Mechatronics and Automation Technology
J. Xu (Ed.)

doi:10.3233/ATDE221143

Research on Collaborative Flexible Job-Shop Scheduling Technology

Tongming XU[a,1], Zhou LI[b] and Huanhu FENG[a]
[a] *Inspur General Software Co., Ltd., China*
[b] *EpicHust Technology (Wuhan) Co., Ltd., China*

Abstract. To improve the management of collaborative manufacturing tasks in MES and promote the digital transformation of discrete manufacturing enterprises, the collaborative manufacturing of discrete manufacturing based on MES has been studied. First, the business flow of collaborative manufacturing in an MES was analyzed systematically. Second, the algorithm of detailed scheduling for collaborative tasks was studied, and a model of external collaborative resources was designed. Then, a solution of collaborative scheduling was presented based on a hybrid genetic algorithm. Finally, a solution for the execution process management of collaborative manufacturing was promoted based on the integration of MES and ERP. The technologies researched can improve the general management of collaborative manufacturing and guide practical production management.

Keywords. Manufacturing execution system, collaborative manufacturing, hybrid genetic algorithm, task scheduling, task monitoring.

1. Introduction

Modern manufacturing enterprises specialize in an increasingly fine division of labor [1–3]. In collaborative manufacturing, a complete production task chain will be decomposed into mutually constrained orders to be executed by different enterprises [4-5], which significantly increases the production complexity and raises the difficulty of managing and executing manufacturing tasks. In response to the demand for collaborative manufacturing, the Collaborative Manufacturing Execution System (C-MES) has been developed based on the traditional MES. The characteristics of C-MES were described by MESA [6] in 2004. In 2005, Jimenez G et al. proposed a web-service-based form of C-MES architecture [7], which uses web service technology to collaborate and integrate different functional modules and systems to implement C-MES. At present, research on C-MES has focused on system construction [8], especially on how to use various web service technologies to realize the architecture and integration of MES in different enterprises. However, there is little research on collaborative manufacturing, namely, how to optimize production processes and improve production efficiency in enterprises based on ensuring the successful completion of collaborative tasks [9]. Therefore, this paper investigates how to use C-MES to efficiently plan and schedule collaborative tasks in discrete manufacturing enterprises, especially those with operation-level production task collaboration needs and strong technical capabilities, where many machining tasks

[1] Corresponding Author, Tongming XU, Inspur General Software Co., Ltd. Jinan, Shandong, China; E-mail: tongminghust@gmail.com.

can be completed by themselves or collaboratively. The scheduling in C-MES first needs to consider the capacity and other conditions to decide which operations need to be collaborative. Then, based on the optimization objectives and constraints, consider the indicators of collaborative vendors, reasonably select collaborative resources (CR) to perform collaborative tasks, complete process-level rolling scheduling, and realize the coordination and unification of process-level planning inside and outside the factory.

2. Detailed Production Planning Scheduling for Collaborative Manufacturing Tasks Based on MES

The study of detailed production planning scheduling for collaborative tasks in this paper is a follow-up study to the research work described in the literature [10]. This literature addresses a hybrid genetic algorithm-based detailed production planning scheduling algorithm and the design and implementation. This paper builds on the findings of that literature and investigates the use of hybrid genetic algorithms to implement detailed production planning scheduling for collaborative tasks.

2.1. Detailed Production Planning Scheduling Problem for Collaborative Manufacturing Tasks

In a collaborative manufacturing environment, the production planning scheduling problem is essentially a flexible job-shop scheduling problem (FJSP) [11]. In the process-level job operation, $\{O_{ih}|i = 1,2,\dots,k; h = 1,2,3\dots n\}$ for each order J_i of the production order set $\{J_i|i = 1,2,\dots,k\}$ of the same process route (h represents the process number and n is the maximum process number in the process template corresponding to the order J_i). Each process O_{ih} can have m_h master resources $\{R_{hq}|q = 1,2,\dots,m_h\}$ that can be executed; there are several collaborative tasks $\{O_{ih}|i = 1,2,\dots,k; 1 <= h <= n)\}$ in the process, each of which can be performed by external resources $\{CR_{hq}|q = 1,2,\dots,m_h\}$(CR, Collaborative Resource), as shown in figure 1 (in figure 1, a collaborative section consists of a group of collaborative processes in which the starting process task and ending process task are $O_{i(j+1)}$ and $O_{i(j+s)}$). For collaborative demand-side enterprises, the main tasks of scheduling are as follows: first, deciding which process task needs to be collaborative based on capacity and cost factors; then, selecting external collaborative resources for each collaborative task with comprehensive consideration of price, quality and efficiency while allocating the execution resources of this enterprise for the self-made operations; and finally, determining the planned start and planned completion time for each work operation according to the limited capacity constraint. The scheduling process should be optimized as much as possible for predefined processing indicators. When selecting collaborative resources, their processing efficiency issues should be considered, and various relevant factors of the enterprise they belong to are as follows: 1) completion time of collaborative tasks; 2) production quality of collaborative tasks; 3) manufacturing cost of collaborative tasks; 4) collaborative information interaction capability; 5) technical security and confidentiality of collaborative processes; 6) collaborative resource assurance capability; and 7) collaborative knowledge accumulation and sustainable development capability.

