li et al. [19] proposed a good hybrid algorithm comprising a GA and a SA algorithm. The local search capability of the hybrid algorithm was enhanced by a strategy of searching and selecting solutions based on the Hamming distance between alternative routes. Hua et al. [20] proposed a synthesis algorithm based on a GA to search for the global or near-global optimal solution in PP problems. Li et al. [15] proposed an effective genetic programming (GP) algorithm, which performed genetic operations on OR-node branches in the network graph. Taking into account manufacturing resources such as tool selection and feed direction, Shin et al. [21] proposed a symbiotic evolutionary algorithm to optimize three objectives, including balancing the machine workload, minimizing part movements, and minimizing tool changes. Wang et al. [22] put forward a PSO

combining two local search strategies to solve the PP problem. Subsequently, Li et al. [8] presented a modified PSO to solve PP problems considering the transmission time between machines. Liu et al. [14] combined ACO with the constraint matrix and the state matrix of the problem, and applied it to solve the PP problem of two prismatic parts.

Although some achievements have been made in applying intelligent algorithms to solve PP problems, the solution quality can be further improved in most of the existing PP instances, since meta-heuristic algorithms cannot guarantee the optimality of the solutions. Besides, as a method of describing problems, a mathematical model can help researchers to understand and comprehend problems more deeply and thoroughly [23]. Therefore, research on PP mathematical models is very meaningful. Floudas and Lin [24] analyzed several mixed-integer programming (MIP) models of PP problems on the methods of time representation, and then proposed several effective optimization approaches to improve the computational efficiency of the model. In view of the complementarity of PP and scheduling, Li et al. [25] established a mathematical model to integrate a PP problem and shop scheduling problem. Xia et al. [26] proposed a feature-based mathematical model for a reconfigurable PP problem. Similarly, based on features, Jin and Zhang [16] established MILP models for PP considering the transmission time between machines.

To the best of our knowledge, mathematical models for PP problems are always feature-based [8,14,17]. Although this kind of modeling method can describe most kinds of jobs, it is still inevitable that some direct precedence constraints must be added, which do not exist in the original network [16]. Therefore, the network graph method is more capable of describing different manufacturing flexibilities than a feature-based method. This paper proposes a novel MILP model directly based on the topology of the network graph; it can describe all types of manufacturing flexibilities of PP without adding or omitting any constraints. By solving the proposed model, new optimal solutions of some famous benchmarks in previous publications are successfully obtained.

## 3. The proposed MILP model for PP

### 3.1. Problem description

The types of flexibilities in PP include process flexibility, machine selection flexibility, and operation sequencing flexibility. There are several approaches to describe PP problems, such as Petri nets [27], feature tables [28], AND/OR graphs, and networks [29,30]. In Fig. 1, the manufacturing flexibilities are represented in the form of a network graph. This network graph is composed of five types of nodes: The starting node, which is virtual, represents the start of a part's production; the ending node, which is also virtual, indicates the end of a part's production; intermediate nodes represent operations; and the OR node, combined with the fifth type of node, the JOIN node, represents the process flexibilities [15]. An intermediate node contains three pieces of information: the operation number in the solid circle, the alternative machine number in { }, and the corresponding processing time in [ ]. For example, the intermediate node 6 indicates that operation 6 can be processed on any machine out of the three alternative machines 3, 7, and 13, and the required processing time is 44, 48, and 49, respectively. The time unit in the paper is omitted just like original data. Arrows connecting pairs of nodes in the network indicate precedence relationship constraints between operations [21]. For example, the arrow between operations 2 and 3 declares that operation 2 must be processed before operation 3. There are no fixed precedence constraints between operations not connected by an arrow. Only one link connected to the OR node will be

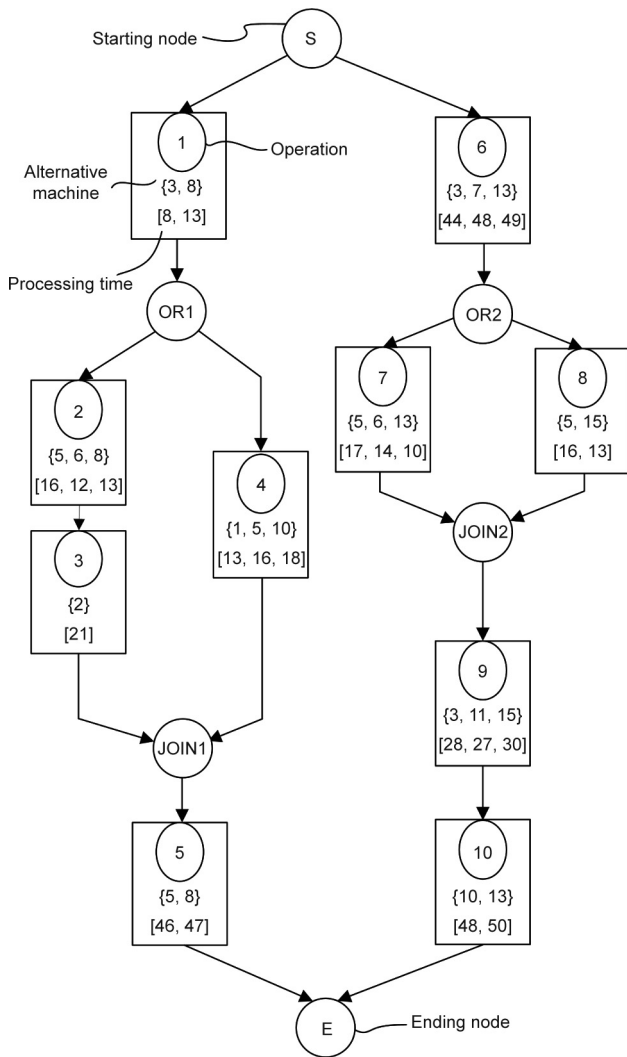


Fig. 1. A flexible process plan network. OR1, OR2, JOIN1, and JOIN2 represent OR node 1, OR node 2, JOIN node 1, and JOIN node 2, respectively.

selected. The operations contained in the link with other operations compose a feasible operation combination for the part [29]. Take Fig. 1 as an example: If operations 2 → 3 are selected between OR node 1 and JOIN node 1, and if operation 7 is selected between OR node 2 and JOIN node 2, then one of the complete feasible process routes is 1 → 2 → 3 → 5 → 6 → 7 → 9 → 10.

3.2. Precedence relationship between operations

According to the definitions of binary variables, there are three modeling methods for PP problems [31]:

$$\theta_{jt} = \begin{cases} 1, & \text{if operation } j \text{ is processed at period } t \\ 0, & \text{otherwise} \end{cases}$$

$$\rho_{jt} = \begin{cases} 1, & \text{if operation } j \text{ is the } t\text{th one of the sequence to be processed} \\ 0, & \text{otherwise} \end{cases}$$

$$q_{jj'} = \begin{cases} 1, & \text{if operation } j \text{ is processed before operation } j' \\ 0, & \text{otherwise} \end{cases}$$

For the large amount of variables and constraints, the binary variable  $\theta_{jt}$  based on time periods is rarely mentioned in PP problem modeling. In the existing literature, the definition  $\rho_{jt}$  describes the operation sequence by the position of operation  $j$  [16,32,33],

which is widely used in the current literature [24,34,35]. In this paper, the proposed MILP model for PP is established based on the third method,  $q_{jj'}$ , for the first time.

