

Research Article

An Optimal Scheduling Method for Data Resources of Production Process Based on Multicommunity Collaborative Search Algorithm

Yanlei Yin ¹, Lihua Wang ¹, Jun Tang,² Wanda Zhang ¹ and Hongwei Niu³

¹Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming, China 650500

²Technology Center, China Tobacco Yunnan Industrial Co. Ltd., 650233, China

³Faculty of Machinery and Vehicles, Beijing Institute of Technology, 100081, China

Correspondence should be addressed to Lihua Wang; kmwanglihua@163.com

Received 28 October 2021; Accepted 23 February 2022; Published 12 March 2022

Academic Editor: Xue-bo Jin

Copyright © 2022 Yin Yanlei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Aiming at the problem of low response speed and unbalanced distribution of data resources of production process (DRPP) for the distributed workshop production environment, an optimization scheduling method of DRPP based on a multicommunity cooperative search algorithm is proposed. A heuristic data resource service scheduling framework including a load manager and dynamic scheduling engine is first built to deal with the uncertainty of data resource service response and the imbalance of resource allocation; a core scheduling optimization mathematical model with the objectives: resource service efficiency, reduced response time, and load balancing, is established. Then, a multicommunity cooperative search algorithm for the scheduling model is presented, and the mapping relationship between the particle position vector and resource allocation is established via binary coding. Thus, the optimization algorithm is mapped to discrete data space, and the multicommunity bidirectional driving evolutionary mechanism is used to realize the cooperative and interactive search between common and model community, which enhances the adaptability of the algorithm to dynamic random scheduling tasks. Finally, the effectiveness of the proposed method is verified by an example of multiprocess quality prediction service scheduling in silk production process, which provides an effective means for solving the complex scheduling problem of production process data.

1. Introduction

The intelligent workshop integrates modern sensing technology, network technology, automation technology, and other advanced technologies, and a large number of intelligent equipment such as sensors and data acquisition devices have been put into use in the workshop [1–3]. And thus, the production workshop has become the collection center of information flow, material flow, and control flow. In the process of product production, a large amount of production, environment, status, and equipment operation data are generated at an unprecedented speed, thus forming workshop big data, which presents the new characteristics of multitask, cross-process, heterogeneous, and polymorphic. However, data has no subjective initiative. To realize the real-time perception and prediction of the production

process, we must first realize the scheduling and optimization of “data,” that is, the scheduling and optimization of production process data. It is noted that the traditional manufacturing mode, data of information flow, material flow, and control flow are still isolated from each other in each stage of production execution, and it is difficult to form a joint force due to the lack of effective data resource scheduling mechanism, which restricts the further improvement of production efficiency and system intelligence level [4]. Therefore, the research on on-demand scheduling of workshop data resources is one of the core problems of intelligent manufacturing in production workshop.

Recently, most scholars at home and abroad focus on the methods and algorithms of workshop data collection, analysis and mining, such as machine vision preprocessing algorithm [5], neural network prediction algorithm [6, 7],

intelligent decision algorithm [8], and multiobjective optimization algorithm [9]. However, the data have no subjective initiative; the data-based analysis and processing algorithm can not actively serve the business needs such as perception, decision-making, and execution of the production process; and the current research has not comprehensively considered the coupling and impact among demand, service, resources, and energy efficiency in the production process.

However, the research of domestic and foreign scholars on workshop process production scheduling and collaborative job scheduling has a good reference for the development of this work. Considering workshop process and production scheduling, literature [10] based on the dependence of the production planning, and scheduling problem of continuous production line on timing, a repair and optimization solution is proposed to solve the problem of energy efficiency in the production process. For the scheduling problem of complex products in multiworkshop production, literature [11] studies the characteristics of BOM structure and process route of complex products. Based on the construction of multilevel process network diagram, an improved particle swarm optimization algorithm is used to ensure the effectiveness of the algorithm search path. Literature [12] considers the problems of process connection and blocking of prefabricated parts in the process of workshop assembly line operation and establishes a scheduling model to minimize the total penalty cost of advance and delay, which improves the production efficiency of Prefabrication Yard. Literature [13] uses a machine learning method to assign jobs based on the priority rules of the decision tree as the scheduling method, which shows good performance in the case scenario with completion goal and total delay goal. Considering from workshop production collaborative job scheduling, Literature [14] estimates the process processing time in the production process through machine learning and uses the estimated processing time to schedule and optimize parallel machines, which reduces the maximum completion time by about 30% on average. Aiming at the optimal comprehensive production and transportation plan of a group of parallel batch machines, literature [15] constructs a 0-1 mixed integer programming model, solves the model, and completes the comprehensive scheduling through an improved genetic algorithm, which reduces the transportation cost. The above research provides an idea for this paper to realize on-demand scheduling data resources for production process.

On the other hand, it should be noted that the workshop production process involves multiprocess cross-production line business collaboration and business requirements. In the process of data resource scheduling, we should not only consider the association and cooperation relationship between different production tasks but also consider the transmission time of data resources between different production tasks, in particular, the uncertainty of concurrent service access affecting demand response, and the impact of these dynamic and uncertain factors on the balance of data resource allocation [16–18].

This paper is concerned with production-oriented data resources scheduling, thus transforming DRPP into a service, and finally into economic benefits. Consequently, this study is to integrate the load manager and dynamic task scheduling engine, and combine them with scheduling processes to form a scheduling scheme, so as to provide intelligent support for production process. Following this idea, this paper is organized as follows: the heuristic data resource service scheduling framework is constructed in Section 1. The problem to be studied and the scheduling mathematical model are proposed in Section 2. Section 3 is devoted to establish the asynchronous parallel scheduling strategy and optimization method, simulation results are given in Section 4, and conclusions are made in Section 5.

2. DRPP Scheduling Process Analysis

According to the execution status of the DRPP in the scheduling center, the data in the resource pool are mobilized to form the optimal execution scheme of tasks. DRPP scheduling is one of the key links in the production decision process. Figure 1 presents a framework of DRPP scheduling that includes decomposition of business requirements, service task analysis, dynamic scheduling of DRPP, load monitoring, and service task execution. Firstly, during production operation, different processes send requests to the scheduling center according to task execution requirements. By queuing, merging, and analyzing the service requirements, the task analyzer degrades the vague and miscellaneous service tasks to form a set of low-granularity service tasks that can be directly served by DRPP. Secondly, the dynamic scheduler preliminarily matches the DRPP according to the task request and then matches the execution task characteristics with the static and real-time attributes of the DRPP to obtain the DRPP set that meets the current production business requirements. The load supervisor of the scheduling center dynamically adjusts the DRPP by monitoring the operation and load of the DRPP in the business process and solves the service interruption caused by uncertain events to ensure the accomplishment of the service process. Finally, the DRPP scheduling engine uses the integrated intelligent optimization algorithm to combine and match the state information of the DRPP and the real-time information of the service task to form the optimal resource service allocation scheme and submits it to the center for execution, so as to complete the scheduling process of the resource service.

