Int j simul model 16 (2017) 2, 357-367
ISSN 1726-4529 Original scientific paper

THE ALGORITHM AND SIMULATION OF MULTI-OBJECTIVE SEQUENCE AND BALANCING PROBLEM FOR MIXED MODE ASSEMBLY LINE

Yang, B.[*]; Chen, W.[**,#] & Lin, C.[**]
[*] Research Center of Economy of the Upper Reaches of the Yangtze River, Chongqing Technology and Business University, Chongqing 400067, China
[**] Research Center for International Business and Economy, Sichuan International Studies University, Chongqing 400031, China
E-Mail: chenwei@sisu.edu.cn ([#] Corresponding author)

Abstract
In order to solve the balancing problem of product processing in a mixed flow assembly line, a modified genetic algorithm is proposed to optimize the instantaneous load and average load in the assembly line. An improved discrete particle swarm optimization algorithm is used to address the disordered and inefficient sequencing problem in processing products in an assembly line. Through a comprehensive consideration of the operating sequence, minimum production cycle, and the average load and instantaneous load of all workstations, the optimal solution was obtained and its load balancing conditions were studied. Based on the final solution and simulation results, the optimal solution was selected as the assembly line balancing alternative. The sequencing analysis result shows that by introducing the modified discrete PSO algorithm in the sequencing solution seeking in a mixed mode assembly line, the disordered and inefficient multi-objective sequencing problem can be effectively solved. According to the simulation result and calculated result, we set the ratio of the number of workstations to transmission rate as 10 and the product launch intervals as 45 s. Compared to the traditional algorithm, the improved algorithm has a smaller targeted function value, much shorter distance between the optimal solution and the ideal solution, and greater convergence capability.
(Received, processed and accepted by the Chinese Representative Office.)

Key Words: Mixed Flow Line, Multi-Objective, Genetic Algorithm, Particle Swarm Algorithm, Balance, Sequence

1. INTRODUCTION

The mixed flow line is an emerging mode of work with extra-higher production efficiency, by which products of similar construction structures and process requirements can be assembled in the same work line.

In recent years, the scale of production of major enterprises continues to expand, and multiple problems ensue from the constant updating of mixed flow lines, among which the major one is sequencing and balancing of mixed flow assembling. The balancing issue is to properly allocate tasks on the production line so that the overall operating time is minimal and all the workstations have the same operating time. The production sequencing issue is to optimize the product sequence on the assembly line to maximize the efficiency of the production line. Improper handling of the problem will seriously affect material supply and cause workload imbalances [1-6].

At present, the researchers have proposed many fruitful methods to improve the balance and sequencing of the mixed flow line, such as the theoretical analysis method [7-10], the simulation method [11] and the combinatory method [12-17].

In order to solve the balancing problem of product processing in the mixed flow assembly line, a modified genetic algorithm is proposed to optimize the instantaneous load and average load in the assembly line. An improved discrete particle swarm optimization algorithm is used to address the problem of disordered and inefficient sequencing in processing products in the assembly line.

2. ANALYSIS OF THE BALANCING PROBLEM OF MIXED-MODE ASSEMBLY LINE

The mixed assembly line balancing problem (MALBP) is a complex research project considering the mutual influence of different products on the basis of single assembly line balance. When the production rhythm is determined, the number of workstations is minimized by balancing studies, and when the number of workstations on the production line is determined, the production rhythm should be minimized.

2.1 Mathematical model

N products are assembled on the same production line as per the present order. Considering different processing times, products of various types are processed in different orders. For the workload of a given assembly line, the number of workstations that can complete the job on time should first be determined. Then, the number of workstations should be adjusted according to the ongoing simulation.