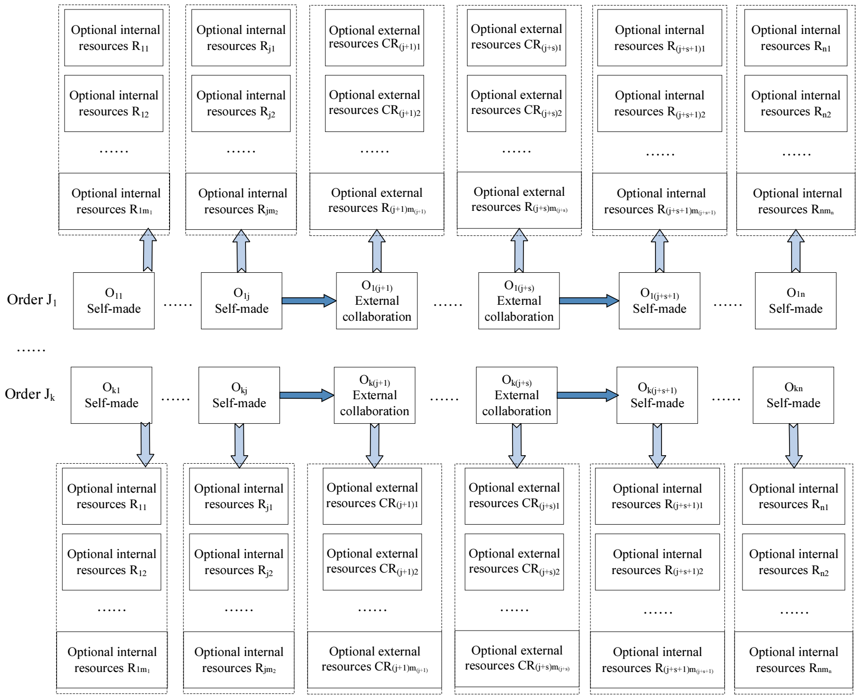

Figure 1. Collaborative manufacturing task scheduling problem description.

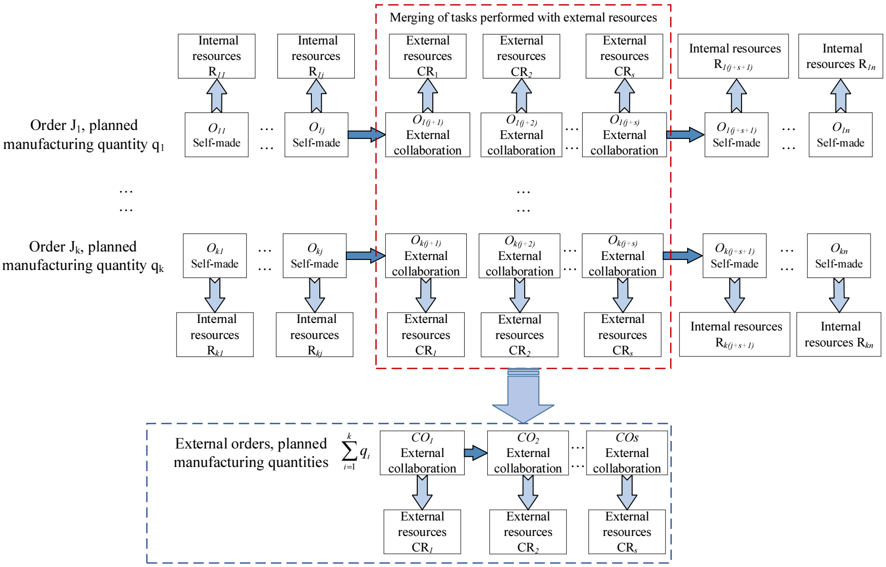

Figure 2. Consolidation of collaborative tasks and order generation for the same process line orders.