Throughout the entire scheduling process, multiservice tasks are executed interactively, and there is a complex relationship between tasks. With the dynamic growth of production business scale, the response time of service tasks must be considered. Additionally, a large number of dynamic and uncertain factors will seriously affect the ability and effectiveness of DRPP service scheduling. The traditional resource scheduling method has low search efficiency and accuracy, which can easily lead to the problems of low response speed and the uneven distribution of resources in the service process. It is difficult to adapt to the allocation of STRs on demand. Therefore, this paper considers

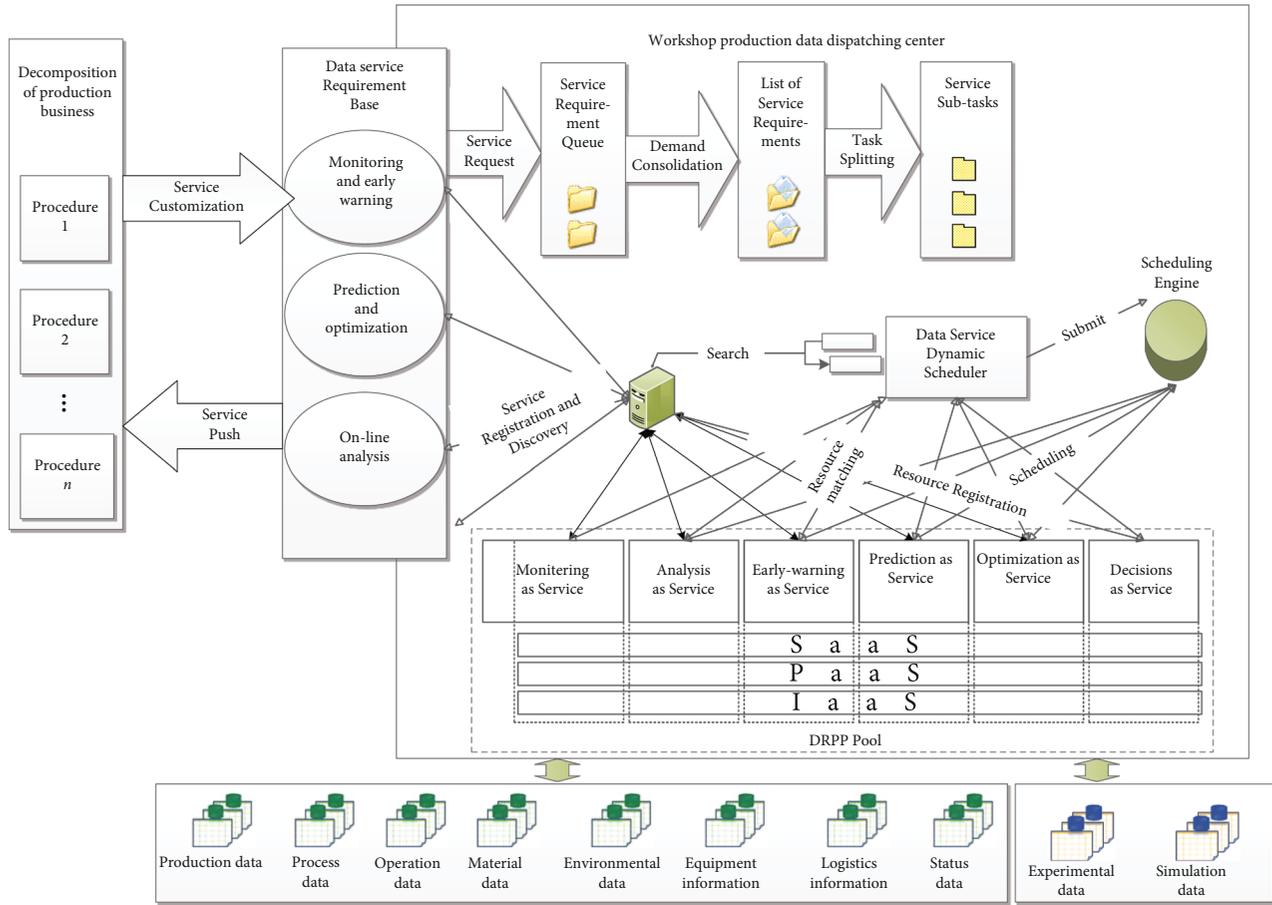


FIGURE 1: Scheduling process of DRPP.

improving service efficiency while solving the problem of the unreasonable allocation of DRPP caused by the uncertainty of service response and the unbalanced load of nodes.

3. The Multiobjective Optimal Scheduling Model of DRPP

3.1. The Response Uncertainty Modeling. The invocation relationship of DRPP in the business process is complex, especially when a data resource service is invoked by multiple business processes, and these business processes run simultaneously; there will be concurrent access. In different service scenarios, there are some fluctuations and uncertainties in the access frequency, concurrency probability, and response time of users in the entity industry. There are certain fluctuations and uncertainties in the access frequency, concurrency probability, and response time of different business links in the production process, such as sequence, selection, parallelism, and the cycle of business processes [19, 20]. Therefore, the uncertainty of the service response of DRPP can be described by service access frequency.

In the following formula, k represents the number of service tasks. l represents the total DRPP resources. p represents the process of any service tasks. θ represents the probability of performing service task i . μ represents the probability of

invoking DRPP j for the service task i . ε represents the probability of a service business being accessed. γ represents the probability of any process branch being selected.