As shown in Fig. 2, a feasible operation sequence of this part is 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 → 9. Operation 1 is in front of all the other eight operations; thus, according to the definition,  $q_{1j'} = 1$ , where  $j' = 2, \dots, 9$ . Operation 2 is processed before all the other seven operations, so  $q_{2j'} = 1$ , where  $j' = 3, \dots, 9$ . The number of “1”s in the matrix  $Q = [q_{jj'}]$  can be obtained as  $(n - 1)n/2$ , where  $n$  stands for the total number of operations, which is also the length of the operation sequence. Fig. 3 shows the precedence relationship transformation from a sequence to a  $Q$  matrix.

For a sequence that has been ordered, every two operations have a sequential relationship. Therefore, the priority relation can be represented by an  $n \times n$  matrix. The  $Q$  matrix contains all the precedence relationships of an operation sequence. The  $Q$  matrix can be regarded as a completely expressing matrix (CEM) for the precedence relationships of an operation sequence. Through the observation and analysis, the characteristic constraints of CEM  $Q$  can be concluded as follows:

(1) The diagonal elements of CEM  $Q$  are equal to 0:

$$q_{jj} = 0, \forall j \tag{1}$$

(2) The sum of the two elements symmetrical about the diagonal is equal to 1:

$$q_{jj'} + q_{j'j} = 1, \forall j \neq j' \tag{2}$$

(3) The sum of the elements in any two different columns is not equal:

$$\sum_j q_{jj'} \neq \sum_j q_{jj''}, \forall j' \neq j'' \tag{3}$$

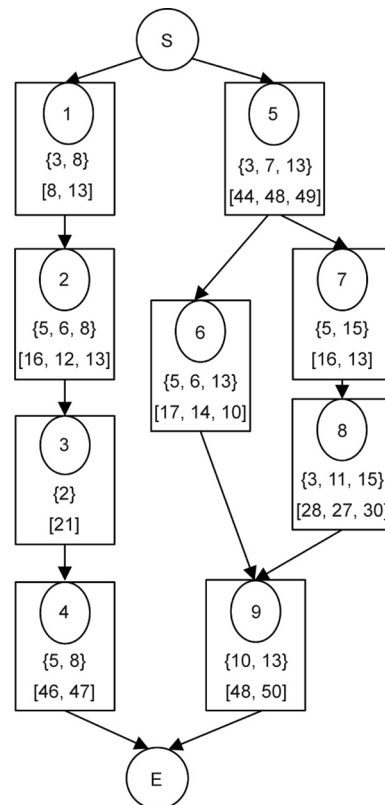


Fig. 2. A process plan network.

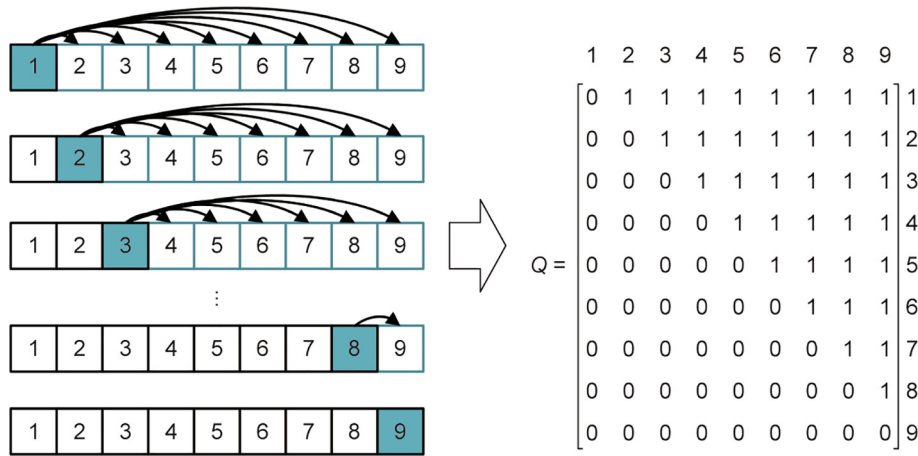


Fig. 3. The transformation of the precedence relationship.

Eq. (1) indicates that precedence relationships only exist between different operations. In Eq. (2),  $q_{ij}=1$  makes  $q_{ji}=0$ , whereas  $q_{ij}=0$  makes  $q_{ji}=1$ , because there is only one precedence relationship between the two operations. Eq. (3) is established because the sum of the column elements corresponds to the position that is unique in the operation sequence. The  $Q$  matrix contains all the precedence relationships between operations. According to the corresponding  $Q$  matrix, it can be quickly and easily determined whether a sequence satisfies the precedence constraints in a network.

The notation  $s_{ij}$  is defined to represent the precedence constraints in Fig. 2:

$$s_{ij} = \begin{cases} 1, & \text{if operation } j \text{ should be processed before } i \text{ according to the network} \\ 0, & \text{otherwise} \end{cases}$$

The corresponding constraint matrix  $S=[s_{ij}]$  is shown in Fig. 4, where each “1” in the  $S$  matrix corresponds to an arrow in the network, representing a precedence constraint. If a sequence satisfies all the precedence constraints, its CEM  $Q$  is supposed to contain all the precedence values shown in matrix  $S$ , which can be formulated as follows:

$$q_{ij} \geq s_{ij}, \forall j, j' \quad (4)$$

Matrix  $S$  can be generated from the network graph, but the matrix  $Q$  is unknown because the operation sequence has not been determined yet. Therefore, Eq. (4) can be regarded as the constraint.

The number of “1”s in CEM  $Q$  is  $(n-1)n/2$ . However, the least number of “1”s required to determine an operation sequence in a

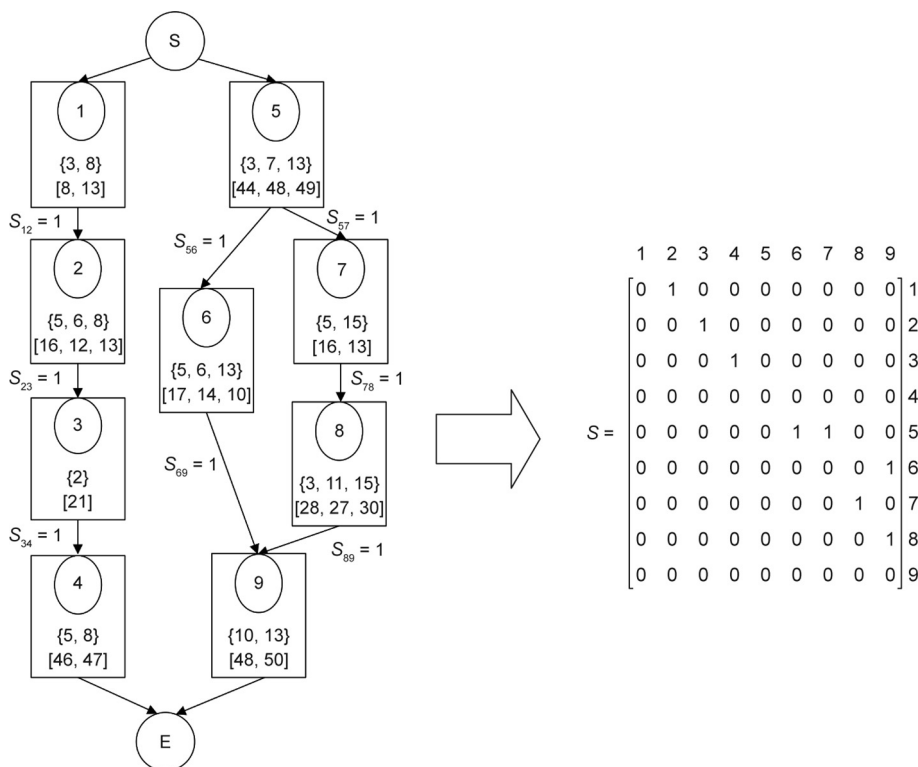


Fig. 4. Precedence constraints in a network and the corresponding matrix S.