In a collaborative manufacturing system, the recipient of collaborative tasks generally does not expose the processing information of a specific manufacturing resource (e.g., station/equipment) to the outside but rather abstracts and encapsulates all the manufacturing resources of a process/section and accepts collaborative tasks as process/section collaborative resources, while the accepted tasks often have minimum lot requirements. Therefore, when collaborative, the tasks of the same process need to be combined by lot and then given to the collaborative resource for execution. As shown in the example in figure 2, a batch of orders with the same process route $\{J_i | i = 1,2, \ldots, k\}$, where the manufacturing quantity of each order J_i is q_i, and the collaborative process segment is $\{O_{ih} | h = j+1, \ldots, j+s\}$. The workstations or equipment resources of external vendors that can produce each process O_{ih} are abstracted in the collaborative system as a process-level collaborative resource with the set $\{CR_h | CR_h = \bigcup_{q=1}^{m_h} CR_{hq}\}$. The collaborative operations of the same process route and the process can be combined to form a manufacturing quantity of $\sum q_i$ and a process set of $\{CO_h | CO_h = \bigcup_{i=1}^{k} O_{ih}\}$ of collaborative orders, which are delivered to collaborative manufacturing resources for execution.

3. Modeling of External Collaborative Manufacturing Resources in MES Planning and Scheduling

The collaborative manufacturing resources are a collection of physical or conceptual objects that are physically and informatively encapsulated to provide specific manufacturing capabilities for the external application requirements of an enterprise. Its model is shown in figure 3. In a collaborative manufacturing environment, the demander of a collaborative manufacturing task can only access and obtain information about the encapsulated CMRs of the enterprise of the task receiver but not the specific actual manufacturing resources.

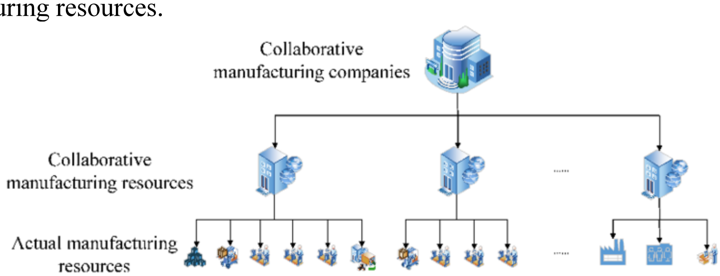

Figure 3. Collaborative manufacturing resource model.

A detailed examination of the definition and characteristics of collaborative manufacturing resources shows the following commonalities:

(1) It can provide the production capacity of various products per unit of time and provide feedback on the completed quantity and quality information of currently accepted tasks.

(2) For the same product processing efficiency, the unit price is related to the urgency of the delivery date and the production lot. There is a minimum production lot limit for the same batch of collaborative orders, and generally, the larger the lot is, the higher the efficiency and the lower the unit price.

(3) Under the same collaborative manufacturing resource, after accepting a batch of processing tasks that occupy all capacity over a period of time, it cannot accept other processing tasks within this period.

Based on the above characteristics, a collaborative resource model is proposed in this paper. Compared with ordinary resources, collaborative resources encapsulate the following unique attributes and business methods.

(1)For a specific product, each collaborative manufacturing resource has multiple production modes, and each production mode has the necessary minimum and maximum processing quantities, processing work hour information (single work hour or total work hour, with the work hour unit information determining the work hour calculation method) and price information (again, with the price unit information determining the price calculation method).

(2)If possible, many similar process tasks must be performed simultaneously to satisfy the minimum/maximum number of processes. Similar operations are defined as tasks whose optional resources contain the same collaborative resources; these tasks require the same production cycle and other process parameters for processing from the collaborative resources.

(3)The production evaluation attributes of the resource are obtained from the relevant management system of the upper level of the enterprise, such as enterprise resource planning (ERP).

From the above model, the scheduling of collaborative tasks can be achieved by changing the task execution logic of the collaborative resource in the decoding algorithm based on the scheduling algorithm in the literature [10]. When the decoding algorithm simulates the task execution of the collaborative resource, it will combine similar frame tasks according to the processing lot corresponding to the selected production mode and jointly occupy a period of processing time of this collaborative resource. Again, the planned start and completion times are calculated based on the work time information on the process route and the resource production calendar.

4. Collaborative Manufacturing Task Scheduling Based on a Hybrid Genetic Algorithm

The main flow of collaborative manufacturing task scheduling is shown in figure 4. Presheduling is used to decide which collaborative tasks need to be handed over to collaborative resources based on capacity and select collaborative resources. Formal scheduling then calculates the process-level production plan in detail based on the prescheduling results. The technical details related to scheduling are described in detail below.

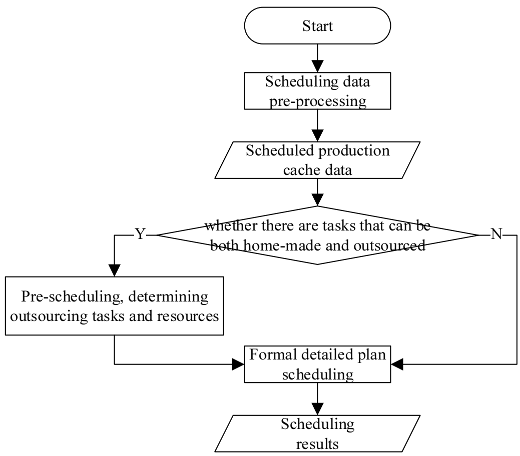

Figure 4. Collaborative manufacturing task scheduling master process.

