Int j simul model 20 (2021) 2, 398-409
Original scientific paper

GAME-BASED HYBRID PARTICLE SWARM OPTIMIZATION OF JOB-SHOP PRODUCTION CONTROL

Wang, X. L.
Management Department, Shijiazhuang University of Applied Technology, Shijiazhuang 050081, China
E-Mail: leilei523192@126.com

Abstract
The traditional multi-objective particle swarm optimization (PSO) cannot effectively handle the production control problem involving multiple types of production lines or production objectives. Therefore, this paper designs a game-based hybrid PSO (GBHPSO) for job-shop production control. Firstly, a job-shop model was established with parts processing line, parts assembly line, and product assembly line, and the production control ideas were designed to combine real-time monitoring of events and operation sequence adjustment. Then, the production control objectives were determined for the three production lines. After that, the GBHPSO was applied to solve the job-shop production control problem, the product utility function was constructed, and the execution low was detailed for the solving algorithm. Experiments demonstrate the effectiveness of our algorithm. The research provides a reference for applying our algorithm in resource allocation of other production fields.
(Received in January 2021, accepted in April 2021. This paper was with the author 1 month for 2 revisions.)

Key Words: Game-Based Hybrid Particle Swarm Optimization (GBHPSO), Production Control, Product Utility

1. INTRODUCTION

With the advent of intelligent manufacturing, the various sensors of job-shop data acquisition and the Internet of things (IoT) for system interconnection have been increasingly applied to job-shop control [1-6]. The difficulty of job-shop control lies in the many uncertain disturbances in the processing environment and the complex changes of working conditions [7-10]. For the frequent disturbances, their degree of impact on job-shop production efficiency and control cost must be quantified. This is essential to the active job-shop scheduling and the early warning of disturbances, and meaningful for improving the rescheduling ability of the job-shop after being disturbed [11-14].

Yahouni et al. [15] constructed a multi-disturbance partition model for the discrete job-shop hierarchy tree; the model accurately defines the priorities of disturbance risk vectors, and differentiated between different types of disturbances. Guo [16] established a mathematical model of the time flow of job-shop operations under ideal and actual conditions, and realized the closed-loop control from discrete job-shop disturbance prediction to production based on the hybrid Bayesian decision tree algorithm. Nababan et al. [17] designed a multi-objective scheduling solving model to optimize the makespan and delivery delay of discrete job-shop, and completed real-time adjustment of some production operations of the job-shop, using the ADSM method. Jahan et al. [18] defined five basic job-shop elements, namely, product, output, technique, production assistance and production time, and created a lean job-shop production management system through systematic design planning (SLP).

Facing the diverse and fast changing market demand, the traditional methods for job-shop scheduling and product assembly cannot meet the requirements on logistics efficiency in the job-shop [19-22]. Bożejko et al. [23] fully considered the onsite job-shop management and intelligent logistics control, and provided the multi-objective optimization function and its constraints for minimizing the path length and distribution cost. The core of scientific job-shop production control is to support real-time interaction between information and managers, on the premise of acquiring the various factors affecting product quality. However, it is often difficult to handle or prewarn the product quality problems on the site of job-shop production. To overcome the difficulty, Al Aqel et al. [24] proposed a job-shop product quality control system based on multiple mobile communication terminals, and explained the design flow of the hardware and software for subsystem interaction, quality prewarning, etc. Most assembly job-shops have problems like poor information interaction and difficulty in real-time monitoring. Through semantic data modelling, Wang et al. [25] unified the information interaction model between the objects of assembly job-shop production control, and realized the logical integration of the assembly system, such that the managers, machines, operators, and materials can operate in an integrated manner.

On job-shop production control, the traditional multi-objective particle swarm optimization (PSO) has relatively good convergence and diversity, when the production lines are independent or the production objectives are not very diversified. However, the traditional approach has great challenges if the working conditions are very complex. To effectively reduce the probability of the PSO falling into local optimum trap under complex working conditions, this paper puts forward a game-based hybrid PSO (GBHPSO) for job-shop production control. Section 2 creates a job-shop model containing parts processing line, parts assembly line, and product assembly line. Section 3 constructs the production control objectives for the three production lines. Section 4 solves the job-shop production control problem with GBHPSO, sets up product utility function and game model, and details the execution low of the solving algorithm. Experimental results show that our algorithm can improve the reliability and reasonability of job-shop production control to a certain extent.

2. JOB-SHOP STRUCTURE AND PRODUCTION CONTROL IDEAS

Fig. 1 shows the proposed job-shop model, which contains parts processing line, parts assembly line, and product assembly line. Before setting up the optimization objectives of the production control system for the job-shop model, the demands for parts should be smoothened, and the features of different production lines should be fully considered. If an uncertain delay event occurs in the job-shop, the information feedback and control of each production line should be rationalized through the production control that combines real-time event monitoring and operation sequence adjustment.

Then, the supply-demand details of each production line were supplemented. On this basis, the entity-relationship diagram of job-shop production control system was obtained (Fig. 2). The diagram presents an institutive picture of the relationship between warehouses, parts, components, and products, and provides the real-time data on the attributes of each entity.