Assuming that there are j DRPP for l subtasks to call when the scheduling center performs a certain service task, and these subtasks are completed on a specific node according to the service process; then, the probability that the process of the service task is executed is

$$\varepsilon = \sum_{j=1}^l \sum_{i=1}^k \theta \mu \quad 1 \leq i \leq l \leq j \leq J. \quad (1)$$

When multiservice tasks are executed interactively, concurrent access often occurs in service invocation. Accomplishing a task involves invoking multiple resource service processes. When there is a selection structure in the process and a process branch covers concurrent access services, the probability of concurrent access to services is as follows:

$$P_{ca} = \sum_{p=1}^p \varepsilon \gamma \quad 1 \leq p \leq L. \quad (2)$$

When all the selected branches in the service process cover the current concurrent access service, the probability

of concurrent access to the services is

$$P_{ca} = \sum_{p=1}^p \varepsilon \quad 1 \leq p \leq L. \quad (3)$$

3.2. The Modeling of Unbalanced Resource Allocation. Suppose a service task $\Omega = \{t_1, t_2, \dots, t_n\}$ needs to call k subtasks to complete, the total number of the DRPP that can provide services is l . These subtasks are completed in different tasks node according to the service process and resource requirements. The expected completion time of task t_i invoking technology resource r_j is defined as

$$T_{ij} = \frac{IL_i}{ES_j}, \quad (4)$$

where IL_i is the total instruction length of service task t_i and ES_j is the execution speed that the DRPP r_j are distributed invocation.

The average load of k different service tasks scheduling l DRPP is defined as the quotient between the total instruction length of k service tasks and the total execution speed of l data resources distributed scheduling, i.e., the completion time of the total service task is

$$Tt = \frac{\sum_{i=1}^k IL_i}{\sum_{j=1}^l ES_j}. \quad (5)$$

For the above scheduling scheme, the load balancing of service resources invoked can be defined as

$$\sigma = \sqrt{l^{-1} \sum_{j=1}^l (Tt_j - Tt)^2}, \quad (6)$$

where Tt_j represents the completion time of the service task j . Obviously, a smaller σ indicates a more balanced task load service scheduling.

3.3. The Multiobjective Optimal Scheduling Model. Considering the DRPP response and the unbalance of resource allocation, this paper established a multiobjective optimal scheduling mathematical model including the service efficiency, response time, and load balance of resource invocation.

In the production process, if e_j^i represents the state of service node i performing subtask j , N represents the subtasks, and M represents the total number of DRPP provided by service tasks; then, the state set of resource services is $E = (e_j^i)_{M \times N}$, $1 \leq i \leq M$, $1 \leq j \leq N$. The set of service efficiency is $Se = \{se_j^i\}_{M \times N}$, $1 \leq i \leq M$, $1 \leq j \leq N$, the response frequency set is $Rt = \{rt_j^i\}_{M \times N}$, $1 \leq i \leq M$, $1 \leq j \leq N$, and the load balancing set is $\delta = \{\sigma_j^i\}_{M \times N}$, $1 \leq i \leq M$, $1 \leq j \leq N$. In these equations, se_j^i is the service efficiency of service node i executing scheduling task j , rt_j^i is the response time of executing

resource scheduling task j for service node i , and σ_j^i is the service efficiency load balancing of resource scheduling task j for service node i . The service status of any resource service node executing a task can then be expressed as

$$e_j^i = \{se_j^i, rt_j^i, \sigma_j^i\}. \quad (7)$$

The set of DRPP mapped by N service tasks is $X = \{x_1, x_2, \dots, x_N\}$, where x is a DRPP invoked for a service subtask. Thus, the mathematical model of multiobjective optimal scheduling considering the uncertainty of service response and the imbalance of resource allocation is as follows:

$$\begin{cases} \min F(x) = (X, E) = (Se(x), Rt(x), \delta(x)), \\ x_j \in R \text{ and } x \leq M, \\ s.t. \quad se_j^i \geq se_{\min}, \\ 0 \leq rt_j^i \leq rt_{\max}, \\ 0 \leq \sigma_j^i \leq \sigma_{\max}, \end{cases} \quad (8)$$

where se_{\min} is the minimum service efficiency value in line with business requirements, rt_{\min} is the maximum service response time that a DRPP node can take, and σ_{\max} is the highest load balancing of a DRPP node.

Set the weights of the service efficiency, load balancing, and response frequency of user requesting DRPP as ω_{Se} , ω_{Rt} , and ω_{δ} , respectively, and $\omega_{Se} + \omega_{Rt} + \omega_{\delta} = 1$.

The optimal scheduling model of DRPP considering the target weight of user demand is as follows:

$$F(x) = \omega_{Se}Se(X) + \omega_{Rt}Rt(X) + \omega_{\delta}\delta(X). \quad (9)$$

4. The Optimal Scheduling Algorithms of DRPP Based on Multicommunity Collaborative Search

4.1. The Evolution Model of Multicommunity Cooperative Network. The basic particle swarm optimization (PSO) algorithm is a single-community optimization model with global optimal particles as its core, which cannot solve the mixed and changeable scheduling problem very well. If this model is extended to task-related multicommunity cooperative optimization, the evolutionary information interaction and association will be generated among these communities, and then, a multicommunity cooperation network (MCCN) with high adaptability to the task will be formed [21, 22]. From a mathematical point of view, a network can be regarded as a combination of a vertex set and edge set. To better describe MCCN and establish its evolution model, the following definitions are first provided.

Definition 1. The threshold FT for community type determination is

$$FT = \frac{\sum_{i=1}^n F_i}{n}, \quad (10)$$

where F_i is the global optimal fitness of community i and n is the number of communities in the cooperative network.

According to the threshold FT of community type determination, the particle community in the collaboration network can be divided into the model community and common community. If F_i satisfies the criterion $F_i \geq FT$, the community has a strong ability of local optimization, which can be divided into model communities and recorded as MC_i . On the contrary, if F_i satisfies the criterion $F_i < FT$, the community has a strong ability of global exploration, so it can be divided into common communities and recorded as CC_i .

Definition 2. Let the cooperative search activity among different communities be a binary group (C, R) , where $C = \{c_1, c_2, \dots, c_j, \dots, c_n\}$ ($1 \leq j \leq n$) is the sequence of communities participating in the cooperative search activity and $R : C \times C$ is the interdependency among communities in the search process. $\forall (r_s : \langle c_i, c_j \rangle) \in R, (s = 1, 2, 3)$ is called the cooperative relationship unit, where r_1 represents the cooperative relationship between a model community and a common community, r_2 is the cooperative relationship between any two model communities, and r_3 is the cooperative relationship between any two common communities. The number of cooperative units among different communities in a cooperative relationship set is called the module of the cooperative relationship set, which is recorded as $\|R\|$.