4.1. Collaborative Manufacturing Task Scheduling Based on a Hybrid Genetic Algorithm

In this paper, the hybrid genetic algorithm-based scheduling algorithm described in the literature [10] is employed to solve the scheduling problem for collaborative manufacturing tasks. The steps of the algorithm have been detailed in the literature, and this section only describes the technical details of the algorithm for the application of collaborative manufacturing task scheduling.

4.1.1. Implementation of a Combined Batch of Process Tasks on External Collaborative Resources in the Decoding Algorithm

The chromosome decoding algorithm of the scheduling algorithm described in the literature [10] is as follows: first, the available time slot records of all resources are generated, and then the planned start and completion times of each task are calculated sequentially within the workable time of the execution resource of each task in the same order as the OS (Operation Sequence) within the chromosome, which satisfies the sequence of processes in the technological procedure. After determining the planned time of each operation, the period of the corresponding execution resource occupied by the operation is removed from the available period record of that resource. The algorithm considers only the production business of common production resources. Based on the above definition of external collaborative resources described, two problems need to be solved to calculate the planned start and finish times of each process task on them. One is the creation of the batches consisting of several similar process tasks that are given to the same external collaborative resource for processing. The other constraint satisfies the constraints of the minimum and maximum processing quantities for external collaborative resources as much as possible unless there are not enough processing quantities of operations for scheduling. Therefore, this paper designs the following decoding algorithm for production tasks on external collaborative resources (described as an example of forwarding scheduling).

(1) Let the planned production quantity of the operation O_{ij} (i is the order number and j is the process number) calculate the planning time to be q . The minimum/maximum processing quantity of the selected external collaborative resource is q_{rmin}/q_{rmax}, and the earliest available start time of task T_{es} is determined for ordinary operations.

(2) Find a batch of process tasks of the same kind as O_{ij} that have been assigned to the external collaborative resource with a scheduled start time before T_{es} and the closest scheduled start time to T_{es}. With a total processing quantity of q_b satisfying $q_b < q_{rmin}$ and $q_b + q <= q_{rmax}$, if they exist, record this batch as T_b and perform steps 3 and 4. If they do not exist, then find a batch of process tasks of the same kind as O_{ij} whose scheduled queue start time is before T_{es} and the scheduled start time is nearest to T_{es}, q_b satisfies and $q_{rmin} <= q_b + q <= q_{rmax}$. If it exists, record this batch of process tasks as T_b and execute steps 3 and 4; if it still does not exist, execute step 3 only.

(3) If there is an eligible set T_b of similar process tasks in step 2, the processing period occupied by T_b on this furnace resource is released, and the set T_r of free periods of this furnace resource is updated. The planned start and completion times T_{ps} and T_{pf} of O_i are calculated based on T_{es} according to the calculation method mentioned above of the planned time of the process task on ordinary resources.

(4) The scheduled start and completion times of the process task in the eligible set T_b in step 2 are set to the T_{ps} and T_{pf} calculated in step 3.

As seen from the above algorithm, the planning time of scheduled operations on the same resource may be adjusted during the planning time calculation of operations on external collaborative resources because of the need to merge adjacent similar tasks to meet the maximum and minimum production lot constraints. Therefore, to ensure that the decoding algorithm works properly, $O_{i(j+1)}$ must be after O_{ij} and O_{ij}, similar tasks in the shop floor task sequence. The specific approach is that in the scheduling data preprocessing stage, the order with similar outside operations O_{ij} as the node is split into two suborders, and in addition to the regular front task O_{ij}, all similar operations of O_{ij} will also become particular front tasks of $O_{i(j+1)}$. (This pretask will not constrain the planning time of $O_{i(j+1)}$ but only its position in the shop (this pretask will not constrain the planning time of $O_{i(j+1)}$ but only its position in the sequence of tasks on the shop floor, i.e., its order of resource selection). The subsequent scheduling will be based on the split suborders.