The quantity demanded for the p^{th} product in a cycle time T is (P = 1, 2, 3, ..., N), and thus the total demand for all products D and the average production rhythm GT can be expressed as:

$$\begin{cases} D = \sum_{p=1}^{N} D_p \\ G_T = T \Big/ \sum_{p=1}^{N} D_p \end{cases} \tag{1}$$

The cycle time T can be divided into several production cycles. In this case, if the quantity demanded for the p^{th} product in every cycle time T is d_p, then we have $d_p = D_p/r$, where r is the common divisor of the product demand. The time of the i^{th} task (i = 1, 2, 3, …, M) of the p^{th} product is t_{pi}, and the minimum number of workstations is obtained according to Eq. (2):

$$Q_{\min} = \frac{\sum_{p=1}^{N} d_p \cdot \sum_{p=1}^{M} t_{pi}}{G_T \cdot \sum_{p=1}^{N} d_p} \tag{2}$$

Subject to multiple factors, the actual number of workstations is sometimes larger than the theoretically calculated value of Q_{min}. If GT is greater than the average operating time of the i^{th} workstation, the task waiting for the i^{th} workstation has to be assigned to the i + 1 workstation so that the current workstation will have a longer idle time when the task is completed, resulting in an increase in the overall assembly time. Therefore, in order to avoid the above problems, Q_{min} should be used to monitor and correct the workstation loads in the simulation process.

The optimization of the balanced load of the workstation is to minimize the load variance of each workstation and to ensure a basically similar mutual load. Therefore, we establish the following optimization equation:

$$\min J = \sqrt{\frac{\sum_{k=1}^{S} \left[\sum_{n=1}^{N} q_n T_{nk} - \sum_{j=1}^{S} \sum_{n=1}^{N} q_n T_{nj} \Big/ S \right]^2}{S}} \tag{3}$$

Meanwhile, we set the following constraint conditions:

$$\begin{cases} T_{nk} = \sum_{i=1}^{N} t_{ni} x_{ij} \quad n = 1,2,3,...N \\ x_{ij} \in \{0,1\} \quad i = 1,2,3,...,N \\ \sum_{n=1}^{N} q_n T_{nk} \le G_T \quad j = 1,2,3,...S \end{cases} \tag{4}$$

where QN is the proportion of the production number of the i^{th} product to all the products'. Constraint Eq. (4) is to guarantee that each workstation processes one task in the same amount of time at the average load of less than GT while satisfying the task priority order.

2.2 Algorithm design

Through a genetic algorithm, we seek the solution to the balancing issue of a mixed-mode assembly line and to optimize it. As the core of the genetic algorithm, the gene string coding consists of a data sequence with the length P. S workstations are assigned with several operation tasks. According to the genetic algorithm, we first randomly generate an initial solution and let the task set be $MA=\{ma_1, ma_2, ..., ma_n\}$. The pre-task and follow-up task of a task ma_i is shown in the overall task sequencing map. We initialize the first workstation, select a single task from MA whose pre-task is 0, and distribute it to workstation s_{tp}. We check whether the cumulative average operating time of the task in s_{tp} has exceeded GT. If it does, the final distribution should be cancelled.

In the initial calculation of the genetic algorithm, due to the large difference between the degrees of individual fitness, there is a small portion of individuals with higher fitness which reduce the overall population fitness scale and tend towards local convergence. Therefore, we optimize the conversion computation of the fitness scale, and the fitness function $f(J_1)$ can be expressed as:

$$f(J_1) = 1 - J_1/\mu \tag{5}$$

Then, it is converted into:

$$f_1(J_1) = f_{av}(J_1) + y\left[f(J_1) - f_{av}(J_1)\right] \tag{6}$$

where $f_{av}(J_1)$ and $f_1(J_1)$ are the respective mean fitness and the converted fitness; y is the conversion coefficient.

We select the population for crossover and let the crossover probability be pc. The gene string decoding between any two individuals should satisfy the sequential order of tasks on the assembly line. Fig. 1 shows the sequence of operations for the hybrid assembly lines of multiple products. The $ch1$ and $ch2$ gene strings are selected as the task assignment scheme. If $ch1$ and $ch2$ have a single crossover point, the 11^{th} task and the 15^{th} task will be assigned to the same workstation, which is obviously unreasonable in terms of resource waste in the entire assembly line.