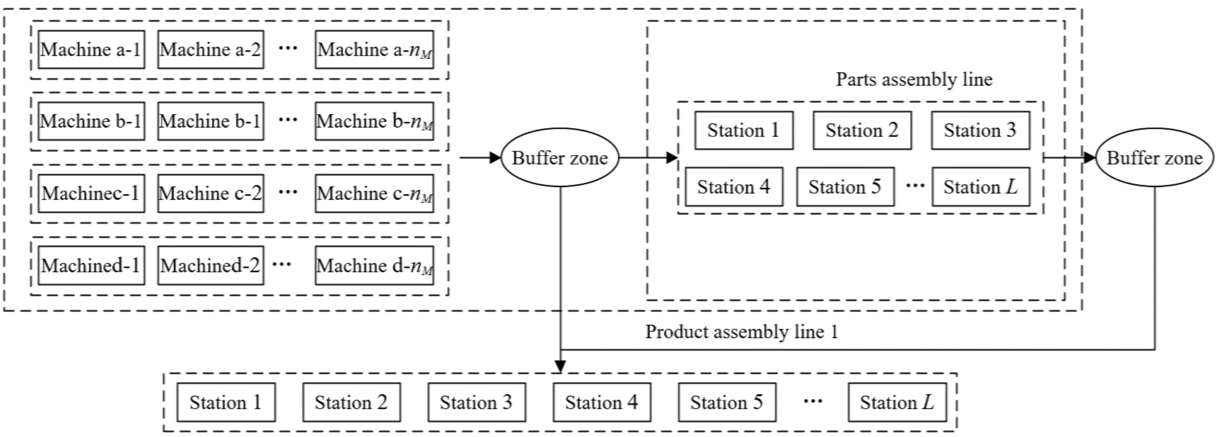

Figure 1: Structure of job-shop model.

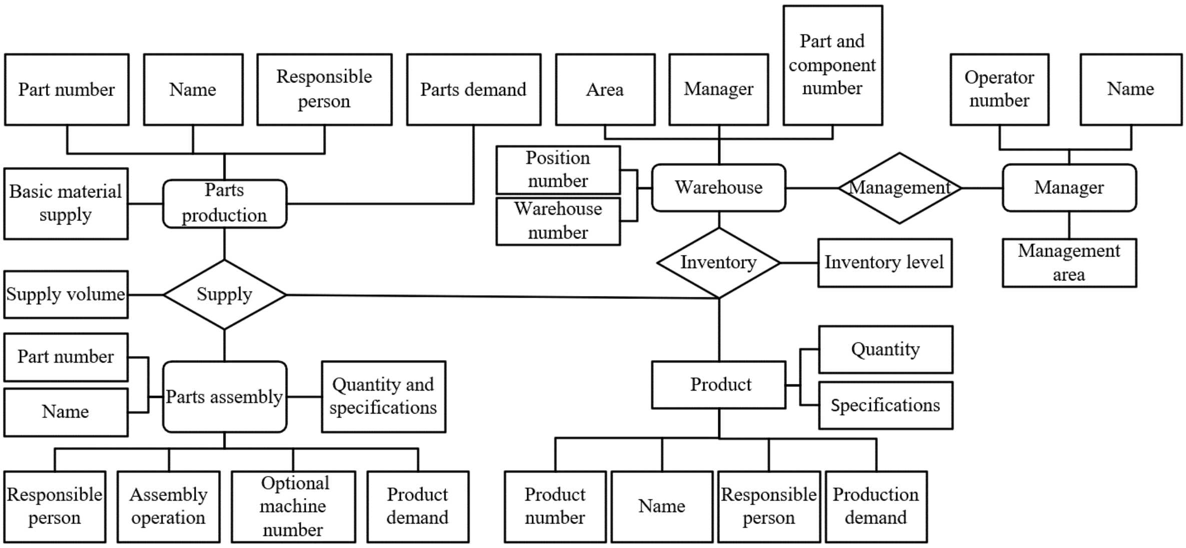

Figure 2: Entity-relationship diagram of job-shop production control system.

3. JOB-SHOP PRODUCTION MODELLING

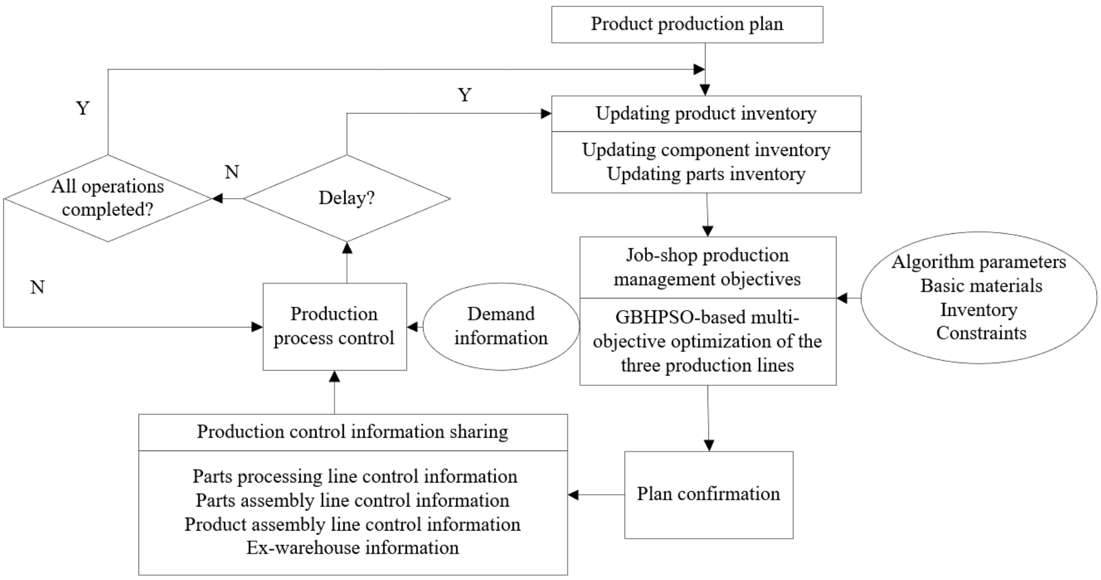

Figure 3: Flowchart of job-shop production control system.