4.1.2. Breakdown of Collaborative Task Sections and Orders Containing Collaborative Tasks

In the collaborative manufacturing process, a group of collaborative operations in a batch of orders (usually consecutive) are often given to collaborative resources as a whole, and the collaborative demander monitors the overall progress and quality of these collaborative tasks. This group of collaborative operations is called a collaborative section (CS). Once an order has a collaborative process designated as a collaborative task, the order needs to be broken down into multiple suborders based on the collaborative section before participating in scheduling. As shown in figure 5, $CS_{k2}, CS_{k4}, \ldots, CS_{kn}, \ldots$ (k represents the order number and n represents the process section number) are the respective collaborative sections on order Jk, where each collaborative section CS_{kn} is composed of multiple adjacent collaborative assignment work orders. If both collaborative sections belong to the same group of consecutive process tasks under the same process template, the two collaborative sections are said to be the same kind of collaborative section.

Figure 5 also shows the decomposition process of orders containing collaborative tasks. An order Jk containing collaborative tasks can be split into a set of suborders $\{J_{k1}, CS_{k2}, \ldots, CS_{k(r-1)}, J_{kr}\}$ according to the location of the collaborative section within the order. J_{k1} has the same earliest possible start time (EPST) as J_k, and J_{kr} has the same latest possible finish time (LPFT) as J_k. The last process task of J_{ki} (or CS_{ki}) is the pretask of the first task of $CS_{k(i+1)}$ (or $Jk_{(i+1)}$). In addition, the last process tasks of all similar collaborative sections of CS_{ki} shall be set as the former tasks of the first task of $J_{k(i+1)}$. In the same way, the former tasks of all similar collaborative sections of CS_{ki} shall be set as the former tasks of the first task of CS_{ki}.

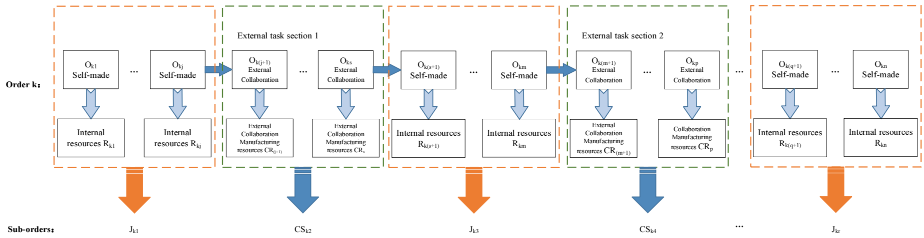

Figure 5. Breakdown of collaborative task sections and orders containing collaborative tasks.

4.1.3. Decoding Algorithms for Collaborative Manufacturing Tasks

As mentioned earlier, collaborative sections in the decoding algorithm should be considered a whole shop floor task for scheduling. On the other hand, the same collaborative manufacturing resource has multiple production modes with different lot sizes. Different batch production modes have various production difficulties for collaborative resources (in general, the smaller the batch is, the greater the production and management difficulties), so the production unit price and production efficiency of different batch production modes may differ. As shown in figure 6, four orders with the same process route and the same external collaborative resources are selected, and the buffer time between the self-production tasks and the collaborative tasks is 16 hours. The scheduling result obtained by assigning tasks to the CR with only 100-150 batches will cause the order J3 to be overdue, while a mix of two production modes, combining CS_{12} and CS_{22}, CS_{32} and CS_{42} into two sets of tasks to be executed by the CR, will result in a more satisfactory scheduling result. In the following, this example will be presented as a decoding algorithm for collaborative manufacturing scheduling, considering the impact of different batch production modes.

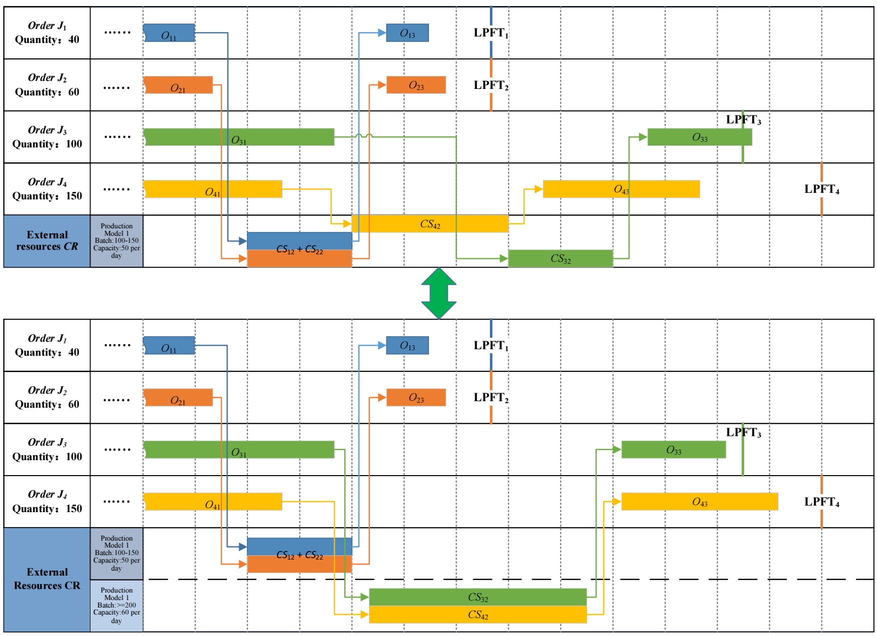

Figure 6. The impact of different batch production patterns of collaborative manufacturing resources on scheduling results.