matrix is  $(n - 1)$ . For example, to determine the sequence  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9$ , it only needs to set eight variables as “1,” which are  $q_{12}, q_{23}, q_{34}, q_{45}, q_{56}, q_{67}, q_{78}$ , and  $q_{89}$ . From this point, it is helpful to define a notation  $v_{ij}$  whose corresponding matrix  $V = [v_{ij}]$  contains the least number of variables equal to 1. The matrix  $V$  can be regarded as an exactly expressing matrix (EEM) of the precedence relationships of the sequence, which contains the exact and least number of “1”s. The EEM  $V$  of the above operation sequence is shown in Fig. 5.

The EEM matrix can also be directly called the precedence matrix, since it just contains the precedence relationship of the two directly adjacent operations. According to the direct precedence relationships, it is simple to obtain an operation sequence by sequentially identifying the elements of the EEM  $V$ . The EEM matrix has several characteristic constraints as follows:

(1) The number of variables that equal to 1 in the EEM  $V$  is  $(n - 1)$ :

$$\sum_j \sum_{j'} v_{jj'} = n - 1 \tag{5}$$

(2) Each row or column of the EEM  $V$  has at most one element equal to 1:

$$\sum_j v_{jj'} \leq 1, \forall j' \tag{6}$$

$$\sum_{j'} v_{jj'} \leq 1, \forall j \tag{7}$$

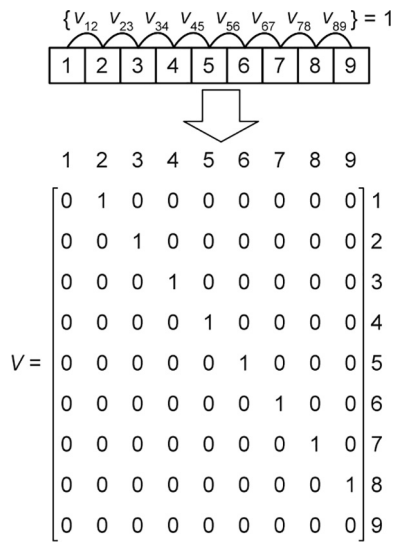


Fig. 5. The EEM  $V$  of the sequence.

(3) The relationship between matrix  $Q$  and matrix  $V$  is expressed as follows:

$$v_{jj'} \leq q_{jj'}, \forall j, j' \tag{8}$$

Matrix  $V$  is a simplified representation of matrix  $Q$ , and both of them can determine a unique operation sequence. However, matrix  $S$  cannot determine a unique operation sequence due to its incomplete representation of the precedence relationships of a determined sequence. Therefore, matrix  $S$  is named the partly expressing matrix (PEM) in this paper. More than one sequence can satisfy the precedence constraints expressed by an  $S$  matrix. CEM  $Q$ , EEM  $V$ , and PEM  $S$  are shown in Fig. 6.

3.3. A discussion on feature-based and network-based process representations

In the current literature, all the mathematical models for PP problems are feature-based. For example, an instance adopted in the literature [8,16] is shown in feature tabular form in Fig. 7(a). In fact, this instance is derived from the 18th example in Ref. [29], as shown in Fig. 7(b). Therefore, Figs. 7(a) and (b) illustrate the same instance expressed in two different representation forms.

In the feature table [8,16], features  $F_2, F_5$ , and  $F_9$  have alternative operations or operation sets, corresponding to the three OR nodes OR1, OR2, and OR3 in the network graph. However, in the tabular form, the operation sets under the same feature are constrained by direct precedence relationships that do not exist in the corresponding network graph. For example, if  $O_4-O_5$  is chosen for feature  $F_2$ , then  $O_5$  must be processed directly after  $O_4$ . However,  $O_5$  does not have to be processed immediately after  $O_4$  according to the original network graph Fig. 7(b). As a result, the solution space might be changed, leading to the failure to obtain the optimal solution. Table 1 shows two optimal solutions respectively obtained by solving a feature-based model and a network-based model with the GAMS/CPLEX solver. The letter  $M$  with number subscript in the bracket means the allocated machine, and it is the same in the follow-up tables.

Observed from Table 1, the main distinction between these two sequences is that the sequence obtained by the network-based method does not have the direct precedence constraint of  $O_4-O_5$ . In addition, the production time of 356 obtained by the network-based method is superior to the value of 357 obtained by the feature-based method. Therefore, the method of building a model directly on the basis of the network is superior to the feature-based method.

The process flexibility of the feature-based representation is implemented through alternative operations or operation sets for features. For the network-based representation, the process flexibility is expressed by selecting the links of OR nodes. The links refer to the arrow connected to the OR nodes. Each link of the OR nodes

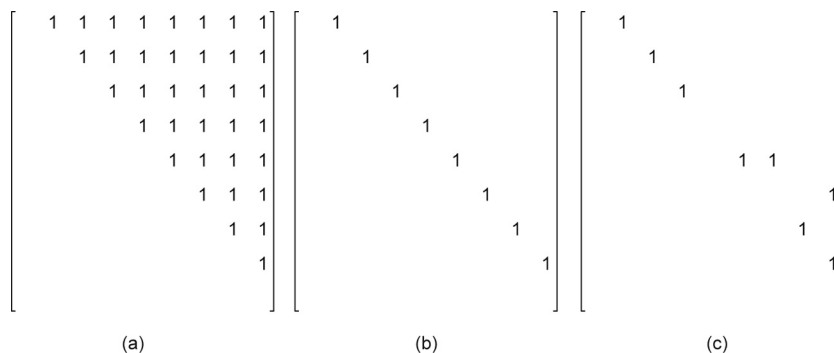


Fig. 6. Three types of precedence relationship matrices: (a) CEM  $Q$ , (b) EEM  $V$ , and (c) PEM  $S$ .