3.1 Production control objectives of product assembly line

In the l^{th} production sequence, the number of i^{th} products is denoted as b_{il}; the number of the j^{th} parts to be provided by the current station to assembly the b_{il} products is denoted as a_{jl}; the total number of the j^{th} parts is denoted as m_j; the inventory of the j^{th} parts in the buffer zone is denoted as HC_j; the total number of positions and parts are denoted as L and J, respectively. Then, the objective function for smoothing the parts demand can be created as:

$$min\sum_{j=1}^{J}\sum_{l=1}^{L}\left(a_{jl}-HC_j-l\times m_j/L\right)^2 \tag{1}$$

Eq. (1) tries to keep the actual demand for the part consistent with the mean demand. Let I be the number of products to manufactured; w_{il} be an indicator about the station of the i^{th} product to be manufactured on the production line. Then, Eq. (1) should satisfy:

$$s.t.\quad w_{il}\in\{0,1\}\ \forall i\in I\ \ l=1,2,...L \tag{2}$$

Eq. (2) shows that w_{il} is a binary function. The value of that function is either zero or one. If the station corresponding to the l^{th} production sequence is only arranged one product:

$$\sum_{i \in I} w_{il} = 1 \ \forall i = 1, 2, ... I \tag{3}$$

Let q_i and e_i be the target quantity and inventory of the i^{th} products, respectively. If the i^{th} products are arranged on the station corresponding to the l^{th} production sequence:

$$\sum_{l=1}^{L} w_{il} = q_i - e_i \ \forall i \in I \tag{4}$$

The number of the i^{th} products can be expressed as:

$$b_{il} = \sum_{l=1}^{L} w_{il} \ \forall l = 1, 2, ... L \tag{5}$$

Let s_{ij} be the number of j^{th} parts required for each product. Then,

$$b_{il} = \sum_{i=1}^{k} b_{il} s_{ij} \tag{6}$$

m_j can be calculated by:

$$m_j = \sum_{i=1}^{I} (q_i - e_i) s_{ij} \tag{7}$$

The completion time F_i and delivery date DP_i of the i^{th} product satisfy:

$$F_i \le DP_i \tag{8}$$

3.2 Production control objectives of parts assembly line

Suppose J parts need to be assembled independently on the L stations of the parts assembly line. The J parts belong to the orders of Q products. Among them, the q^{th} product contains M_q parts: $H^q{}_1$, $H^q{}_2 \ldots H^q M_q$, $\sum^{M}_{q=1} M_q = J$. Let $t_{lHj}{}^q$ be the processing time of the j^{th} part $H^q{}_j$ of the q^{th} product on the l^{th} station; TSq be the processing time of M_q parts; $OV_{jl}{}^q$ be the completion time of part $H^q{}_j$ on station l. Then, the binary function $T(H^q{}_j)$ about whether a part is assembled on time can be established as:

$$T\left(H_j^q\right) = \begin{cases} 1 & OV_{j,l}^q \le TS_q \\ 0 & others \end{cases} \tag{9}$$

If $T(H^q{}_j) = 1$, the assemblage is on time; if $T(H^q{}_j) = 0$, the assemblage is delayed. The integrity coefficient Φ_q can be defined as:

$$\phi_q = \begin{cases} 1 & \sum_{j=1}^{M_q} T\left(H_j^q\right) = M_q \\ 0 & \sum_{j=1}^{M_q} T\left(H_j^q\right) < M_q \end{cases} \tag{10}$$

Let ω_q be the weight of the q^{th} product order. Then, we have the following model:

$$max \sum_{q=1}^{Q} \omega_q \phi_q \tag{11}$$

Eq. (11) can be converted into:

$$min \frac{1}{\sum_{q=1}^{Q} \omega_q \phi_q} \tag{12}$$

Let ΔlH_j be the waiting time for part H_j on the l^{th} station. Eq. (12) should satisfy:

$$s.t.\ \ OV_{1,1}^{1}=t_{1H_1^1}+\Delta_{1H_1} \tag{13}$$

The completion time of the first part in the parts assembly sequence on the first machine, and that of parts on the first station can be respectively obtained by:

$$OV_{1,1}^{1}=OV_{1,(l\text{-}1)}^{1}+t_{lH_1^1}+\Delta_{lH_1} \tag{14}$$

$$OV_{j,1}^{q}=OV_{h,1}^{i}+t_{1H_j^q}+\Delta_{1H_j^q} \tag{15}$$

Starting with the second sequence, the completion time of pats on the other stations can be obtained by:

$$OV_{j,l}^{q}=max\left\{OV_{h,l}^{i},OV_{j,(l\text{-}1)}^{q}\right\}+t_{1H_j^q}+\Delta_{lH_j^q} \tag{16}$$

Let $W_{cl\tau}$ be the number of the c^{th} parts in the buffer zone at time τ; W_{clHj} be the total number of the c^{th} parts required for part H_j on the l^{th} station. Then, ΔlH_j can be obtained by:

$$\Delta_{lH_j}=\begin{cases}\hat{\tau}-\tau & W_{cl\tau}<W_{clH_j}\\ 0 & W_{cl\tau}>W_{clH_j}\end{cases} \tag{17}$$

3.3 Production control objectives of parts processing line

The production control of parts processing line can be simplified as rationalizing the production sequence and operation interval of parts, with the aim to minimize the difference between completion time of parts and demand time. Let M_C and n_M be the number of parts and machines, respectively; OV_{cbl}, τ_{cbl}, and O_c be the completion time, processing time, and completion moment of the b^{th} operation of the c^{th} part on the l^{th} station, respectively; S_c be the start moment of the secondary operation after the completion of the c^{th} part. Then, an objective function can be established as:

$$min\sum_{c=1}^{M_C}\left(OV_c-S_c\right) \tag{18}$$

Eq. (18) needs to satisfy the constraint on the processing time of the c^{th} part:

$$s.t.\ OV_{c\tau\tau}-OV_{c(b\text{-}1)q}\geq\tau_{cbl} \tag{19}$$

Eq. (19) shows that the machine corresponding to the l^{th} station cannot start the b^{th} operation until completing the $(b\text{-}1)^{th}$ operation. The resource constraint on the l^{th} station can be given by:

$$OV_{qjl}-OV_{hel}\geq\tau_{qjl} \tag{20}$$

Eq. (20) shows that the l^{th} station cannot start processing another part until completing the processing of the current part. OV_{cbl} can be calculated by:

$$OV_{cbl}=max\left(OV_{(c\text{-}1)hl},OV_{c(b\text{-}1)q}\right)+\tau_{cbl} \tag{21}$$

4. GBHPSO-BASED PRODUCTION CONTROL STRATEGY

4.1 Algorithm design

Fig. 4 presents the game relationships between the independent production lines. It can be seen that the GBHPSO firstly quantifies the subjective factors of the management effect of each production line into computable form or analysable form based on the game theory, and then fuses the two kinds of factors into the PSO for each production line.