Four orders are set, and the process templates are all PR: $P_1 - CS_2 - P_3$, where CS_2 is a collaborative section, and P_1 and P_3 are self-made processes. Then, let these four orders be $J_i = O_{i1}, GD_{i2}, O_{i3}$ and $i = 1,2,3,4$ for the order code identification. Suppose that the external collaborative manufacturing resource CR is selected for all the external segments of these four orders in a particular chromosome individual. According to the preceding, in the sequence of shop floor tasks of this chromosome, CS_{i2} is arranged after all O_{i1} and O_{i3} is arranged after all CS_{i2}. An order will be divided into three suborders for processing in scheduling. An example of a chromosome containing collaborative tasks is shown in table 1 (where OS is the process execution sequence, RS is the resource selection sequence, and each gene position in RS indicates the resource selected by the process at the corresponding position in OS). The decoding process is as follows.

(1) According to the previous algorithm, decoding O_{21}, O_{31}, O_{11}, and O_{41} generally yields the corresponding scheduled end times T_{f21}, T_{f11}, T_{f31}, and T_{f41}, as shown in figure 7.

(2) Since the execution of the external segment does not occupy the resources of this enterprise, CS_{12} CS_{42} are skipped for the time being, and the decoding of other gene bits is continued until the decoding reaches the position of O_{13}. At this point, based on the occupied time on T_{21}, T_{11}, T_{31}, T_{41}, and internal resources R_{13}, R_{23}, R_{33}, R_{43}, the earliest available start time without considering related external tasks is calculated and set as T'_{ES23}, T'_{ES13}, T'_{ES33}, T'_{ES43}, and the results are shown in figure 7. Two virtual inventory caches correspond to the process template PR, the collaborative section CS_2 and the collaborative resource CR, referred to as the self-made - collaborative inventory and the collaborative-self-made inventory, which correspond to the processes under the process template PR obtained from the enterprise by the collaborative resource CR, respectively. The inventory of the output of the work operation under P1 and the inventory of the output of CS_2 under the process template PR obtained by the enterprise from CR. Based on the planned end time of the tasks before each synergistic section, we establish the time sequence of self-made - collaborative inventory stock quantity. Two virtual inventory caches correspond to process template PR, collaborative section CS_2 and collaborative resource CR, referred to as self-made - collaborative inventory and collaborative - self-made inventory, which correspond to the output inventory of the process task under process P_1 under process template PR obtained by collaborative resource CR from the enterprise and the output inventory of section CS_2 under process template PR obtained by the enterprise from CR. Based on the planned end time of the tasks before each synergistic section obtained earlier, the self-made - collaborative inventory stock quantity time series is established.

(3) Sort T'_{ESi3} in chronological order, select T'_{ESk3} andO_{k3} of the corresponding order in demand from early to late and let the manufacturing quantity of J_k be q_k. If the inventory quantity q_s at the moment of T'_{ESk3} in the collaborative-self-made inventory is less than q_k, then add a collaborative section task CS_{CRk} on CR consistent with CS_2. Let its planned start time be T_{CSk}; then, its processing quantity qCk is the inventory quantity in the self-made - collaborative inventory at the moment of T_{CSk}. It is required that qCk is as small as possible and T_{CSk} is as early as possible while satisfying $q_{ck} + q_s >= q_k$ and meeting CR's minimum production lot requirement. The planned end time of CS_{CRk} is calculated based on the determined order parameters, which in turn determines the collaborative-self-made inventory constraint T_{ESk3} on the earliest start time of O_{k3}. If $T_{ESk3} <= T'_{ESk3}$, try to delay T_{CSk} to the next moment when the inventory quantity in the self-made - collaborative inventory is more extensive and repeat the above process until

$T_{ESk3} > T'_{ESk3}$ or qCk has reached the maximum. Select the T_{CSk} that makes $T_{ESk3} <= T'_{ESk3}$ (or the T_{CSk} that makes T_{ESk3} the smallest if there is none) and the corresponding qCk that is as large as possible as the parameter of the final order CS_{CRk}, and refresh the self-made - collaborative inventory and collaborative - self-made inventory records. The detailed steps are shown in figure 7.

(4) Repeat step 3 until all O_{i3} decoding is completed, replace the newly generated coworker task CS_{CRk} with CS_{i2} in the original chromosome according to the succession relationship, and adjust the corresponding task succession relationship; the detailed steps are shown in figure 7.