Generally, if $r_s(c_1, c_2) = 1$, there is an edge between two cooperative communities, and the more cooperative relationship units, the greater the edge weight between two different nodes. If there is no cooperative relationship between different communities, then $r_s(c_1, c_2) = 0$.

Definition 3. Let $\omega_{i,j} = \sum_{s=1}^{\|R\|} r_s(c_1, c_2)$ be the cooperative weights among different communities in MCCN, where the cooperative weight between c_i and c_j is also called the edge weight of MCCN.

To complete the comprehensive quantitative evaluation of community nodes, the evaluation indexes of the optimum value $g_{\text{best}i}$ of community nodes are introduced: collaboration distance H_i and responsivity e_i .

Definition 4. Collaboration distance. The global optimal value $G_{\text{best}i}$ of the community i is, respectively, compared with the individual optimal position $p_{\text{best}j}$ of the m particles, and the absolute value is obtained, that is to say, the cooperative distance of the global optimal value $G_{\text{best}i}$ is $H_i = (h_1, h_2, \dots, h_m)$.

Definition 5. Responsivity. The threshold D of the qualified distance is set. According to the formula $ev_i = \begin{cases} 0, & h_i > D, \\ 1, & h_i < D, \end{cases}$, the response value of the community particle to the optimal value $g_{\text{best}i}$ of the node can be obtained by traversing the cooperative distance H_i , and then, the responsivity e_i of the global optimal value $G_{\text{best}i}$ can be obtained by adding the response values in sequence.

Definition 6. Community node strength. In MCCN, the strength of the community nodes is defined as s_i

$$s_i = \sum_{c_j \in U_i} \omega_{ij} + e_i \quad (11)$$

where ω_{ij} is the cooperative weight between the community node c_i and c_j , e_i is the responsivity of the community node, and U_i is the neighborhood of the community node c_i , and it satisfies

$$U_i \left\{ c_j \left| \bigvee_{s=1}^{\|R\|} r_s(c_i, c_j) \neq 0 \right. \right\}. \quad (12)$$

Generally, MCCN can be represented by its adjacency matrix as $A(G)_{n \times n} = (B)_{n \times n}$. If $E_{n \times 1} = [e_1, e_2, \dots, e_n]$ is the responsivity matrix of MCCN, then the node strength matrix is as follows: $S_{n \times 1} = [\omega_{11}A(G)_{11} + \omega_{12}A(G)_{12} + \dots + \omega_{1n}A(G)_{1n} + e_1, \omega_{21}A(G)_{21} +$

$$\omega_{22}A(G)_{22} + \dots + \omega_{2n}A(G)_{2n} + e_2, \dots, e_1, \omega_{n1}A(G)_{n1} + \omega_{n2}A(G)_{n2} + \dots + \omega_{n1}A(G)_{11} + e_n].$$

Definition 6 shows that the strength of community nodes not only takes into account the cooperative weights among the nodes of the community but also the optimization of the particles within the node itself. It is a comprehensive evaluation of the community's local information and the ability of the community itself, which can better reflect the community's ability to seek optimal guidance in the entire cooperative network.

Therefore, MCCN can be represented by undirected weighted graphs $G(C, R, W, S)$. $C = \{c_1, c_2, \dots, c_n\}$ represents different types of cooperative community node set, $R = \{r_1(c_1, c_2), r_2(c_1, c_2), \dots, r_k(c_1, c_2), \dots, r_s(c_1, c_2)\}$ represents cooperative relationship edge set, $W = \{\omega_{11}, \omega_{12}, \dots, \omega_{ij}, \dots, \omega_{mn}\}$ ($1 \leq i, j \leq n$) is the cooperative edge weight set among, and $S = \{s_1, s_2, \dots, s_n\}$ is the strength set of community nodes, where s_i is the value of the i -th row of the node strength matrix, representing the attributes of community nodes to measure their search ability. By Definition 6, MCCN can be expressed by adjacent augmentation matrix M as follows: $M(G) = (B, E)_{n \times (n+1)}$. The evolution model of the MCCN cooperative network is therefore

$$B_{n \times n} = B[i, j]_{n \times n} = \begin{cases} \omega, \sum_{s=1}^{\|R\|} r_s(c_i, c_j) = 1, r_s \in R, \\ 0, \sum_{s=1}^{\|R\|} r_s(c_i, c_j) = 0, \text{ or } i = j, r_s \in R. \end{cases} \quad (13)$$

On this basis, the asynchronous parallel search strategy among different communities is formulated to reduce the communication between communities, and the efficient search is realized through the driving evolution mechanism to improve the optimization ability of the algorithm to the task scheduling. The rules of multi-population coevolution are as follows.

Rule 1. Evolutionary rules within microbial communities. In the process of multicommunity coevolution, the particles in a single community can be iteratively optimized according to formula (13) for speed and location updating, and the global optimum value can be generated within the community.

$$\begin{cases} v_{id}^{t+1} = \omega \cdot v_{id}^t + c_1 \cdot r_1 \cdot (P_{id}^t - x_{id}^t) + c_2 \cdot r_2 \cdot (P_{gd}^t - x_{id}^t), \\ x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1}, i = 1, 2, \dots, m, d = 1, 2, \dots, D, \end{cases} \quad (14)$$

where t is the number of iterations of particle search, ω is the inertial weight, $c_1 = c_2 = 2$ is the acceleration constant, and r_1 and r_2 are two random functions varying in the range of $[0, 1]$.

Rule 2. Driving coevolution rules between communities.

Rule 2.1. $\forall (r_1 : \langle CC_i, MC_j \rangle) \in R, \exists g_{\text{best}i} = \max \{g_{\text{best}1}, g_{\text{best}2}, \dots, g_{\text{best}m}\}, G_{\text{best}j} = \min \{G_{\text{best}1}, G_{\text{best}2}, \dots, G_{\text{best}n}\},$ and $g_{\text{best}i} \geq G_{\text{best}j}$. The common community is CC , and the model community is MC . The particulate CC_i in CC enters MC , and the last community MC_j in MC is eliminated. After introducing the model learning factor p_n into the internal evolution rules of CC , the new iterative evolution formula is as follows:

$$\begin{cases} v_{id}^{t+1} = \omega \cdot v_{id}^t + c_1 \cdot r_1 \cdot (P_{id}^t - x_{id}^t) + c_2 \cdot r_2 \cdot (P_{gd}^t - x_{id}^t) + c_3 (P_{nd}^t - x_{id}^t), \\ x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1}, i = 1, 2, \dots, m, d = 1, 2, \dots, D, \end{cases} \quad (15)$$

where $P_{nd} = \sum_{i=1}^n G_{\text{best}i} / n$ and c_3 is a random function and satisfies the convergence constraints $c_1 r_1 + c_2 r_2 + c_3 \in [0, 4]$.