Let N_G be the number of particles in the swarm; e be the dimensionality of the search space. After r iterations, the position and velocity of the i^{th} particle can be described as $A_{ir}=(A^1_{ir}, A^2_{ir}, \ldots, A^e_{ir})$ and $\upsilon_{ir}=(\upsilon^1_{ir}, \upsilon^2_{ir}, \ldots, \upsilon^e_{ir})$, respectively, $i\in[r, N_G]$, $r\in[0, R_{max}]$. Suppose the global optimal and local optimal are $G_b = (g_1, g_2, ..., g_e)$ and $P_b = (p_i^1, p_i^2, ..., p_i^e)$, respectively. The values of G_b and P_b can be derived from the fitness based on position vectors. Let θ be the dynamic inertial weight; σ_1 and σ_2 be the learning factors. Then, the particle velocity can be updated by:

$$\upsilon_{i,j}(\tau+1) = \theta\upsilon_{i,j}(\tau) + \sigma_1 \times \eta_1 \times \left[P_b(i,j) - A_{i,j}(\tau)\right] + \sigma_2 \times \eta_2 \times \left[G_b(j) - A_{i,j}(\tau)\right] \quad (22)$$

where, η_1 and η_2 are random numbers in [0, 1]. The particle position can be updated by:

$$A_{i,j}(\tau+1) = A_{i,j}(\tau) + \upsilon_{i,j}(\tau+1) \quad (23)$$

Let R and R_{max} be the current and maximum number of iterations, respectively. The value of θ can be obtained by:

$$\theta = \theta_{max} - r\frac{\theta_{max} - \theta_{min}}{R_{max}} \quad (24)$$

To prevent premature convergence caused by excessively fast velocity, the particle velocity is bounded by a maximum υ_{max} and a minimum υ_{min}. The positions A_i corresponding to the maximum and minimum velocities can be denoted as A_{max} and A_{min}, respectively.

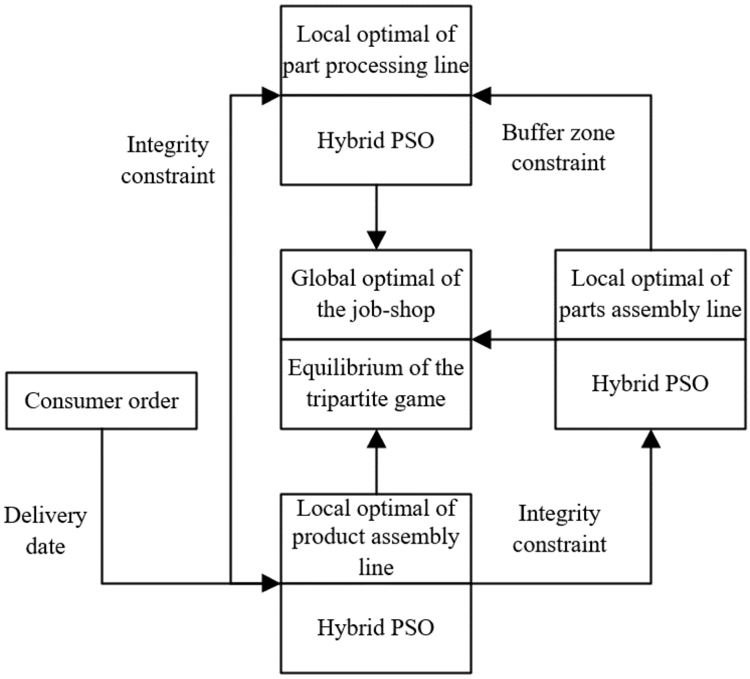

Figure 4: Game relationships between independent production lines.

4.2 Product utility function and game modelling

The cooperative relay game in the GBHPSO is equivalent to a Stackelberg game that includes a first-mover and a responder. As the first-mover, each station can set the unit price of the cooperation according to the total demand of all products to be manufactured. As the responder, each product can determine its ranking or share of processing and assembly operations according to the cooperation price.

Let $PR = \{1, 2, \ldots, m_{pr}\}$ be the set of the products to be manufactured in the system; γ_i be the start moment of its operations applied by the production control strategy of the i^{th} product to the l^{th} station; $\gamma = \{\gamma_1, ..., \gamma_{mpr}\}$ be the set of the production control strategies of all products $\{\gamma_1, ..., \gamma_{mpr}\}$. Then, the pricing strategy of the l^{th} station can be defined as:

$$BU(l) = s\left(\sum_{i=1}^{m_{pr}} \gamma_i\right)^{\zeta} \quad (25)$$

If the total processing time of all products exceeds the working time limit of the station node, the station will reject the product requiring the shortest processing time, in order to maximize its utility. In addition, the station offers all products the same price.