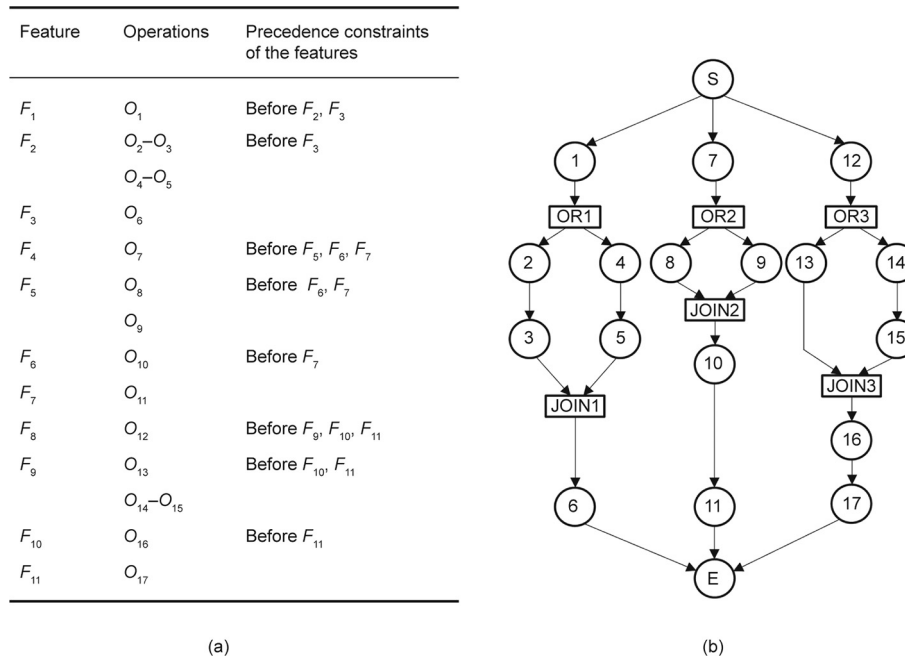


Fig. 7. Two representation forms of the same instance: (a) feature tabular form and (b) network graph.

Table 1

Two optimal solutions obtained by using the feature-based model and the network-based model.

Result	Feature-based model	Network-based model
Optimal process plan	$O_7(M_3)-O_1(M_3)-O_{12}(M_5)-O_4(M_5)-O_5(M_9)-O_{13}(M_9)-O_9(M_{15})-O_{10}(M_{15})-O_{16}(M_{15})-O_6(M_8)-O_{17}(M_{10})-O_{11}(M_{10})$	$O_7(M_3)-O_1(M_3)-O_4(M_1)-O_8(M_{13})-O_{12}(M_{13})-O_{10}(M_3)-O_5(M_9)-O_{13}(M_9)-O_{16}(M_4)-O_6(M_8)-O_{11}(M_{10})-O_{17}(M_{10})$
Production time	357	356

corresponds to a choice for the process flexibility. In Fig. 8, according to the order from the top to bottom and the left to right, the OR nodes and their links are numbered as shown. If link 1 of OR node 1 is selected, then operations 2, 3, and 4 will be selected. Additionally, whether link 1 or 2 of OR node 2 is selected, operations 6 and 7 will not be selected. The reason is that, in addition to OR node 2, operations 5, 6, 7, and 8 are controlled by link 2 of OR node 1. The selection performed by the OR node is valid only if the link at which this OR node lies is selected.

A binary parameter  $w_{jrl}$  is introduced to describe the control function of OR nodes on the operations. The definition of the parameter  $w_{jrl}$  is stated as follows:

$$w_{jrl} = \begin{cases} 1, & \text{if operation } j \text{ is controlled by the } l\text{th link of the } r\text{th OR node} \\ 0, & \text{otherwise} \end{cases}$$

For the example in Fig. 8, the corresponding values of  $w_{jrl}$  are shown in Table 2.

Hence, the controlling function of the OR node can be concluded as: Operation  $j$  will be selected only under the condition that all the controlling links of operation  $j$  are selected. On the contrary, operation  $j$  will not be selected as long as one of its controlling links is not selected. A binary variable  $u_{rl}$  is introduced to describe the choice of the links, and another binary variable  $x_j$  is used to describe the operation choosing state. The definitions of  $u_{rl}$  and  $x_j$  are given as follows:

$$u_{rl} = \begin{cases} 1, & \text{if the } l\text{th link of } r\text{th OR node is selected} \\ 0, & \text{otherwise} \end{cases}$$

$$x_j = \begin{cases} 1, & \text{if operation } j \text{ is selected} \\ 0, & \text{otherwise} \end{cases}$$

The model established in Section 3.4 is based on these two variables,  $u_{rl}$  and  $x_j$ , and on a parameter,  $w_{jrl}$ .

### 3.4. Mathematical model for PP

The proposed MILP model is precedence-based and OR-node-based. Most of the process optimization objective functions are time related [8] or cost related [14,18]. In this paper, with the objective of minimizing the production time, the transmission time between machines is taken into account in the model. The sets, subscripts, parameters, and variables of the model are introduced below (Table 3).

The total production time as the objective can be formulated as follows:

$$\min TT = \sum_j MT_j + \sum_j \sum_k T_{jk} \cdot y_{jk} \tag{9}$$

On the right side, the first part of Eq. (9) refers to the total transmission time, and the second is the total processing time. The constraints of the model are displayed as follows:

(1) OR-node controlling constraints:

$$x_j \leq M \cdot (1 - w_{jrl}) + M \cdot u_{rl}, \forall j, r, l \tag{10}$$

$$x_j \geq 1 - M \cdot \sum_r \sum_l w_{jrl} \cdot (1 - u_{rl}), \forall j \tag{11}$$

$$\sum_l u_{rl} = 1, \forall r \tag{12}$$

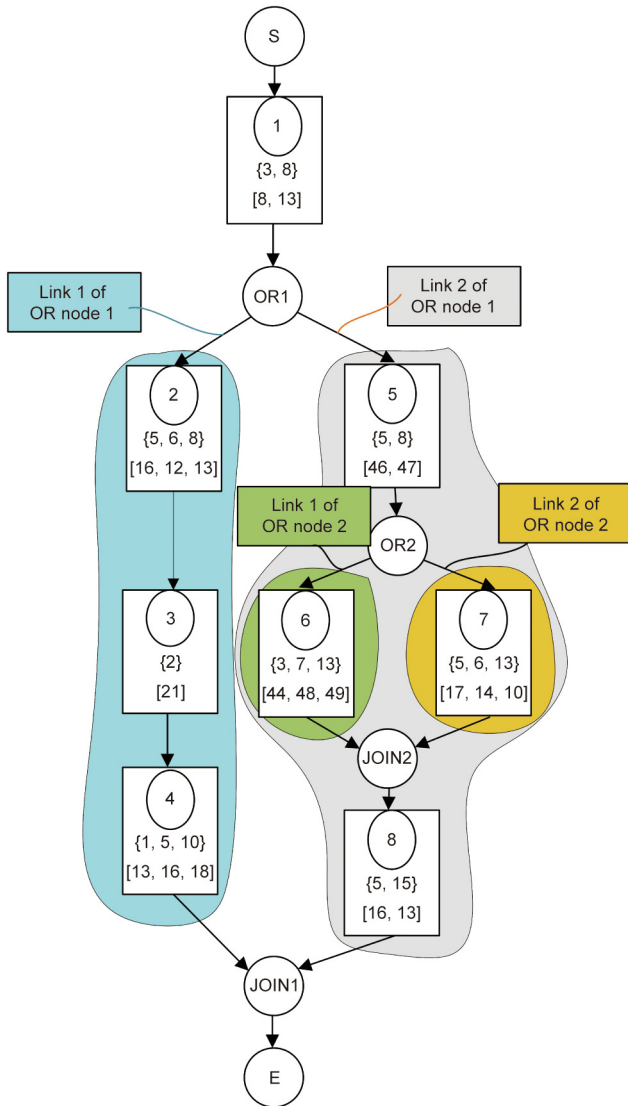


Fig. 8. An example of a controlling discussion.

The constraint in Eq. (10) indicates the “unselected” condition: Operation  $j$  will not be selected as long as one of all operation  $j$ 's controlling links is not selected. The constraint in Eq. (11) is the “selected” condition. The constraint in Eq. (12) means that only one link of an OR node can be chosen.