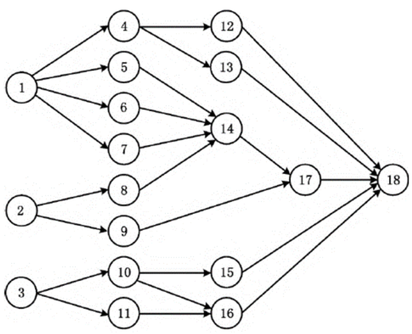

Figure 1: Operation sequence of multi-variety mixed model assembly.

To address the above problems, this paper proposes a method of verification and optimization after crossover, so as to ensure the effectiveness of the computational result. When the verification is done in the order of the task, invalid gene strings are reallocated and converted into valid ones. The population size is set within 40-100 depending on the actual situation.

EM-Plant is used as the simulation software. Fig. 2 is a typical example of EM-Plant-based modelling. There are four workstations (workstations 1 4) for the products to be processed from left to right. EventController is used to record the event time; ProcTime1 – ProcTime4 are the task times in the four respective workstations. During calculation, the task distribution results are programmed with Simtalk in terms of product task time, distribution task order and other elements.

The mean utilization rate of workstations (represented by J_2) is used as the evaluation index of the balancing issue in the mixed flow assembly line. The higher J_2 is, the higher the balancing rate becomes. The optimization function of the balancing of a mixed-mode assembles line can be expressed as:

$$\min J = \omega_1 J_1 + \omega_2 \left(1 - J_2\right) \tag{7}$$

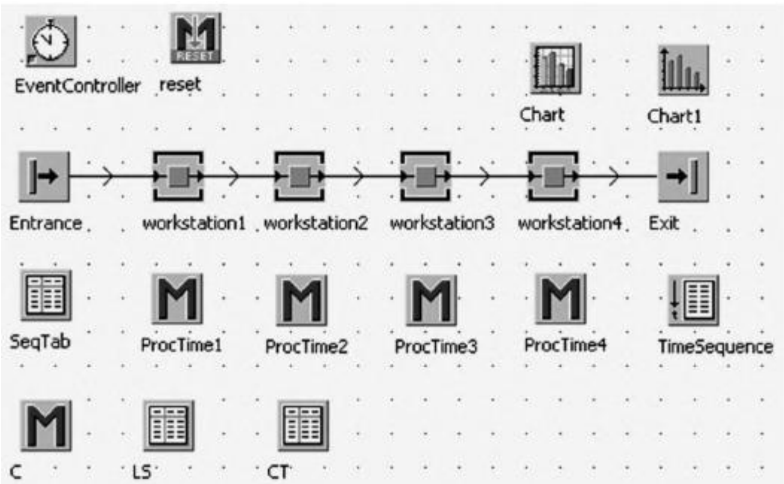

Figure 2: Simulation model of mixed-model assembly lines.

2.3 Simulation result analysis

We conducted a simulation analysis of the solution of the hybrid assembly line balancing problem using a genetic algorithm in order to verify its validity. Three kinds of products were assembled in one mixed-mode assembly line, whose expected production outputs are respectively $D_1 = 400$, $D_2 = 200$ and $D_3 = 300$. The average was higher than $GT = 31$ s. The comprehensive task sequences of the three kinds of products are shown in Fig. 3.

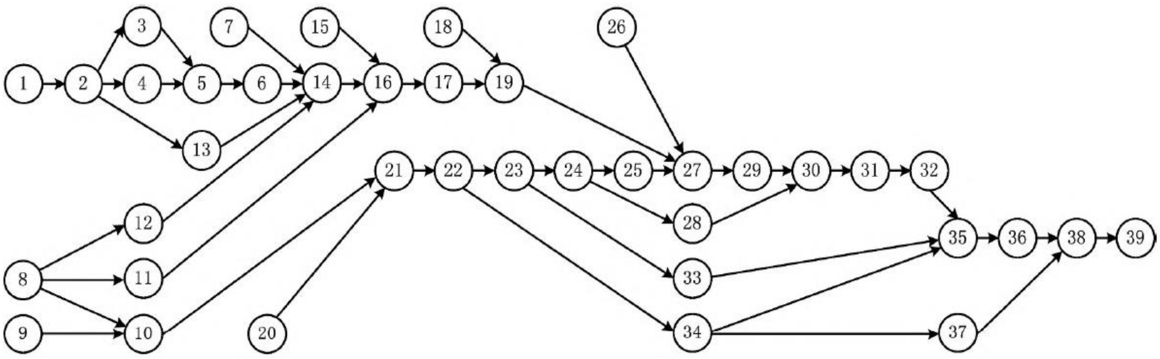

Figure 3: Comprehensive operation sequences of three kinds of products.