In the GBHPSO, every network node hopes to obtain more processing time *SH* at a small cost Ψ. During the cooperation, the nodes seek a compromise by the energy efficiency function $v = SH/\Psi$. When the product demand at the start moment is γ, the utility function can be expressed as:

$$O_i = \left(SH_{c_i,e_i}\left(\Psi_i,\gamma-\gamma_i\right)+SH^*_{c_i,e_i}\left(\Psi_i,\gamma_i\right)\right)/\Psi_i - BU\cdot\gamma_i \tag{26}$$

Let *K* and *D* be the delay and actual processing time, respectively. Then, $SH_{ci,ei}(\Psi_i, \gamma\text{-}\gamma_i) = (\gamma\text{-}\gamma_i)g(\beta_{bi,ei})K/D$ is the time consumed by the i^{th} product on a randomly selected station during the period $\gamma\text{-}\gamma_i$. $SH^*_{ci,\,ei}(\Psi_i, \gamma_i) = \gamma_i g(\beta^*_{bi,\,ei})K/D$ is the time consumed by the i^{th} product and the l^{th} station, which jointly occupies the start moment γ_i. The i^{th} product needs to pay $BU\,\gamma_i$ for the occupation.

To find the equilibrium solution to the cooperative game between all products and stations, it is necessary to determine the optimal strategy of the game parties. Combining Eqs. (25) and (26):

$$O_i = \frac{K\gamma}{D\Psi_i}g\left(\beta_{c_i,e_i}\right)+\frac{K\gamma_i}{D\Psi_i}g\left(\beta^*_{c_i,e_i},\beta_{c_i,e_i}\right)-\gamma_i\left(s\left(\sum_{j=1}^{m_{Pr}}\gamma_j\right)^{\zeta}\right) \tag{27}$$

where, $g(\beta^*_{ci,ei}, \beta_{ci,ei}) = g(\beta^*_{ci,ei}) - g(\beta_{ci,ei})$ is the time gain of the completion time through the cooperative production of the i^{th} product over that of a randomly selected station. Eq. (27) shows that the utility of each product is affected by the production control strategy of every other product in the network. Let $\gamma_i' = \{\gamma_i'\}^{mpr}_{j=1,j\neq i}$ be the set of optimal production control strategies of all the products other than the i^{th} product. Then, the optimal strategy of the i^{th} product can be expressed as:

$$\gamma_i' = max\left(0, arg\,\max_{\gamma_i} O_i\left(\gamma_i,\gamma_i'\right)\right) \tag{28}$$

The above equilibrium strategy is not optimal for the global utility of all the products to be manufactured in the job-shop. To improve the utility of all products, the Pareto optimal solution can be constructed as:

$$O_i^{\mu} = \frac{K\gamma}{D\Psi_i}g\left(\beta_{c_i,e_i}\right)+\frac{K\mu\gamma_i'}{D\Psi_i}\Delta g\left(\beta^*_{c_i,e_i},\beta_{c_i,e_i}\right)-\mu\gamma'\left(s\left(\sum_{j=1}^{m_{pr}}\gamma_j'\right)^{\zeta}\right) \tag{29}$$

Suppose under the equilibrium strategy $\{\gamma_i'\}^{mpr}_{i=1}$, all products change its start moment of its operations with parameter $\mu(\mu > 0)$. Then, the new set of production control demands can be described as $\{\mu\gamma_i'\}^{mpr}_{i=1}$. Under the new strategy, the utility function of the i^{th} product can be given by:

$$\frac{\partial O_i^{\mu}}{\partial\mu} = \frac{K\gamma_i'}{D\Psi_i}\Delta g\left(\beta^*_{c_i,e_i},\beta_{c_i,e_i}\right)-s\gamma_i'\left(1+\zeta\right)\mu^{\zeta}\left(\sum_{j=1}^{m_{pr}}\gamma_j'\right)^{\zeta} \tag{30}$$

According to Eq. (28), the equilibrium conditions can be defined as:

$$\frac{K}{D\Psi_i}\Delta g\left(\beta^*_{c_i,e_i},\beta_{c_i,e_i}\right) = s\left(\sum_{j=1}^{m_{pr}}\gamma_j'\right)^{\zeta}+\zeta\gamma_i\left(\sum_{j=1}^{m_P}\gamma_j'\right)^{\zeta-1} \tag{31}$$

Combining Eqs. (30) and (31):

$$\frac{\partial O_i^{\mu}}{\partial\mu} = s\gamma_i'\left(\sum_{j=1}^{m_{pr}}\gamma_j'\right)^{\zeta}\left(1-\mu^{\zeta}\right)+s\zeta\,\gamma_i'\left(\sum_{j=1}^{m_{pr}}\gamma_j'\right)^{\zeta-1}\left(\gamma'-\mu^{\zeta}\sum_{j=1}^{m_{pr}}\gamma_j'\right) \tag{32}$$

Whether the equilibrium strategy is Pareto optimal can be judged by:

$$\frac{\partial O_i^{\mu}}{\partial \mu}\Big|_{\mu=1} = s\zeta\gamma_i'\left(\sum_{j=1}^{m_{pr}}\gamma_j'\right)^{\zeta-1}\left(\sum_{j=1,\ j\neq i}^{m_{pr}}\gamma_j'\right) \tag{33}$$

Since γ'_i, s, and ζ are greater than zero, $\partial^2 O^{\mu}{}_i/\partial^{\mu}\,\big|_{\mu=1}$ must be negative. This means the original equilibrium strategy is not optimal.

4.3 Algorithm flow

The flow of the job-shop production control problem by the GBHPSO is detailed in Fig. 5.