(2) Precedence constraints:

$$q_{jj} = 0, \forall j \tag{13}$$

$$q_{j'j} + q_{jj'} \leq 1 + M \cdot (2 - x_j - x_{j'}), \forall j \neq j' \tag{14}$$

$$q_{j'j} + q_{jj'} \geq 1 - M \cdot (2 - x_j - x_{j'}), \forall j \neq j' \tag{15}$$

$$M \cdot (2 - x_j - x_{j'}) + \sum_{j''} q_{jj''} - \sum_{j''} q_{j''j} \neq 0, \forall j \neq j' \tag{16}$$

$$q_{j'j} + q_{jj'} \leq M \cdot x_j, \forall j, j' \tag{17}$$

$$(2 - x_j - x_{j'}) \cdot M + q_{j'j} \geq s_{j'j}, \forall j, j' \tag{18}$$

Eqs. (13)–(18) refer to the constraints in the precedence relationships. Unlike Eqs. (1)–(4), the operation selecting condition is added to this group of constraints. Eq. (13) corresponds to Eq. (1). Eqs. (14) and (15) correspond to Eq. (2), on the condition that operation  $j$  and  $j'$  are selected. Under the same condition, the constraint in Eq. (16) corresponds to Eq. (3). As for Eq. (17), the meaning is that the precedence relationships  $q_{j'j}$  and  $q_{jj'}$  should be set as 0 if operation  $j$  is unselected. The constraint in Eq. (18) guarantees that the selected operation sequence obeys the precedence constraints.

(3) EEM  $V$  and CEM  $Q$  constraints:

$$\sum_j \sum_{j'} v_{j'j} = \sum_j x_j - 1 \tag{19}$$

$$\sum_{j'} v_{j'j} \leq 1, \forall j \tag{20}$$

$$\sum_j v_{j'j} \leq 1, \forall j' \tag{21}$$

$$v_{j'j} \leq q_{j'j}, \forall j, j' \tag{22}$$

Since a matrix  $V$  contains the precedence relationships between the two adjacent operations, the transmission time can easily be calculated. The constraints in Eqs. (19)–(21) describe the property of matrix  $V$ . Because matrix  $V$  is derived from matrix  $Q$ , Eq. (22) is presented.

(4) Machine selection constraint:

$$\sum_k z_{jk} \cdot y_{jk} = x_j, \forall j \tag{23}$$

Eq. (23) means that there is only one machine that can be assigned for the selected operation, and there is no need to assign any machine for an unselected operation.

(5) Transmission constraints:

$$MT_j \leq TK_{kk'} + M \cdot (1 - v_{j'j}) + M \cdot (2 - y_{jk} - y_{j'k'}), \forall j, j', k, k' \tag{24}$$

$$MT_j \geq TK_{kk'} - M \cdot (1 - v_{j'j}) - M \cdot (2 - y_{jk} - y_{j'k'}), \forall j, j', k, k' \tag{25}$$

Eqs. (24) and (25) formulate the transmission time of operation  $j$  from the current processing machine to the next machine.

#### 4. Experiments and discussions

To verify the proposed model, five groups of comparative experiments are carried out based on famous benchmarks. All the experiments are directly compared with the results of other reported methods. On a personal computer (PC) with 3.7 GHz and 16 GB random-access memory (RAM), the proposed model is coded in the GAMS, and the solver CPLEX is used to solve the PP problems. In this paper, the parameter Gap (%) is also introduced to evaluate the proposed model and the computation results. The Gap value represents the relative tolerance of the obtained solution; its definition is  $(BF - BP)/BP$ , where BF is the current best

Table 2  
The values of parameter  $w_{jrl}$ .

Values of $r$ and $l$	$w_{jrl}$							
	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$	$j = 7$	$j = 8$
$r = 1, l = 1$	0	1	1	1	0	0	0	0
$r = 1, l = 2$	0	0	0	0	1	1	1	1
$r = 2, l = 1$	0	0	0	0	0	1	0	0
$r = 2, l = 2$	0	0	0	0	0	0	1	0

**Table 3**  
Definitions of the sets, subscripts, parameters, and variables of the mathematical model for PP.

Items	Definition
<b>Sets and subscripts</b>	
$j, j'$	Operations, $j = 1, 2, \dots, n$ , where $n$ is the total number of operations
$k, k'$	Machines, $k = 1, 2, \dots, K$ , where $K$ is the total number of machines
$r$	OR nodes, $r = 1, 2, \dots, R$ , where $R$ is the total number of OR nodes
$l$	Links, $l = 1, 2, \dots, L$ , where $L$ is the total number of links of one node
<b>Parameters</b>	
$s_{jj'}$	The precedence constraints in the network: If operation $j$ is supposed to be processed before operation $j'$ , then $s_{jj'} = 1$ ; otherwise, $s_{jj'} = 0$
$z_{jk}$	1, machine $k$ is optional for operation $j$ ; 0, otherwise
$w_{jrl}$	1, operation $j$ is controlled by the $l$ th link of the $r$ th OR node; 0, otherwise
$T_{jk}$	The processing time of operation $j$ on machine $k$
$TK_{kk'}$	The transmission time from machine $k$ to $k'$
$M$	A large enough positive number
<b>Variables</b>	
$q_{jj'}$	1, operation $j$ is processed before operation $j'$ ; 0, otherwise
$v_{jj'}$	1, operation $j$ is processed directly before operation $j'$ ; 0, otherwise
$y_{jk}$	1, operation $j$ is processed on machine $k$ ; 0, otherwise
$u_{rl}$	1, the $l$ th link of the $r$ th OR node is selected; 0, otherwise
$x_j$	1, operation $j$ is selected to process the part; 0, otherwise
$MT_j$	The transmission time from the machine processing operation $j$ to the next machine to process the operation
$TT$	The total production time

solution of the objective function, and BP is the lower bound. The smaller the Gap value is, the closer the current solution is to the optimal solution. The computation time of GAMS/CPLEX is set as 3600 s. If the optimal solution is not found within the time limit, the computation will be terminated and the best known solution will be output. The transmission time between machines adopted by the cases in experiments 1, 2, 4, and 5 is shown in Table 4 [8,18].

4.1. Experiment 1

The three cases in experiment 1 are adopted from Jin and Zhang [16], where a dynamic programming (DP)-like heuristic algorithm is applied to solve the PP problem. The results obtained by the proposed MILP model and the DP-like heuristic are presented in Table 5. The production time of case 1 obtained by the MILP method is 357, which is better than the 360 provided by the DP-like heuristic. Furthermore, all three optimal solutions to the cases are found by the MILP model.