Rule 2.2. $\forall (r_2 : \langle MC_i, MC_j \rangle) \in R, \exists$ the community node strength S_{MCi} satisfies $S_{MCi} \geq S_{MCj}$ for any S_{MCj} .

\Rightarrow Global optimum value of model community: $PG = G_{\text{best}i}$.

Rule 2.3. $\forall (r_3 : \langle CC_i, CC_j \rangle) \in R, \exists$ the community node strength S_{CCi} satisfies $S_{CCi} \geq S_{CCj}$ for any S_{CCj} .

\Rightarrow Global optimum value of common community: $Pg = G_{\text{best}i}$.

4.2. The Coding Strategy for Optimal Scheduling of DRPP. Particle swarm optimization (PSO) is a computational model for real continuous space, and it is difficult to solve the task scheduling problem in discrete space [23]. Therefore, the binary system is used to encode the speed and position of particles, and the mapping from the particle swarm optimization algorithm to discrete space, and from the particle search space to the optimal scheduling scheme, is realized by reconstructing the particle expression.

In the above algorithm, an n row, n column matrix $X : n \times n$ is defined as the position vector matrix of particles. The rows represent the situation of providing STR when any service task is executed, the columns indicate the distribution of service tasks in the scheduling process, and any particle represents the potential solution of the scheduling problem. The coding of the particle position is as follows:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nn} \end{bmatrix}, \quad (16)$$

where $x_{ij} \in \{0, 1\}, \sum_{j=1}^n x_{ij} = 1$.

According to the coding scheme, each row of the location matrix X has one and only one element value is 1, which indicates that DRPP are allocated to service task R . Each DRPP can be invoked by multiple service tasks simultaneously, and the execution of any scheduling task cannot be interrupted.

The defined speed $V : n \times n$ is shown in equation (16), which represents the basic exchange order of particle's assignments to the execution of tasks.

$$V = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ v_{21} & v_{22} & \cdots & v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{n1} & v_{n2} & \cdots & v_{nn} \end{bmatrix}, \quad (17)$$

$$v_{ij} \in \{0, 1\}, v_{ij} + v_{ji} = 0 \text{ or } 1.$$

The exchange operations of addition, subtraction, multiplication, and division in the algorithm are defined as $\Theta, \theta, \otimes,$ and $\oplus,$ respectively. The specific operation rules are as follows:

- (1) $A \cdot \theta \cdot B$: represents $\exists x_{ij} = v_{ij} \Rightarrow x_{ij} = v_{ij} = 0$ in position matrix A and velocity matrix B ; on the contrary, it is 1; $\exists x_{ij} = v_{ij+n} = 1 \Rightarrow v_{ij+n} = 0$
- (2) $A \Theta B$: indicates $\exists x_{ij} = v_{ij} \Rightarrow x_{ij} = v_{ij} = 0$ in position matrix A and velocity matrix B ; the other elements are randomly chosen as 0 or 1
- (3) $c_i \otimes B$: indicates whether the particle performs a Θ operation or not with matrix B according to the

corresponding probability value of the random number c_i

- (4) $A \oplus B$: represents $\forall x_{ia} = 1, x_{jb} = 1, \exists v_{ij} = 1 \Rightarrow x_{ib} = 1, x_{ja} = 1$ in position matrix A and velocity matrix B . According to the above definition of switching operation rules, formula (10) can be updated as follows:

$$\begin{cases} v_{id}^{t+1} = v_{id}^t \ominus c_1 \otimes (P_{id}^t \cdot \theta \cdot x_{id}^t) \ominus c_2 \otimes (P_{gd}^t \cdot \theta \cdot x_{id}^t), \\ x_{id}^{t+1} = x_{id}^t \oplus v_{id}^{t+1}, i = 1, 2, \dots, m; d = 1, 2, \dots, D \end{cases} \quad (18)$$

The coding scheme is simple and feasible and thus meets the requirements of multiservice task scheduling of DRPP. It also clearly describes the mapping relationship between the particle population evolution space and the service task scheduling scheme, thus avoiding repeated searches in the process of particle evolution.

4.3. The Optimal Scheduling Algorithms of DRPP Based on Multicomunity Collaborative Search. Based on the multi-community cooperative search algorithm and its coding scheme, the optimal scheduling process of multiservice tasks for distributed DRPP is shown in Figure 2. The specific steps are as follows.

Step 1. Initialization of population particles. According to the encoding strategy between the particle search space and the task scheduling scheme described in Section 3.2, the initialization of n communities is carried out, and a random location (DRPP allocation scheme) and speed of population particles are given. The number of communities, the number of iterations of particles within the community members, and the acceleration coefficient of particles and the inertia weight coefficient are set.

Step 2. Initialized population particles are evenly distributed into the q process to form a community of size $\text{int}(n/q)$. Residual particles are randomly allocated to the q process. The fitness of each particle in the q community is calculated according to the comprehensive optimization scheduling function constructed in Section 3.3.

Step 3. Asynchronous parallel evolutionary computation is performed by running each community separately in the q process.

Step 4. Calculate the fitness values F_i of each community and divide all communities into either the model community or common community according to the threshold value.

Step 5. According to the interactive evolution mechanism between different particle populations in Section 3.1, the position and velocity of particles are updated according to formula (14), and the global optimal locations of the model and common communities are saved to the optimal storage area.