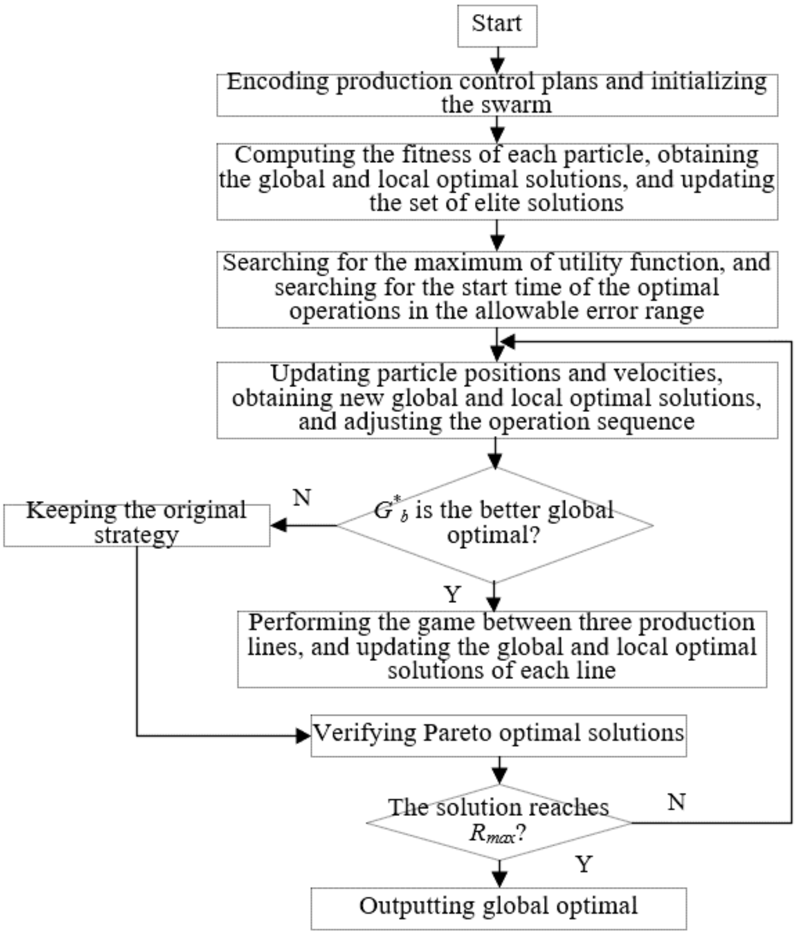

Figure 5: Flowchart of our algorithm.

5. EXPERIMENTS AND RESULTS ANALYSIS

To verify the effectiveness of our algorithm on production scheduling of job-shops involving multiple production lines, the production control strategy was studied for an industrial job-shop of an industrial manufacturer producing various kinds of products. The job-shop contains six machines. Table I presents the operation time of each operation in product assembly. Table II presents the operation time of each operation in parts assembly.

Figs. 6 and 7 are the Gantt charts of product assembly and parts assembly, respectively. The x- and y-axes represent machine number and production time, respectively; *nm* stands for the m^{th} operation of the n^{th} product or part. It can be seen that the assembly operations on the two lines are arranged uniformly, indicating that our algorithm can effectively improve the reliability and reasonability of job-shop production control.

Table I: Operation time of each operation in product assembly.

Operation number	A	B	C	D	E	F
1	24	13	25	33	21	22
2	21	20	27	26	24	25
3	43	22	56	44	21	43
4	25	59	22	28	56	21
5	41	38	19	12	26	16

Table II: Operation time of each operation in parts assembly.

Machine number	*a*		*b*		*c*	
	Operation	Time	Operation	Time	Operation	Time
1	1	11	1	13	1	17
2	2	13	2	16	2	14
3	3	13	3	15	3	13
4					4	11
5						
6	4	14	4	12		
Machine number	*d*		*e*		*f*	
	Operation	Time	Operation	Time	Operation	Time
1			1	17	1	15
2	1	16				
3	2	15			2	16
4			2	16		
5	3	17	3	13	3	12
6	4	11	4	15	4	11

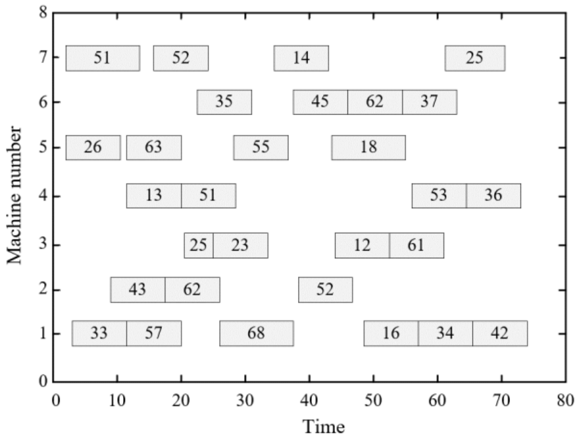

Figure 6: Gantt chart of product assembly.

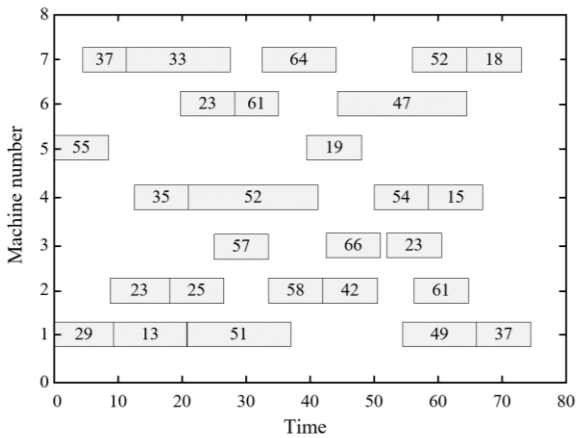

Figure 7: Gantt chart of parts assembly.

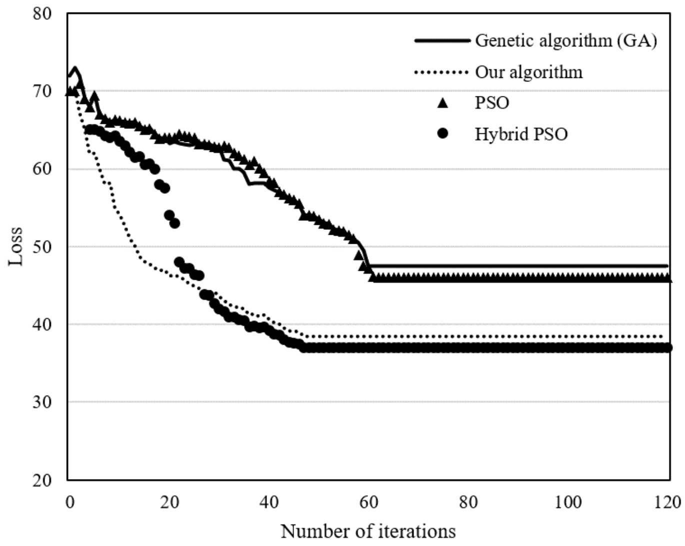

Figure 8: Iterative losses of different algorithms.