The demand for each product in a single production cycle is $d_1 = 4$, $d_2 = 2$, and $d_3 = 3$, respectively. According to Eq. (2), $Q_{min} = 6$. Table I shows the optimal solution of tasks and average loads obtained according to the genetic algorithm. As can be seen from the table, the average loads of the six workstations are lower than *GT*, which verifies the feasibility of the average-load-based workstation number determination. After inputting the results into the simulation model, we find that the instantaneous load of a small portion of products exceeds *GT* and that the maximum value of the actual assembly line rhythm reaches 37.1 s. As a result, the production rate does not fulfil the requirement of the production plan. Based on the above calculation results, it can be concluded that we should add an extra workstation to lower the assembly rhythm.

Table I: Optimal solution of genetic algorithm.

Coding	1 1 1 1 1 2 2 2 2 3 4 3 4 4 2 5 5 2 6 2 3 4 5 5 6 2 6 5 6 6 6 6 5 4 6 6 6 6 6				
Number	1	2	3	4	5	6
Operation tasks	1, 2, 3, 4, 5	6, 7, 8, 9, 15, 18, 20, 26	10, 12, 21	11, 13, 14, 22, 34	16, 17, 23, 24, 28, 33	19, 25, 27, 29, 30, 31, 32, 35, 36, 37, 38, 39
Average load/s	30.28	29.22	29.78	30.05	29.33	29.15

We reset the number of workstations as seven, and re-calculate *GT* as 27.18 s according to Eq. (2). We studied instantaneous loads of the assembly line as load distribution schemes vary, and the simulation model is shown in Fig. 4.

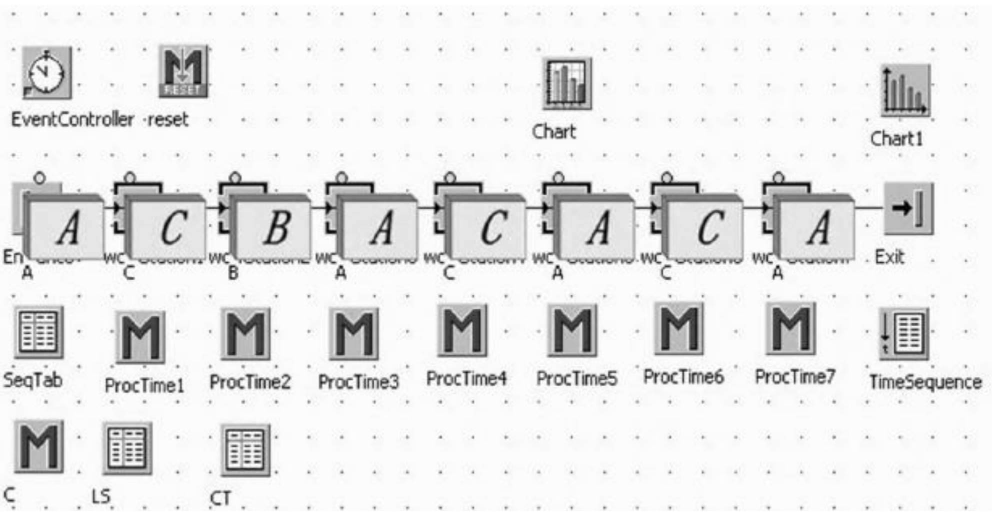

Figure 4: Momentary duty of mixed-model assembly lines.