Table 1. Example of a chromosome containing a collaborative task.

OS …	O_{21} O_{11}	…	O_{31} O_{41}	…	CS_{12}	…	CS_{22} CS_{42}	…	CS_{32}	…	O_{13}	…	O_{33} O_{43} O_{23}	…
RS …	R_{21} R_{11}	…	R_{31} R_{41}	…	CR	…	CR CR	…	CR	…	R_{13}	…	R_{33} R_{43} R_{23}	…

From the above decoding algorithm, it is known that the collaborative section of each order containing collaborative tasks may be split and merged during the decoding process to form new collaborative tasks that are not consistent with the manufacturing quantity of the original order, and these collaborative tasks will be the basis for determining the collaborative orders afterward. The parts before the tasks before and after the subsequent tasks of the collaborative section of the original order will be split into their own relatively independent orders. The constraint of the start time of the following tasks of the collaborative section is determined by the inventory of the manufactured goods of the corresponding collaborative section, which is consistent with managing the collaborative tasks of processes in the actual production of discrete manufacturing enterprises. The logic of determining each production lot of collaborative resources in the decoding algorithm is to select the maximum lot that has the most negligible impact on the start time of the subsequent self-production tasks in a greedy way to try to balance the productivity of the self-production operations and the collaborative cost and management cost of the collaborative tasks.

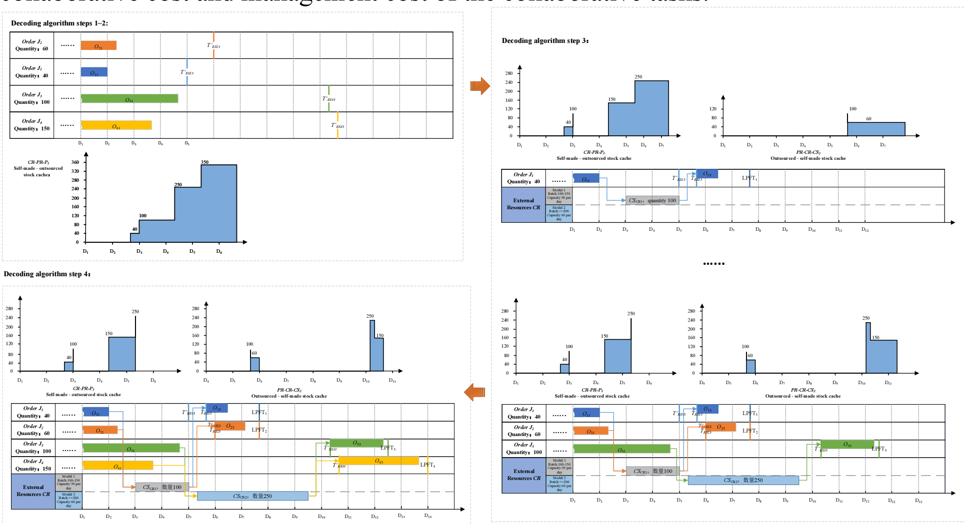

Figure 7. Chromosome decoding with collaborative tasks.

4.2. Determination of Prescheduled Production and Collaborative Tasks

The purpose of prescheduling is to decide which collaborative tasks need to be collaborative and select collaborative resources based on the actual workshop capacity. The general idea of prescheduling is first to set all the operations that are not sure whether to collaborate as self-made tasks for scheduling, check whether the company's production capacity can meet the order, and if not enough, set some of the collaborative tasks on the bottleneck manufacturing resources as collaborative tasks for rescheduling, and then repeat the process until the scheduling results meet the established order delivery requirements. In complex cases, prescheduling may require experienced schedulers to participate in the iterative scheduling process, modifying the collaborative task settings, and rescheduling.

Table 2. Definition of data related to the calculation of the overall fitness of collaborative resources

order	Order Priority	Collaborative section	Priority of the selected external synergy resources
J_1	P_1	CS_{11}	w_{11}
		CS_{12}	w_{21}
		…	…
…	…	CS_{1r1}	w_{1r1}
		…	…
		CS_{k1}	w_{k1}
J_k	P_k	CS_{k2}	w_{k2}
		…	…
		CS_{krk}	w_{krk}

The optimization objectives based on the selection of collaborative resources are the number of on-time orders q_o, the total extension time of backorders t_d, the overall fit of collaborative manufacturing resources w_r, and the collaborative processing cost C. Among them, the priority of the first two is greater than the collaborative processing cost C. Table 2 defines the relevant data for the calculation of the overall fit of collaborative manufacturing resources if we set w_{kj} as the collaborative priority obtained by hierarchical analysis, r_k as the number of collaborative sections, and P_k as the collaborative processing cost. Resource priority, r_k is the number of collaborative sections, and P_k is the order priority. Then, the overall fit of collaborative manufacturing resources is defined as shown in equation (1).