Step 6. If all the particle populations satisfy the search termination condition, then the algorithm ends, the global optimal solution is obtained from the storage area, and the

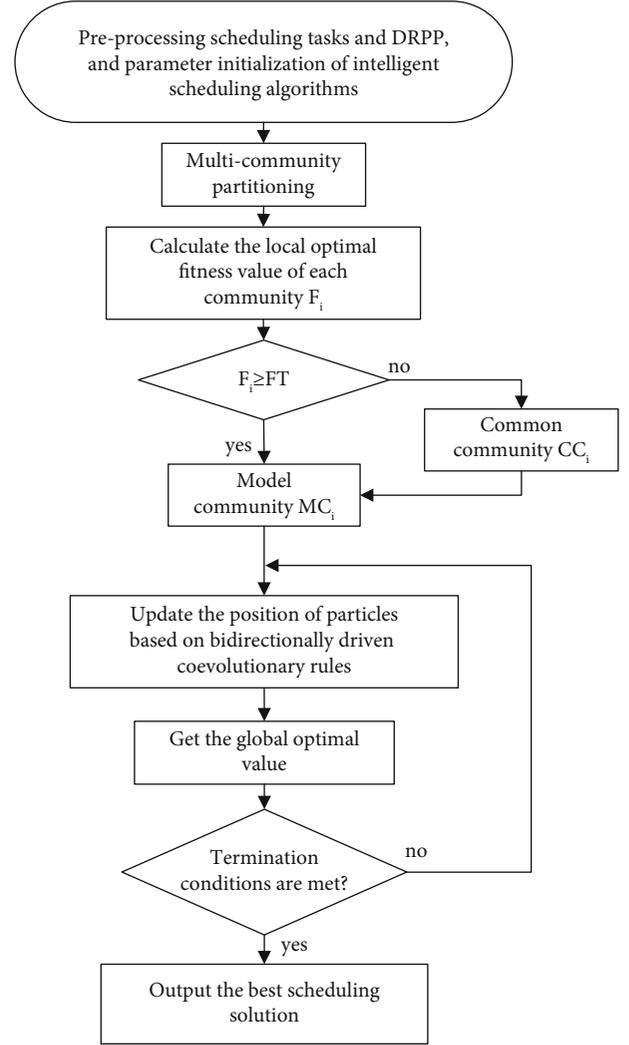


FIGURE 2: Flow chart of scheduling optimization.

optimal scheduling scheme is output; otherwise, it will return to step 5.

5. Application Cases and Analysis

In this section, a DRPP scheduling case for the silk production line quality prediction and early warning service are given to validate the proposed model and algorithm. As shown in Figure 3, the quality prediction and early warning service include the “single operation quality prediction and early warning,” “multiprocess quality prediction and early warning,” “quality prediction and optimization of the whole production line,” and many other services, where each service activity needs to invoke DRPP using service tasks, such as the model, standards, algorithm, and component to provide on-demand service for different business links of the the tobacco production line.

For the task of quality prediction and optimization of silk drying process service in the service platform, it is divided into five subtasks: online data reading, prediction algorithm call, online real-time prediction, prediction result

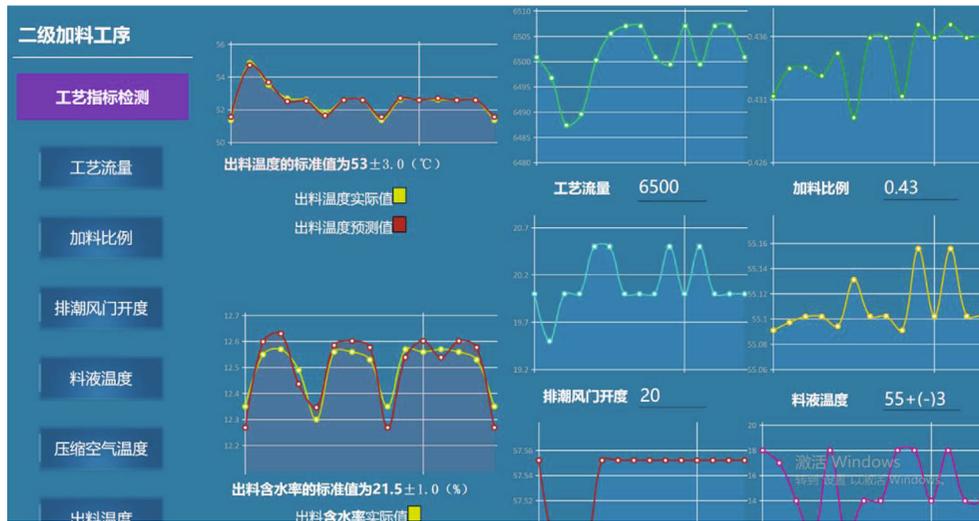


FIGURE 3: The DRPP service platform.

TABLE 1: Particle encoding diagram.

Sub-tasks	Resources					
	X_1	X_2	X_3	X_4	X_5	X_6
t_1	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}
t_2	x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}
t_3	x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}
t_4	x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}
t_5	x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}
t_6	x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}

analysis, optimization algorithm call, and optimization parameter return, which is recorded as task set $\Omega = \{t_1, t_2, \dots, t_5\}$. And then, the available DRPP set corresponding to task set Ω is $X = \{x_1, x_2, \dots, x_N\}$. Following the idea proposed in Sections 1 and 2, the DRPP invoking process can be implemented as follows:

(1) Coding settings

Define the location vector of the particle in the scheduling algorithm as the matrix $X : 5 \times 5$, as shown in Table 1. Row i represents the allocation of service task t_i , and column j represents the service situation of DRPP; if $x_{ij} = 1$, it means that the task t_i is served by data resources X_j . At this time, each particle represents a service task scheduling scheme.

(2) Scheduling algorithms

To verify the effectiveness of the presented scheduling method, the simulation experiment is carried out on the service platform based on the Xeon E5-2609V2 processor and RAM64G, and the service efficiency, response time, and load balancing in the process of multiservice task scheduling are collected as sample data. For comparison, the previously reported ERTPSO algorithm [24], DPSO algorithm [25], LAPSO algorithm [26], and the M-CBDCSM algorithm proposed in the present study are used to solve the optimal