4.2. Experiment 2

The four cases in experiment 2 are from different publications: Case 1 comes from Zhang and Nee [36], and cases 2–4 come from Li

**Table 4**  
The transmission time matrix [8,18].

Machine	Machine														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	5	7	9	10	11	7	6	14	13	12	10	5	6	9
2	5	0	3	4	5	7	2	7	6	4	7	12	10	7	8
3	7	3	0	6	5	4	3	7	2	4	3	5	6	8	9
4	9	4	6	0	4	4	6	7	4	10	12	13	14	15	16
5	10	5	5	4	0	10	12	7	8	9	10	12	11	10	8
6	11	7	4	4	10	0	4	4	5	5	6	7	6	7	8
7	7	2	3	6	12	4	0	5	6	6	6	7	7	7	8
8	6	7	7	7	7	4	5	0	4	2	3	4	2	4	3
9	14	6	2	4	8	5	6	4	0	5	7	4	7	6	8
10	13	4	4	10	9	5	6	2	5	0	8	10	12	14	7
11	12	7	3	12	10	6	6	3	7	8	0	7	10	14	10
12	10	12	5	13	12	7	7	4	4	10	7	0	10	12	10
13	5	10	6	14	11	6	7	2	7	12	10	10	0	8	8
14	6	7	8	15	10	7	7	4	6	14	14	12	8	0	9
15	9	8	9	16	8	8	8	3	8	7	10	10	8	9	0

and McMahon [37]. The detailed information of these cases can be found in the corresponding papers. Computational results from a modified PSO algorithm are given in the work of Li et al. [8]. The modified PSO algorithm is one of the state-of-the-art algorithms that are used to solve combinatorial optimization problems. The comparison between PSO and MILP is given in Table 6. It can be observed that the MILP model obtains the better solution in case 1. Furthermore, the optimal solutions for cases 2–4 are found by the MILP model method within short computational time (less than one second).

4.3. Experiment 3

The two cases in experiment 3 are adopted from Li et al. [15], and the machine transmission time matrix is shown in Table 7. These two cases employed the same part from Li et al. [15]. The only distinction between the two cases is that machine 2 is assumed to be broken down in case 2. The comparative results are listed in Table 8. Both the GP algorithm and the MILP model are able to find the optimal solutions of the two instances.

4.4. Experiment 4

The 17 cases in experiment 4 are adopted from the well-known Kim dataset [29], which consists of 18 parts, and the comparative calculation results are from Ref. [8]. Because there are some problems regarding the data of part 4 from Ref. [8], it is not selected here. The results obtained by the modified PSO algorithm, the simple GA, and the simple SA are presented in Table 9. Within a reasonable computational time, the proposed MILP model can find 13 optimal solutions out of 17 cases. For the cases that the optimal solutions are not found, the MILP model still obtains better solutions compared to the other algorithms, such as cases 3, 6, 12, and 15.

4.5. Experiment 5

The 11 cases in experiment 5 are adopted from another famous Shin benchmark [21], and the comparative calculation results are from Li et al. [8]. Because there are some problems regarding the data of some parts in Ref. [8], 11 parts (omitting parts 9, 10, 12, 13, 15, 17, and 18) are selected in this experiment group. The solutions obtained by the modified PSO algorithm, the simple GA, and the simple SA are presented in Table 10. Within a reasonable computational time, the proposed MILP model can find nine optimal solutions out of 11 instances. For the cases that the optimal solutions are not found, the MILP model still obtains better solutions, such as case 3.



**Table 5**  
Comparative results of experiment 1.

Case		DP-like heuristic [16]	MILP
1	Process plan	$O_{12}(M_5)-O_7(M_3)-O_1(M_3)-O_4(M_{10})-O_5(M_9)-O_{13}(M_9)-O_9(M_{15})-O_{10}(M_{15})-O_{16}(M_{15})-O_6(M_8)-O_{17}(M_{10})-O_{11}(M_{10})$	$O_7(M_3)-O_1(M_3)-O_{12}(M_5)-O_4(M_5)-O_5(M_9)-O_{13}(M_9)-O_9(M_{15})-O_{10}(M_{15})-O_{16}(M_{15})-O_6(M_8)-O_{17}(M_{10})-O_{11}(M_{10})$
	Production time	360	357 <sup>a</sup>
	Computing time	—	3546.74
	Gap	—	0%
2	Process plan	$O_{13}(M_1)-O_4(M_1)-O_2(M_1)-O_1(M_1)-O_7(M_1)-O_8(M_1)-O_9(M_1)-O_{12}(M_1)-O_{10}(M_1)-O_{11}(M_1)-O_5(M_1)-O_6(M_1)-O_3(M_1)$	$O_{13}(M_1)-O_2(M_1)-O_4(M_1)-O_1(M_1)-O_3(M_1)-O_{10}(M_1)-O_{11}(M_1)-O_7(M_1)-O_8(M_1)-O_9(M_1)-O_5(M_1)-O_6(M_1)-O_{12}(M_1)$
	Production time	222	222 <sup>a</sup>
	Computing time	—	12.16
	Gap	—	0%
3	Process plan	$O_1(M_2)-O_3(M_2)-O_5(M_2)-O_6(M_5)-O_9(M_5)-O_4(M_5)-O_7(M_5)-O_8(M_4)-O_2(M_4)$	$O_1(M_2)-O_3(M_2)-O_5(M_2)-O_6(M_5)-O_4(M_5)-O_9(M_5)-O_7(M_5)-O_8(M_4)-O_2(M_4)$
	Production time	212	212 <sup>a</sup>
	Computing time	—	0.44
	Gap	—	0%

<sup>a</sup> Indicates that the optimal solution is found.

**Table 6**  
Comparative results of experiment 2.

Case		Modified PSO [8]	MILP
1	Process plan	$O_{12}(M_4)-O_2(M_4)-O_{15}(M_4)-O_{16}(M_4)-O_3(M_2)-O_{13}(M_1)-O_8(M_1)-O_9(M_1)-O_{10}(M_1)-O_4(M_1)-O_1(M_1)-O_{17}(M_1)-O_{14}(M_1)-O_5(M_1)-O_6(M_1)-O_7(M_1)-O_{11}(M_1)-O_{18}(M_1)$	$O_{16}(M_4)-O_{15}(M_4)-O_2(M_4)-O_{12}(M_4)-O_1(M_2)-O_8(M_2)-O_9(M_2)-O_{10}(M_2)-O_3(M_2)-O_5(M_2)-O_6(M_1)-O_7(M_1)-O_{14}(M_1)-O_4(M_1)-O_{13}(M_1)-O_{17}(M_1)-O_{18}(M_1)-O_{11}(M_1)$
	Objective value	359	358
	Computing time	—	3600.00
	Gap	—	4.71%
2	Process plan	$O_1(M_4)-O_5(M_4)-O_{18}(M_4)-O_2(M_4)-O_6(M_4)-O_{11}(M_4)-O_{12}(M_4)-O_{13}(M_4)-O_{14}(M_4)-O_7(M_4)-O_4(M_4)-O_{17}(M_4)-O_{15}(M_4)-O_{16}(M_4)-O_8(M_4)-O_9(M_4)-O_{10}(M_4)-O_{19}(M_4)-O_{20}(M_4)-O_3(M_4)$	$O_1(M_4)-O_{18}(M_4)-O_5(M_4)-O_5(M_4)-O_7(M_4)-O_8(M_4)-O_9(M_4)-O_{10}(M_4)-O_2(M_4)-O_{11}(M_4)-O_{17}(M_4)-O_4(M_4)-O_6(M_4)-O_{12}(M_4)-O_{13}(M_4)-O_{14}(M_4)-O_{19}(M_4)-O_{20}(M_4)-O_{15}(M_4)-O_{16}(M_4)$
	Objective value	341	341 <sup>a</sup>
	Computing time	—	0.20
	Gap	—	0%
3	Process plan	$O_1(M_4)-O_2(M_4)-O_9(M_4)-O_{10}(M_4)-O_{11}(M_4)-O_5(M_4)-O_3(M_4)-O_6(M_4)-O_4(M_4)-O_7(M_4)-O_{12}(M_4)-O_{13}(M_4)-O_8(M_4)-O_{14}(M_4)$	$O_1(M_4)-O_2(M_4)-O_5(M_4)-O_4(M_4)-O_3(M_4)-O_6(M_4)-O_9(M_4)-O_{10}(M_4)-O_{11}(M_4)-O_7(M_4)-O_{12}(M_4)-O_{13}(M_4)-O_{14}(M_4)-O_{14}(M_4)$
	Objective value	176	176 <sup>a</sup>
	Computing time	—	0.44
	Gap	—	0%
4	Process plan	$O_{16}(M_4)-O_3(M_4)-O_5(M_4)-O_1(M_4)-O_6(M_4)-O_7(M_4)-O_{14}(M_4)-O_{11}(M_4)-O_{15}(M_4)-O_9(M_4)-O_{12}(M_4)-O_4(M_4)-O_{13}(M_4)-O_2(M_4)-O_8(M_4)-O_{10}(M_4)$	$O_{16}(M_4)-O_6(M_4)-O_7(M_4)-O_5(M_4)-O_{11}(M_4)-O_{13}(M_4)-O_{12}(M_4)-O_9(M_4)-O_{14}(M_4)-O_1(M_4)-O_3(M_4)-O_2(M_4)-O_{15}(M_4)-O_4(M_4)-O_{10}(M_4)-O_8(M_4)$
	Objective value	187	187 <sup>a</sup>
	Computing time	—	0.39
	Gap	—	0%