$$x = \sum_{k=1}^{n}\left[P_k\left(\frac{\sum_{j=1}^{r_k} w_{kj}}{r_k}\right)\right] \tag{1}$$

4.3. Formal Detailed Planning of Production Scheduling

After the production plan of the collaborative task is determined, the detailed planning scheduling must be started to generate the formal production plan. In formal scheduling, since the collaborative tasks and corresponding collaborative resources have been determined, the main goal of formal scheduling is to optimize the production plan of the self-production tasks according to the required optimization objectives and to handle the interface between the self-production tasks and the collaborative tasks. The collaborative tasks and the corresponding collaborative resources will be fixed in the standard scheduling algorithm. At the same time, other collaborative tasks that are not included in the collaborative tasks will be treated as ordinary homemade operations and made to choose only the internal production resources as the execution resources. Other details of the algorithm are not discussed here.

5. Conclusion

This paper discusses the problem of collaborative manufacturing planning management at the level in discrete manufacturing enterprises in the context of collaborative manufacturing, proposes a collaborative manufacturing production planning and process control scheme based on MES manufacturing data and business integration, and illustrates the necessary technical details of the planning scheduling and execution process exception handling process of collaborative manufacturing tasks. By analyzing the business characteristics of external coproduction, a modeling method of external collaborative resources is proposed, and the model is combined with a hybrid genetic algorithm to design a hybrid genetic algorithm considering the external coproduction business to solve the process-level coproduction scheduling problem. Based on MES production exception management and the integration of manufacturing data with ERP and other systems, a solution for collaborative production process monitoring is proposed, with specific guiding significance for the digital transformation of discrete manufacturing enterprises. With the development trend of intelligent production operation management systems, subsequent research will focus on optimizing scheduling algorithms and implementing intelligent production scheduling designed to improve the intelligence level of production operation management.

References

[1] Sundar R, Balaji A N, Kumar R M S. A review on lean manufacturing implementation techniques[J]. Procedia Engineering, 2014, 97: 1875-1885. https://doi.org/10.1016/j.proeng.2014.12.341.

[2] Zhong R Y, Xu X, Klotz E, et al. Intelligent manufacturing in the context of industry 4.0: a review[J]. Engineering, 2017, 3(5): 616-630. doi:https:// doi.org/10.1016/J.ENG.2017.05.015.

[3] Zhou J, Li P, Zhou Y, et al. Toward new-generation intelligent manufacturing[J]. Engineering, 2018, 4(1): 11-20. https://doi.org/10.1016/j.eng.2018.01.002.

[4] Wang H, Liu L, Fei Y, et al. A collaborative manufacturing execution system oriented to discrete manufacturing enterprises[J]. Concurrent Engineering, 2016, 24(4): 330-343. https://doi.org/10.1177/1063293X16640591.

[5] Panetto H, Iung B, Ivanov D, et al. Challenges for the cyber-physical manufacturing enterprises of the future[J]. Annual Reviews in Control, 2019, 47: 200-213. doi:https://doi.org/10.1016/j. arcontrol.2019.02.002.

[6] Internationa M. Collaborative manufacturing explained. MESA International—White Paper[J]. 2004. http://www. mesa. org.

[7] Jimenez G, Molina A, Canche L. Manufacturing Execution Systems Interoperability and Web Services Technologies[C]//ASME International Mechanical Engineering Congress and Exposition. 2005, 42142: 191-197. doi:https://doi.org/10.1115/IMECE2005-82330.

[8] Younus M, Peiyong C, Hu L, et al. MES development and significant applications in manufacturing-A review[C]//2010 2nd International Conference on Education Technology and Computer. IEEE, 2010, 5: V5-97-V5-101. doi:10.1109/ICETC.2010.5530040.

[9] Wang X, Zhang C, Jin Y, et al. CPSP: A cloud-based production service platform supporting co-manufacturing of cross-enterprise[C]//2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design ((CSCWD)). IEEE, 2018: 455-460. doi:10.1109/CSCWD.2018.8465354.

[10] Zhou L, Chen Z, Chen S. An effective detailed operation scheduling in MES based on hybrid genetic algorithm[J]. Journal of Intelligent Manufacturing, 2018, 29(1): 135-153. doi:https://doi.org/10.1007/s10845-015-1097-6.

[11] Zhang G, Gao L, Shi Y, An effective genetic algorithm for the flexible job-shop scheduling problem, Expert Systems with Applications, 2011, 38 (4): 3563–3573. doi:https://doi.org/10.1016/j.eswa.2010.08.145.