Among a total of eighteen schemes generated on the basis of the calculation result, there are only two schemes whose rhythms are higher than the design rhythm, while the remaining schemes have lower rhythms than the design rhythm, which proves the feasibility of using seven workstations. The average utilization rate of the seven workstations is set as target J_2, and the optimization objective function value is calculated according to Eq. (7). The relationship curves of the objective function value and the distribution scheme are shown in Fig. 5, showing that when J_1 takes the minimum value, J_2 fails to reach the maximum value. In light of the fact that the calculation principle of the genetic algorithm is based on the workshop integrated operation sequence, and the simulation calculation is based on the instantaneous load balance, the calculation result and the simulation result are of a certain deviation.

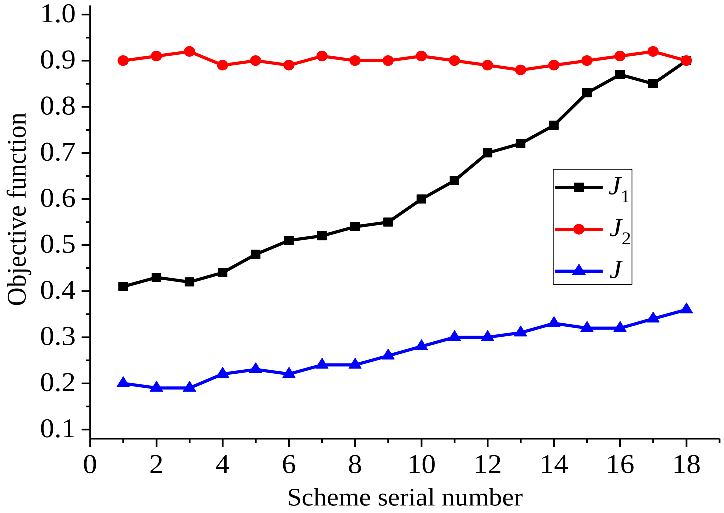

Figure 5: Curve of objective function with distribution scheme.

After comparing the J values among the eighteen schemes, scheme 2 shows the smallest J value, and thus it is regarded as the optimal balanced distribution scheme in the mixed assembly line. The work tasks and the corresponding average load for each workstation in scheme 2 are shown in Table II.

Table II: Optimal distribution scheme and average load.

Workstation	Operation tasks	Average load/s
1	1, 2, 3, 4, 14, 17, 20, 25	25.73
2	5, 7, 3	26.046
3	6, 8, 10, 11,12	24.379
4	21, 22	26.69
5	9, 15, 16, 34, 38	24.163
6	18, 23, 24, 26, 28	25.51
7	19, 27, 29, 30, 31, 32, 33, 35, 36, 37, 39	25.88

3. ANALYSIS OF THE SEQUENCING PROBLEM OF MIXED-MODE ASSEMBLY LINE

3.1 The improved PSO algorithm

The sequencing problem of the hybrid assembly line is another core issue to be addressed. In this paper, the discrete particle swarm optimization (PSO) algorithm is modified into a discrete particle swarm optimization algorithm that is applied to the sequencing analysis of hybrid assembly lines. The ideal result achieved by the particle swarm optimization algorithm is the search for a global optimal solution without the convergence of initial search and calculation parameters. Because of the lack of diversity and computational stability of the traditional PSO algorithm, this paper adds an adaptive particle escape strategy to ensure an effective global search.

The entire search space has A dimensions and I particles as expected. The i^{th} generation of particles can be expressed as:

$$X_i(t) = \left[x_{i1}(t), x_{i2}(t), x_{i3}(t), \ldots, x_{iA}(t)\right] \tag{8}$$

where $X_i(t)$ represents a possible solution of the objective function. The global optimization model of the particle swarm can be expressed as follows:

$$v_{ij}(t) = wv_{ij}(t-1) + c_1 r_1 \left[p_{ij}(t) - x_{ij}(t) \right] + c_2 r_2 \left[g_{ij}(t) - x_{ij}(t) \right] \tag{9}$$

$$x_{ij}(t) = x_{ij}(t-1) + v_{ij}(t) \tag{10}$$

where T is the genetic algebra of the population; w is the inertia weight; c_1 and c_2 are the influencing factors; r_1 and r_2 are random numbers between (0, 1). The PSO algorithm can effectively represent the mapping between the solution and the particles, and the convergence threshold will affect the convergence rate of the algorithm and the optimal solution of the acquisition. If the convergence threshold is represented by T_j, the escape frequency $F_j(t)$ can be expressed as:

$$F_j(t) = F_j(t-1) + \sum_{i=1}^{N} b_{ij}(t) \tag{11}$$

$$b_{ij}(t) = \begin{cases} 0 & v_{ij}(t) > T_j \\ 1 & v_{ij}(t) < T_j \end{cases} \tag{12}$$