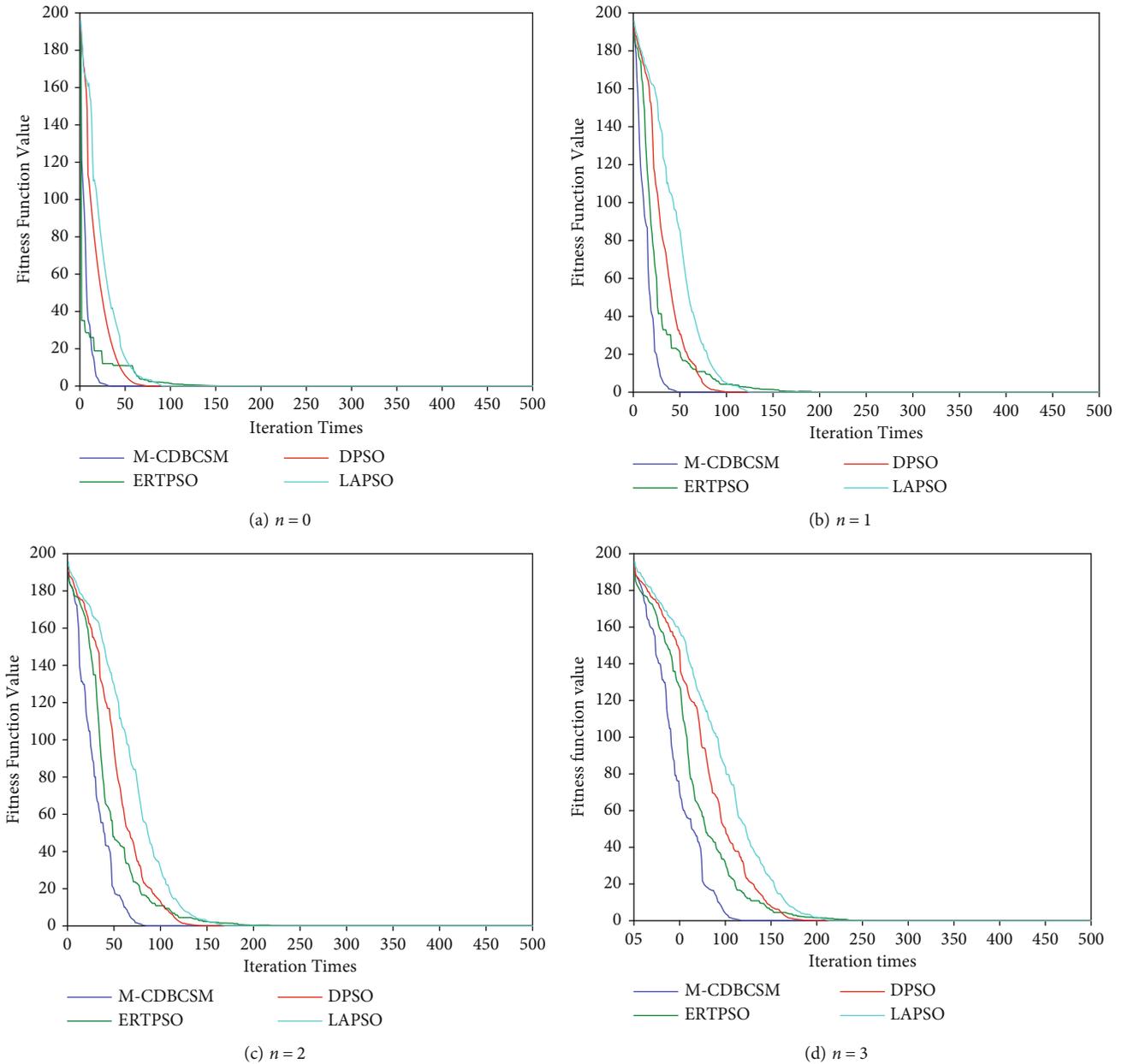


FIGURE 4: Comparison of simulation experiments.

TABLE 2: Comparative experimental results of different task sizes.

Task scale	Convergent algebra	Optimal value	Standard deviation	Average error
5	15	60	$1.73e-011$	$1.21e-012$
10	15	130	$2.81e-011$	$1.61e-012$
15	17	185	$1.58e-010$	$2.57e-012$
20	18	255	$3.95e-010$	$3.43e-012$
25	18	300	$6.12e-010$	$2.69e-011$

($n=0$, iteration = 200).

scheduling scheme for the engine fault identification and maintenance service. The simulation parameters are specified as the population size $q=30-500$, the dimension 30, the evolutionary algebra $i=500$, the inertia weight $\omega=1.1$, and the acceleration constant $c_1=c_2=2$. Moreover, a random function $\text{rand}()$ is introduced into the service response time Rt_d ; $Rt_d = Rt + n \cdot \text{rand}()$, where $\text{rand}()$ varies randomly in a range $[0, 1]$, and $n=0, 1, 2, 3$. All the simulation experiments were carried out 500 times, and other parameters are the same to those used in the relevant literature.

Simulation results of the relevant algorithm are provided in Figure 4. It is shown that the multicommunity cooperative search algorithm can better adapt to the random changes of service response time in the process of

multiservice task scheduling, in particular for the interactive evolutionary rules between communities are selected adaptively, the algorithm converges to the global optimal value quickly and stably. In particular, the optimal scheduling scheme can be found before 50 generations under different search conditions corresponding to the discrete particle swarm optimization (DPSO). Especially in the face of dynamic and random multitask scheduling, hybrid genetic algorithm is difficult to adaptively carry out individual mutation and cross-operation and then can not track the dynamic change of service scheduling, which makes it difficult to avoid premature phenomenon under high random search conditions, and its algorithm performance is far lower than multicommunity cooperative search algorithm.

To further validate the strong adaptability of the algorithm in the face of multiple scheduling tasks, the experiments of large population based on multi scheduling tasks is also simulated. For this purpose, the population and the scheduling tasks can be given as $p = 1000$, $t = 5-20$. The corresponding results are provided in Table 2. Although the number of service tasks is increasing, the convergence rate of multi community cooperative search algorithm does not decline significantly in the process of scheduling. From aforementioned results, one can conclude that the proposed algorithms in this paper can achieve better performance for large population and tasks-varying parameters in terms of convergence speed and steady-state errors.

6. Conclusion

This paper addresses the multitask adaptive scheduling of DRPP. A heuristic scheduling framework are employed to deal with the uncertainty of DRPP service response and the imbalance of resource allocation. The load manager and dynamic scheduling engine are employed to approximate the uncertainty of scheduling service. Moreover, we propose novel cooperative search algorithm of the task scheduling model driven by scheduling objectives with service efficiency, reduced response time, and load balancing, so that fast scheduling convergence can be proved even in the dynamic random sense. The proposed scheduling schemes are robust against dynamic random disturbances. To guarantee the discrete data space of optimization algorithm, a binary coding strategy to map the particle position vector to resource allocation is introduced. Simulation examples are provided to verify the efficacy of the proposed algorithm.

Data Availability

The production process data used to support the findings of this study have not been made available because the enterprise of the data source requires confidentiality.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This study is supported by the National Natural Science Foundation of China (52065033) and Yunnan provincial major science and technology special plan projects: digitization research and application demonstration of Yunnan characteristic industry (202002AD080001).