Based on the previously collected real-time data on each entity, the performance of our algorithm was further verified through comparative analysis. The swarm size was initialized as 500; the maximum number of iterations was set to 150; the mutation rate was defined as 6 %. Then, the iterative process of our GBHPSO was compared with that of other algorithms on MATLAB. Obviously, our algorithm converged faster than the other algorithms. Therefore, the introduction of position vector effectively improves the ranking of operations.

Table III compares the processing time computed by different algorithms. It can be seen that our algorithm is superior to the traditional hybrid PSO in the processing time on each production line, and ends up with a smaller solving error.

Table III: Processing time computed by different algorithms.

Algorithm	Production line	Processing time	Error
Our algorithm	Product assembly	352	0.02
	Parts assembly	241	0.01
	Parts processing	157	0.01
Traditional hybrid PSO	Product assembly	387	-11.9
	Parts assembly	253	6.1
	Parts processing	175	7.9

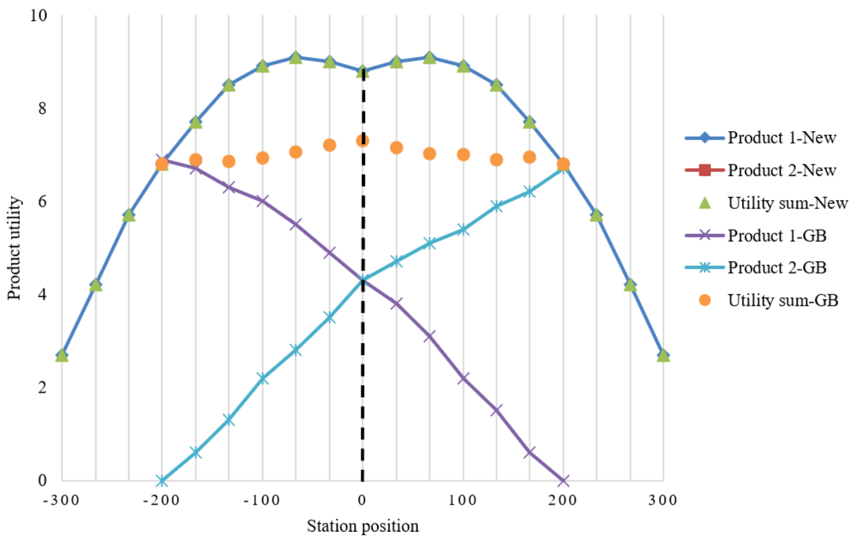

Figure 9: Global optimal product utility vs. Pareto optimal product utility.

Fig. 9 compares the product utilities obtained under the global optimal and Pareto optimal approaches. When two products apply for assembly to the same machine (i.e., the product assembly demands occur at the same time), the product utility of global optimal strategy was slightly lower than that of Pareto optimal strategy, owing to the competition between products. This is because, under Pareto optimal solution, the products and stations maximize their utilities through cooperation. By contrast, the hybrid PSO prioritizes the objectives of independent production lines. This non-cooperative equilibrium strategy could not maximize the overall utility of the job-shop.

6. CONCLUSIONS

This paper applies the GBHPSO to explore the production control strategy for job-shop. Specifically, a job-shop model involving multiple different production lines was established, and the production control ideas were presented, which combine real-time monitoring of delay events and operation sequence adjustment. Then, the production control objectives were defined for parts processing line, parts assembly line, and product assembly line. Next, the workflow of solving job-shop production control problem with GBHPSO was detailed. Through experiments, the Gantt charts of product assembly and parts assembly were obtained, and compared to verify the effectiveness of GBHPSO. Our algorithm was also compared with other methods in iterative losses and processing times. The comparison reveals the superiority of our algorithm in convergence speed and processing time. Finally, the global optimal strategy was contrasted with Pareto optimal strategy in terms of product utility. The results confirm that our algorithm has an advantage in maximizing the overall utility of the job-shop.

REFERENCES

[1] Bissoli, D. C.; Zufferey, N.; Amaral, A. R. S. (2021). Lexicographic optimization-based clustering search metaheuristic for the multiobjective flexible job shop scheduling problem, *International Transactions in Operational Research*, Vol. 28, No. 5, 2733-2758, doi:10.1111/itor.12745

[2] Zhang, Z.; Guan, Z. L.; Zhang, J.; Xie, X. (2019). A novel job-shop scheduling strategy based on particle swarm optimization and neural network, *International Journal of Simulation Modelling*, Vol. 18, No. 4, 699-707, doi:10.2507/IJSIMM18(4)CO18

[3] Park, J.; Nguyen, S.; Zhang, M.; Johnston, M. (2015). Evolving ensembles of dispatching rules using genetic programming for job shop scheduling, Machado, P.; Heywood, M. I.; McDermott, J.; Castelli, M.; Garcia-Sanchez, P.; Burelli, O.; Risi, S.; Sim, K. (Eds.), *Genetic Programming (European Conference on Genetic Programming), LNCS*, Vol. 9025, Springer, Cham, 92-104, doi:10.1007/978-3-319-16501-1_8

[4] Denkena, B.; Schinkel, F.; Pirnay, J.; Wilmsmeier, S. (2021). Quantum algorithms for process parallel flexible job shop scheduling, *CIRP Journal of Manufacturing Science and Technology*, Vol. 33, 100-114, doi:10.1016/j.cirpj.2021.03.006