The relationship between the global search of the PSO algorithm and the local search can be adjusted by the Eqs. (11) and (12). Based on the above analysis, the steps to improve the discrete particle swarm optimization algorithm are as follows: (1) initialize relevant parameters; (2) generate $X_i(t)$; (3) calculate the objective function, the global threshold and the optimal solution; (4) calculate self-adaptive escape of particles; (5) repeat iterations until the result of the calculation reaches the set threshold.

3.2 Simulation results analysis

In the mixed-mode assembly line, we can adjust parameters such as the number of workstations, assembly line transmission speed, and product launch interval according to the situation. Fig. 6 is the effect of the solution with the ratio of the L number of workstations to transmission velocity (L/V) and interval of product launch. As can be seen from Fig. 6, as L/V increases, the objective function value decreases; with an increase in the product launch interval, the objective function value first decreases and then increases. The smaller the objective function is, the higher the production efficiency of the assembly line is. Therefore, we determine $L/V = 10$ and the interval as 45 s.

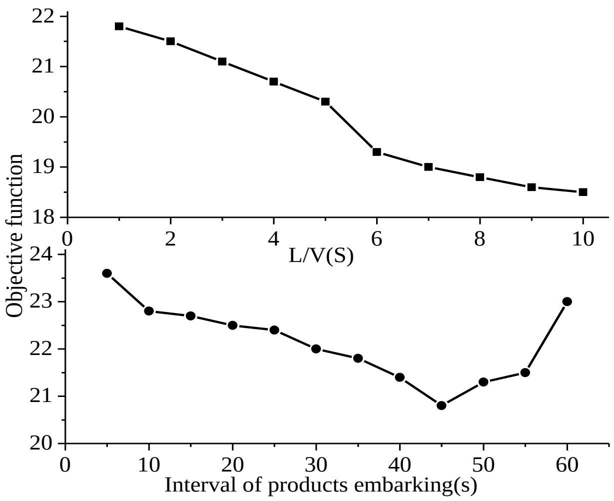

Figure 6: Effect of the solution with L/V and interval of products.

In order to verify the effectiveness of the proposed method, we compare the proposed method with the basic PSO algorithm with four parameters, as shown in Table III. K_1 is the condition coefficient for adjusting T_j and $F_j(t)$; K_2 is the correlation coefficient controlling convergence rate.

Table III: Parameter setting of four kinds of algorithms.

Algorithm	w	c_1	c_2	K_1	K_2
PSO1	[0.4, 1.2]	1.8	1.8	—	—
PSO2	0.9	1.8	1.8	—	—
IE-PSO1	[0.4, 1.2]	1.8	1.8	4	5
IE-PSO2	0.9	1.8	1.8	4	5

The distance between the optimal solution and the ideal solution is expressed by the difference distance method:

$$dd = \sqrt{\sum_{h=1}^{H} \left(f_o(h) - f_{\min} \right)^2} \tag{13}$$

where H is the number of optimal solutions; $f_o(h)$ is the h^{th} optimal solution; f_{min} is the ideal solution. This is calculated ten times based on the four methods while leaving the initial condition unchanged. Fig. 7 shows the distance between the optimal solution and the ideal solution obtained in this way.

It can be seen from the Fig. 7 that compared with the traditional algorithm, the distance between the optimal solution and the ideal solution obtained by the improved algorithm is obviously reduced and the convergence ability is higher. This is because the algorithm proposed in this paper increases the diversity of original particles, and thus increases the global search capacity of the system.