<sup>a</sup> Indicates that the optimal solution is found.

**Table 7**  
Transmission time between the machines.

Machine	Machine									
	1	2	3	4	5	6	7	8	9	10
1	0	5	8	12	15	4	6	10	13	18
2	5	0	3	7	10	6	4	6	10	13
3	8	3	0	4	7	10	6	4	6	10
4	12	7	4	0	3	14	10	6	4	6
5	15	10	7	3	0	18	12	10	6	4
6	4	6	10	14	18	0	5	8	12	15
7	6	4	6	10	12	5	0	3	7	10
8	10	6	4	6	10	8	3	0	4	8
9	13	10	6	4	6	12	7	4	0	4
10	18	13	10	6	4	15	10	8	4	0

4.6. Discussion

The proposed MILP model obtained 28 optimal solutions out of 37 instances within acceptable calculation time. The solutions found by the MILP model are better than those obtained by the high-performance heuristic [16] and meta-heuristic

algorithms [8,15]. Experiments 4 and 5 were carried out on two widely-used benchmarks [21,29], and the better results suggest the superiority of the proposed model.

As shown in Table 11, the proposed model contains four types of subscripts, that is, operation, machine, OR node, and links, while the model reported in Ref. [16] contains six types

of subscripts, that is, feature, operation set, operation, machine, position, and place. Fewer subscripts in the proposed model make the computation more effective than that of the model developed by Ref. [16]. Furthermore, the OR-node-based modeling method makes the proposed model more universal for solving different types of PP problems. This is why most optimal

solutions can be obtained for both of the Kim [29] and Shin [21] benchmarks.

### 5. Conclusions and future work

Considering the topology of network graph, this paper proposed a new MILP mathematical model based on OR nodes. Firstly, for precedence relationships between operations, three precedence matrices were introduced. Secondly, for better generality, the notations  $w_{jrl}$ ,  $u_{rl}$ , and  $x_j$  were introduced to describe the controlling function of the OR nodes. Finally, the proposed MILP model were coded in the mathematical programming solver CPLEX and tested on public benchmarks. The extensive comparative results verified the correctness and superiority of the proposed model.

In this work, an OR-node-based modeling method was proposed for the first time, demonstrating a new perspective for PP problems and their extension research. The analyses of the three precedence matrices in the paper also revealed the essence of the operation sequencing sub-problem, which is beneficial for further comprehension of the PP problem. However, there are still some limitations in this PP model research. Optimal solutions cannot be found for the minority of instances, and the computational efficiency is not always satisfactory, which implies that the proposed approaches can be further improved. Some simplification and speed-up strategies are urgently required for the further research work.

**Table 8**  
Comparative results of experiment 3.

Case	GP [15]	MILP
1 Process plan	$O_1(M_2)-O_2(M_7)-O_3(M_2)-O_4(M_2)-O_9(M_3)$	$O_1(M_2)-O_2(M_7)-O_3(M_2)-O_4(M_2)-O_9(M_3)$
Objective value	213	213 <sup>a</sup>
Computing time	–	0.16
Gap	–	0%
2 Process plan	$O_{10}(M_9)-O_{11}(M_3)-O_{15}(M_7)$	$O_{10}(M_8)-O_{11}(M_2)-O_{15}(M_6)$
Objective value	224	224 <sup>a</sup>
Computing time	–	0.13
Gap	–	0%

<sup>a</sup> Indicates that the optimal solution is found.

**Table 9**  
Comparative results of experiment 4.

Case	Objective value				Gap	Computational time
	SA	GA	PSO	MILP		
1	303	303	303	292 <sup>a</sup>	0%	0.08
2	359	359	359	351 <sup>a</sup>	0%	191.50
3	502	502	498	486	45.5%	3600.00
5	314	314	314	280 <sup>a</sup>	0%	26.45
6	409	409	408	399	40.5%	3600.00
7	304	304	304	304 <sup>a</sup>	0%	27.02
8	358	358	358	353 <sup>a</sup>	0%	63.44
9	393	392	391	391	38.1%	3600.00
10	264	264	264	264 <sup>a</sup>	0%	1.27
11	271	271	271	264 <sup>a</sup>	0%	22.24
12	442	442	442	433	34.9%	3600.00
13	216	216	216	215 <sup>a</sup>	0%	10.69
14	269	269	269	244 <sup>a</sup>	0%	56.77
15	358	357	357	353	5.1%	3600.00
16	248	248	248	244 <sup>a</sup>	0%	3071.28
17	314	314	314	300 <sup>a</sup>	0%	3600.00
18	361	361	360	356 <sup>a</sup>	0%	3600.00

<sup>a</sup> Indicates that the optimal solution is found.

**Table 10**  
Comparative results of experiment 5.

Case	Objective value				Gap	Computing time
	SA	GA	PSO	MILP		
1	267	267	267	266 <sup>a</sup>	0%	19.55
2	165	165	163	162 <sup>a</sup>	0%	30.06
3	299	297	296	268	33.7%	3600.00
4	268	268	267	267	20.2%	3600.00
5	204	204	204	199 <sup>a</sup>	0%	463.20
6	204	204	204	189 <sup>a</sup>	0%	268.33
7	137	137	137	116 <sup>a</sup>	0%	0.94
8	181	181	181	178 <sup>a</sup>	0%	0.20
11	151	150	149	149 <sup>a</sup>	0%	69.88
14	120	120	120	120 <sup>a</sup>	0%	11.06
16	170	167	167	167 <sup>a</sup>	0%	620.38

<sup>a</sup> Indicates that the optimal solution is found.