References

- [1] W. Xu, S. Guo, X. Li, C. Guo, R. Wu, and Z. Peng, "A dynamic scheduling method for logistics tasks oriented to intelligent manufacturing workshop," *Mathematical Problems in Engineering*, vol. 2019, 18 pages, 2019.
- [2] Z. F. Liu, W. Chen, C. B. Yang, Q. Cheng, and Y. S. Zhao, "Intelligent manufacturing workshop dispatching cloud platform based on digital twins," *Computer Integrated Manufacturing Systems*, vol. 25, no. 6, pp. 1444–1453, 2019.
- [3] S. Huang, Y. Guo, and S. Zha, "Review on internet-of-manufacturing-things and key technologies for discrete workshop," *Jisuanji Jicheng Zhizao Xitong/Computer Integrated Manufacturing Systems, CIMS*, vol. 25, no. 2, pp. 284–302, 2019.
- [4] M. K. Lim, W. Xiong, and Z. Lei, "Theory, supporting technology and application analysis of cloud manufacturing: a systematic and comprehensive literature review," *Industrial Management & Data Systems*, vol. 120, no. 8, pp. 1585–1614, 2020.
- [5] X. Sun, G. Gao, B. Tian, B. Li, S. Zhang, and J. Huang, "Intelligent recognition of production date based on machine vision," in *Journal of Physics: Conference Series, Volume 1267, 2019 3rd International Conference on Artificial Intelligence, Automation and Control Technologies (AIAC T 2019)*, Xi'an, China, April 2019.
- [6] X. Ye, X. Wu, and Y. Guo, "Real-time quality prediction of casting billet based on random forest algorithm," in *2018 IEEE International Conference on Progress in Informatics and Computing (PIC)*, Suzhou, China, 2019.
- [7] L. Zhiyong, Z. Rou, Z. Jie, and C. Ting, "Research on optimization of centrifugal process parameters based on support vector machine," in *2020 39th Chinese Control Conference (CCC)*, Shenyang, China, 2020.
- [8] H. Li, C. Wang, S. Jiang, S. Liu, Y. Rong, and X. Li, "The study of intelligent scheduling algorithm oriented to complex constraints and multi-process roller grinding workshop," *Advances in Mechanical Engineering*, vol. 12, no. 11, 2020.
- [9] X. P. Yang and X. L. Gao, "Optimization of dynamic and multi-objective flexible job-shop scheduling based on parallel hybrid algorithm," *International Journal of Simulation Modelling*, vol. 17, no. 4, pp. 724–733, 2018.
- [10] J. Liang, Y. Wang, Z. H. Zhang, and Y. Sun, "Energy efficient production planning and scheduling problem with processing technology selection," *Computers & Industrial Engineering*, vol. 132, pp. 260–270, 2019.
- [11] L. Qiao, Z. Zhang, and Z. Huang, "A scheduling algorithm for multi-workshop production based on BOM and process route," *Applied Sciences*, vol. 11, no. 11, p. 5078, 2021.
- [12] Y. Dan, G. Liu, and Y. Fu, "Optimized flowshop scheduling for precast production considering process connection and blocking," *Automation in Construction*, vol. 125, article 103575, 2021.

- [13] F. Benda, R. Braune, K. F. Doerner, and R. F. Hartl, "A machine learning approach for flow shop scheduling problems with alternative resources, sequence-dependent setup times, and blocking," *OR Spectrum*, vol. 41, no. 4, pp. 871–893, 2019.
- [14] Y. Hirochika and N. Hirofumi, "Estimation of processing time using machine learning and real factory data for optimization of parallel machine scheduling problem," *Operations Research Perspectives*, vol. 8, article 100196, 2021.
- [15] X. Feng and Z. Xu, "Integrated production and transportation scheduling on parallel batch-processing machines," *IEEE Access*, vol. 7, pp. 148393–148400, 2019.
- [16] J. F. Vijay, "Cloud data analysis using a genetic algorithm-based job scheduling process," *Expert Systems*, vol. 36, no. 5, 2019.
- [17] L. S. Dias and M. G. Ierapetritou, "Integration of planning, scheduling and control problems using data-driven feasibility analysis and surrogate models," *Computers & Chemical Engineering*, vol. 134, p. 106714, 2020.
- [18] A. Jlrđ, A. Nf, and B. Tj, "Multi-process production scheduling with variable renewable integration and demand response," *European Journal of Operational Research*, vol. 281, no. 1, pp. 186–200, 2020.
- [19] C. Tsay, A. Kumar, J. Flores-Cerrillo, and M. Baldea, "Optimal demand response scheduling of an industrial air separation unit using data-driven dynamic models," *Computers & Chemical Engineering*, vol. 126, pp. 22–34, 2019.
- [20] O. J. Fisher, N. J. Watson, J. E. Escrig et al., "Considerations, challenges and opportunities when developing data-driven models for process manufacturing systems," *Computers & Chemical Engineering*, vol. 140, p. 106881, 2020.
- [21] A. Allahverdi and F. S. al-Anzi, "A PSO and a Tabu search heuristics for the assembly scheduling problem of the two-stage distributed database application," *Computers and Operations Research*, vol. 33, no. 4, pp. 1056–1080, 2006.
- [22] M. Yu, Y. Zhang, K. Chen, and D. Zhang, "Integration of process planning and scheduling using a hybrid GA/PSO algorithm," *International Journal of Advanced Manufacturing Technology*, vol. 78, no. 1-4, pp. 583–592, 2015.
- [23] A. Brusaferrri, E. Leo, L. Nicolosi, D. Ramin, and S. Spinelli, "Integrated automation system with PSO based scheduling for PCB remanufacturing plants," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, Helsinki, Finland, 2019.
- [24] Y. Yin, S. Linfu, and Y. Chengfeng, "Particle swarm optimization algorithm for real-time parameter evaluation based on embedded diamond thinking," *Journal of Systems Simulation*, vol. 21, no. 14, pp. 4317–4323, 2009.
- [25] H. Kaijun and L. Huaiwei, "DPSO-based task scheduling algorithm in cloud environment," *Computer Engineering*, vol. 40, no. 1, pp. 59–62, 2014.
- [26] Y. Chengyu, C. Dongning, and Z. Ruixing, "Hybrid-particle interaction particle swarm optimization algorithm," *Chinese Journal of Mechanical Engineering*, vol. 51, no. 6, pp. 198–207, 2015.