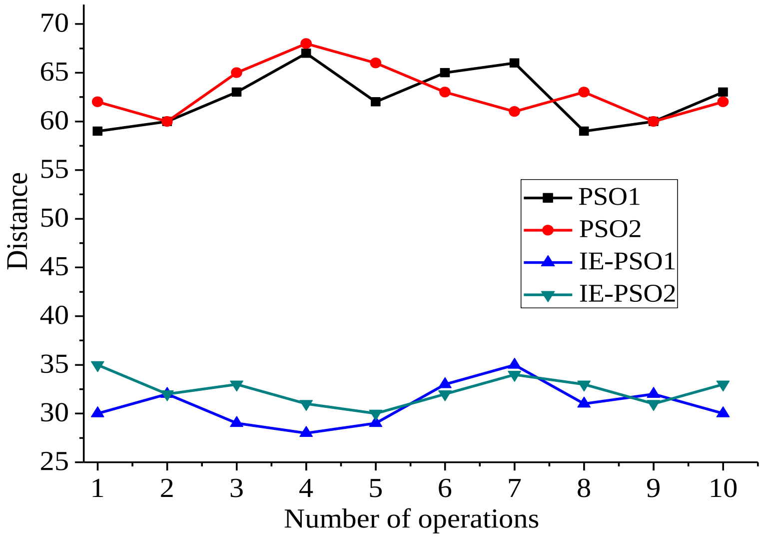

Figure 7: Relation curve of number of operations and distance with four kinds of algorithms.

The following evaluation parameters are proposed to compare the improved PSO algorithm with the traditional algorithm. The minimum objective function O_{min} is:

$$O_{\min} = \sum_{z=1}^{Z} \min f(n) / Z \tag{14}$$

where Z is the number of operations. The maximum objective function and the average objective function are:

$$\begin{cases} O_{\max} = \sum_{z=1}^{Z} \max f(n) / Z \\ O_{mean} = \sum_{z=1}^{Z} f(n) / Z \end{cases} \tag{15}$$

Standard deviation:

$$SD = \sqrt{\sum_{z=1}^{Z} \left(f(n) - O_{mean} \right)} \Big/ Z \tag{16}$$

The calculation results of Eqs. (14) to (16) are collated into statistical data, as shown in Table IV, where $w = 0.8$. The table shows that compared with the traditional algorithm, the algorithm proposed in this paper has a smaller objective function, and the improved PSO algorithm performs better in terms of the fluctuation of relative deviation. When the value of w is gradually increased from 0.1 to 1.0, we find that the algorithm output is satisfactory when $w = 0.6\text{-}0.9$.

Table IV: Calculation results of different evaluation index.

Algorithm	O_{min}	O_{max}	O_{mean}	SD
PSO1	31.16	42.63	35.88	3.485
PSO2	29.98	41.91	34.79	3.394
IE-PSO1	20.66	30.48	25.91	0.903
IE-PSO2	20.17	29.95	26.03	1.277

The improved discrete PSO algorithm is compared with the traditional genetic algorithm in calculating the evolution curve of the multi-objective hybrid assembly line. The comparison curve is shown in Fig. 8, where the crossover probability is set as 0.8, the mutation probability is 0.15, and the genetic algebra is set as 200 generations.

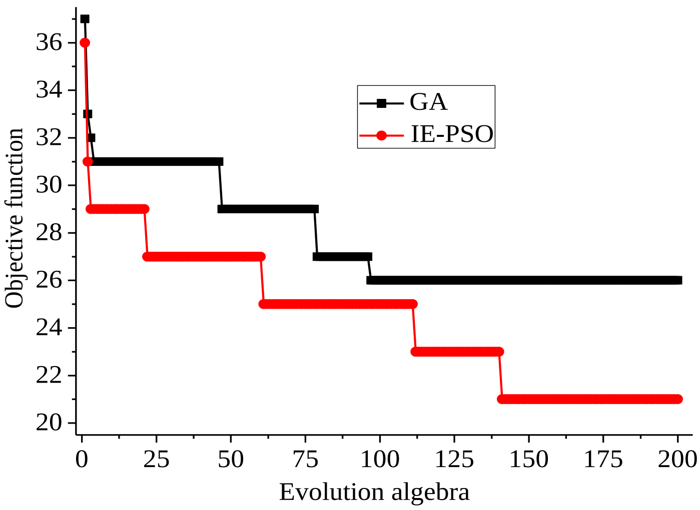

Figure 8: Evolution algebra curve of GA and IE-PSO algorithm.