**Table 11**  
The subscripts of the models.

Subscripts	Definition
Proposed MILP model	
$j, j'$	Operations
$k, k'$	Machines
$r$	OR nodes
$l$	Links
Reported model in Ref. [16]	
$i, i'$	Features
$j, j'$	Operation sets
$k, k'$	Operations
$l, l'$	Machines
$r$	OR nodes
$h$	Place, where a feature can be assigned

## Acknowledgements

This work is supported in part by the National Natural Science Foundation of China (51825502 and 51775216), and in part by the Program for Huazhong University of Science and Technology (HUST) Academic Frontier Youth Team (2017QYTD04).

## Compliance with ethics guidelines

Qihao Liu, Xinyu Li, and Liang Gao declare that they have no conflict of interest or financial conflicts to disclose.

## References

- Zhou J, Li P, Zhou Y, Wang B, Zang J, Meng L. Toward new-generation intelligent manufacturing. *Engineering* 2018;4(1):11–20.
- Zhong RY, Xu X, Klotz E, Newman ST. Intelligent manufacturing in the context of industry 4.0: a review. *Engineering* 2017;3(5):616–30.
- Zhou J, Zhou Y, Wang B, Zang J. Human–cyber–physical systems (HCPSs) in the context of new-generation intelligent manufacturing. *Engineering* 2019;5(4):624–36.
- Leo Kumar SP. State of the art-intense review on artificial intelligence systems application in process planning and manufacturing. *Eng Appl Artif Intell* 2017;65:294–329.
- Miljković Z, Petrović M. Application of modified multi-objective particle swarm optimisation algorithm for flexible process planning problem. *Int J Comput Integrated Manuf* 2017;30(2–3):271–91.
- Xu X, Wang L, Newman ST. Computer-aided process planning—a critical review of recent developments and future trends. *Int J Comput Integrated Manuf* 2011;24(1):1–31.
- Li X, Gao L, Pan Q, Wan L, Chao KM. An effective hybrid genetic algorithm and variable neighborhood search for integrated process planning and scheduling in a packaging machine workshop. *IEEE Trans Syst Man Cybern Syst* 2019;49(10):1933–45.
- Li X, Gao L, Wen X. Application of an efficient modified particle swarm optimization algorithm for process planning. *Int J Adv Manuf Technol* 2013;67(5–8):1355–69.
- Gan PY, Lee KS, Zhang YF. A branch and bound algorithm based process-planning system for plastic injection mould bases. *Int J Adv Manuf Technol* 2001;18(9):624–32.
- Wen X, Li X, Gao L, Sang H. Honey bees mating optimization algorithm for process planning problem. *J Intell Manuf* 2014;25(3):459–72.
- Zhang J, Xiao Mi, Gao L, Pan Q. Queuing search algorithm: a novel metaheuristic algorithm for solving engineering optimization problems. *Appl Math Model* 2018;63:464–90.
- Liu Q, Li X, Gao L, Li Y. A modified genetic algorithm with new encoding and decoding methods for integrated process planning and scheduling problem. *IEEE Trans Cybern. Epub* 2020 Oct 15.
- Li WD, Ong SK, Nee AYC. Optimization of process plans using a constraint-based tabu search approach. *Int J Prod Res* 2004;42(10):1955–85.
- Liu X, Yi H, Ni Z. Application of ant colony optimization algorithm in process planning optimization. *J Intell Manuf* 2013;24(1):1–13.
- Li XY, Shao XY, Gao L. Optimization of flexible process planning by genetic programming. *Int J Adv Manuf Technol* 2008;38(1–2):143–53.
- Jin L, Zhang C. Process planning optimization with energy consumption reduction from a novel perspective: mathematical modeling and a dynamic programming-like heuristic algorithm. *IEEE Access* 2019;7:7381–96.
- Jiang Z, Jiang Ya, Wang Y, Zhang H, Cao H, Tian G. A hybrid approach of rough set and case-based reasoning to remanufacturing process planning. *J Intell Manuf* 2019;30(1):19–32.
- Jin L, Tang Q, Zhang C, Shao X, Tian G. More MILP models for integrated process planning and scheduling. *Int J Prod Res* 2016;54(14):4387–402.
- Li WD, Ong SK, Nee AYC. Hybrid genetic algorithm and simulated annealing approach for the optimization of process plans for prismatic parts. *Int J Prod Res* 2002;40(8):1899–922.
- Hua G, Zhou X, Ruan X. GA-based synthesis approach for machining scheme selection and operation sequencing optimization for prismatic parts. *Int J Adv Manuf Technol* 2007;33(5–6):594–603.
- Shin KS, Park JO, Kim YK. Multi-objective FMS process planning with various flexibilities using a symbiotic evolutionary algorithm. *Comput Oper Res* 2011;38(3):702–12.
- Wang YF, Zhang YF, Fuh JYH. A hybrid particle swarm based method for process planning optimisation. *Int J Prod Res* 2012;50(1):277–92.
- Shabaka AI, ElMaraghy HA. A model for generating optimal process plans in RMS. *Int J Comput Integrated Manuf* 2008;21(2):180–94.
- Floudas CA, Lin X. Mixed integer linear programming in process scheduling: modeling, algorithms, and applications. *Ann Oper Res* 2005;139(1):131–62.
- Li X, Gao L, Shao X, Zhang C, Wang C. Mathematical modeling and evolutionary algorithm-based approach for integrated process planning and scheduling. *Comput Oper Res* 2010;37(4):656–67.
- Xia Q, Etienne A, Dantan J, Siadat A. Reconfigurable machining process planning for part variety in new manufacturing paradigms: definitions, models and framework. *Comput Ind Eng* 2018;115:206–19.
- Lee K, Jung M. Petri net application in flexible process planning. *Comput Ind Eng* 1994;27(1–4):505–8.
- Sormaz D, Sarkar A. Process sequencing problem in distributed manufacturing process planning. In: Goldengorin B, editor. *Optimization problems in graph theory*. Berlin: Springer; 2018. p. 293–324.
- Kim YK, Park K, Ko J. A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling. *Comput Oper Res* 2003;30(8):1151–71.
- Zhang S, Wong TN. Integrated process planning and scheduling: an enhanced ant colony optimization heuristic with parameter tuning. *J Intell Manuf* 2018;29(3):585–601.
- Pan CH. A study of integer programming formulations for scheduling problems. *Int J Syst Sci* 1997;28(1):33–41.
- Fattahi P, Saidi Mehrobad M, Jolai F. Mathematical modeling and heuristic approaches to flexible job shop scheduling problems. *J Intell Manuf* 2007;18(3):331–42.
- Zhang L, Tang Q, Wu Z, Wang F. Mathematical modeling and evolutionary generation of rule sets for energy-efficient flexible job shops. *Energy* 2017;138:210–27.
- Özgvün C, Özbakır L, Yavuz Y. Mathematical models for job-shop scheduling problems with routing and process plan flexibility. *Appl Math Model* 2010;34(6):1539–48.
- Liu Q, Li X, Gao L. Mathematical modeling and a hybrid evolutionary algorithm for process planning. *J Intell Manuf* 2021;32(3):781–97.
- Zhang Y, Nee A. Application of genetic algorithms and simulated annealing in process planning optimization. In: Pham DT, editor. *Computational intelligence in manufacturing handbook*. Boca Raton: CRC Press; 2000. p. 243–68.
- Li WD, McMahon CA. A simulated annealing-based optimization approach for integrated process planning and scheduling. *Int J Comput Integrated Manuf* 2007;20(1):80–95.