[5] Genova, K.; Kirilov, L.; Guliashki, V. (2015). A survey of solving approaches for multiple objective flexible job shop scheduling problems, *Cybernetics and Information Technologies*, Vol. 15, No. 2, 3-22, doi:10.1515/cait-2015-0025

[6] Caldeira, R. H.; Gnanavelbabu, A. (2019). Solving the flexible job shop scheduling problem using an improved Jaya algorithm, *Computers & Industrial Engineering*, Vol. 137, Paper 106064, 16 pages, doi:10.1016/j.cie.2019.106064

[7] Pongchairerks, P. (2019). A two-level metaheuristic algorithm for the job-shop scheduling problem, *Complexity*, Vol. 2019, Paper 8683472, 11 pages, doi:10.1155/2019/8683472

[8] Kawaguchi, S.; Kokubo, T.; Fukuyama, Y. (2017). Parallel reactive tabu search for job-shop scheduling problems considering energy management, *2017 IEEE Symposium Series on Computational Intelligence*, 2906-2913, doi:10.1109/SSCI.2017.8280877

[9] Rohaninejad, M.; Sahraeian, R.; Nouri, B. V. (2017). Minimising the total cost of tardiness and overtime in a resumable capacitated job shop scheduling problem by using an efficient hybrid

algorithm, *International Journal of Industrial and Systems Engineering*, Vol. 26, No. 3, 318-343, doi:10.1504/IJISE.2017.084421

[10] Mishra, S. K.; Bose, P. S. C.; Rao, C. S. P. (2017). An invasive weed optimization approach for job shop scheduling problems, *The International Journal of Advanced Manufacturing Technology*, Vol. 91, No. 9-12, 4233-4241, doi:10.1007/s00170-017-0091-x

[11] Cinar, D.; Topcu, Y. I.; Oliveira, J. A. (2015). A taxonomy for the flexible job shop scheduling problem, Migdalas, A.; Karakitsiou, A. (Eds.), *Optimization, Control, and Applications in the Information Age*, Springer, Cham, 17-37, doi:10.1007/978-3-319-18567-5_2

[12] Renna, P. (2015). Deteriorating job scheduling problem in a job-shop manufacturing system by multi-agent system, *International Journal of Computer Integrated Manufacturing*, Vol. 28, No. 9, 936-945, doi:10.1080/0951192X.2014.928747

[13] Mokhtari, H.; Dadgar, M. (2015). Scheduling optimization of a stochastic flexible job-shop system with time-varying machine failure rate, *Computers & Operations Research*, Vol. 61, 31-45, doi:10.1016/j.cor.2015.02.014

[14] Yazdani, M.; Zandieh, M.; Tavakkoli-Moghaddam, R.; Jolai, F. (2015). Two meta-heuristic algorithms for the dual-resource constrained flexible job-shop scheduling problem, *Scientia Iranica*, Vol. 22, No. 3, 1242-1257

[15] Yahouni, Z.; Mebarki, N.; Sari, Z. (2019). Evaluation of a new decision-aid parameter for job shop scheduling under uncertainties, *RAIRO – Operations Research*, Vol. 53, No. 2, 593-608, doi:10.1051/ro/2017073

[16] Guo, J. (2019). Simulation study on flexible job shop scheduling optimization of multi-process planning routes considering energy consumption, *Academic Journal of Manufacturing Engineering*, Vol. 17, No. 3, 164-172

[17] Nababan, E. B.; Sitompul, O. S.; Sianturi, J. (2019). Analysis of tabu list length on job shop scheduling problem, *Journal of Physics: Conference series*, Vol. 1235, Paper 012047, 7 pages, doi:10.1088/1742-6596/1235/1/012047

[18] Jahan, M. V.; Dashtaki, M.; Dashtaki, M. (2015). Water cycle algorithm improvement for solving job shop scheduling problem, *Proceedings of the 2015 International Congress on Technology, Communication and Knowledge*, 576-581, doi:10.1109/ICTCK.2015.7582732

[19] Phanden, R. K.; Jain, A. (2015). Assessing the impact of changing available multiple process plans of a job type on mean tardiness in job shop scheduling, *The International Journal of Advanced Manufacturing Technology*, Vol. 80, No. 9-12, 1521-1545, doi:10.1007/s00170-015-7123-1

[20] Driss, I.; Mouss, K. N.; Laggoun, A. (2015). A new genetic algorithm for flexible job-shop scheduling problems, *Journal of Mechanical Science and Technology*, Vol. 29, No. 3, 1273-1281, doi:10.1007/s12206-015-0242-7

[21] Shahrabi, J.; Adibi, M. A.; Mahootchi, M. (2017). A reinforcement learning approach to parameter estimation in dynamic job shop scheduling, *Computers & Industrial Engineering*, Vol. 110, 75-82, doi:10.1016/j.cie.2017.05.026

[22] Ham, A. (2017). Flexible job shop scheduling problem for parallel batch processing machine with compatible job families, *Applied Mathematical Modelling*, Vol. 45, 551-562, doi:10.1016/j.apm.2016.12.034

[23] Bożejko, W.; Gnatowski, A.; Pempera, J.; Wodecki, M. (2017). Parallel tabu search for the cyclic job shop scheduling problem, *Computers & Industrial Engineering*, Vol. 113, 512-524, doi:10.1016/j.cie.2017.09.042

[24] Al Aqel, G.; Li, X.; Gao, L. (2019). A modified iterated greedy algorithm for flexible job shop scheduling problem, *Chinese Journal of Mechanical Engineering*, Vol. 32, No. 1, Paper 21, 11 pages, doi:10.1186/s10033-019-0337-7

[25] Wang, J.; Liu, Y.; Ren, S.; Wang, C.; Wang, W. (2021). Evolutionary game based real-time scheduling for energy-efficient distributed and flexible job shop, *Journal of Cleaner Production*, Vol. 293, Paper 126093, 16 pages, doi:10.1016/j.jclepro.2021.126093