It is seen from the figure that the improved discrete PSO algorithm is better than the traditional GA algorithm, both in terms of evolutionary speed and optimal solution. This is because of the relative simplicity of the evolutionary model of the PSO algorithm. With crossover and mutation, the traditional GA algorithm is more likely to obtain a local optimal solution and mutation in the process of calculation.

4. CONCLUSION

Aiming at the problem of product processing in the mixed flow line, an improved genetic algorithm is proposed to optimize the instantaneous load and average load in an assembly line. In order to solve the problem of product processing in the operation line, an improved discrete particle swarm optimization algorithm is used to solve the problem of sequence chaos and low efficiency. We draw the following conclusions:

(1) Through a comprehensive consideration of the operating sequence, minimum production cycle, and the average load and instantaneous load of all workstations, the optimal solution was obtained and its load balancing conditions were studied. Based on the final

solution and simulation results, the optimal solution was selected as the assembly line balancing alternative. The optimal values achieved by respective mathematical modelling and simulation calculation can act as effective results both statically and dynamically.

(2) The sequencing analysis result shows that by introducing the modified discrete PSO algorithm in the sequencing solution seeking in a mixed mode assembly line, the disordered and inefficient multi-objective sequencing problem can be effectively solved. According to the simulation result and calculated result, we set the ratio of the number of workstations to transmission rate as 10 and the product launch intervals as 45 s. Compared to the traditional algorithm, the improved algorithm has a smaller targeted function value, much shorter distance between the optimal solution and the ideal solution, and greater convergence capability.

ACKNOWLEDGEMENT

The authors acknowledge funding from the National Social Science Foundation of China (Project No. 16XJY014) and the key project of humanities and social sciences from the Chongqing Municipal Education Commission (Project No. 16SKGH103), as well as the contributions from all partners of the mentioned projects. Besides, Chen Wei is the corresponding author and can be contacted at: chenwei@sisu.edu.cn.

REFERENCES

[1] Sabuncuoglu, I.; Erel, E.; Tanyer, M. (2000). Assembly line balancing using genetic algorithms, *Journal of Intelligent Manufacturing*, Vol. 11, No. 3, 295-310, doi:10.1023/A:1008923410076

[2] Haq, A. N.; Rengarajan, K.; Jayaprakash, J. (2006). A hybrid genetic algorithm approach to mixed-model assembly line balancing, *The International Journal of Advanced Manufacturing Technology*, Vol. 28, No. 3, 337-341, doi:10.1007/s00170-004-2373-3

[3] Hwang, R. K.; Katayama, H.; Gen, M. (2008). U-shaped assembly line balancing problem with genetic algorithm, *International Journal of Production Research*, Vol. 46, No. 16, 4637-4649, doi:10.1080/00207540701247906

[4] Simaria, A. S.; Vilarinho, P. M. (2004). A genetic algorithm based approach to the mixed-model assembly line balancing problem of type II, *Computers & Industrial Engineering*, Vol. 47, No. 4, 391-407, doi:10.1016/j.cie.2004.09.001

[5] Akpınar, S.; Bayhan, G. M. (2011). A hybrid genetic algorithm for mixed model assembly line balancing problem with parallel workstations and zoning constraints, *Engineering Applications of Artificial Intelligence*, Vol. 24, No. 3, 449-457, doi:10.1016/j.engappai.2010.08.006

[6] Akgündüz, O. S.; Tunalı, S. (2010). An adaptive genetic algorithm approach for the mixed-model assembly line sequencing problem, *International Journal of Production Research*, Vol. 48, No. 17, 5157–5179, doi:10.1080/00207540903117857

[7] Özcan, U.; Toklu, B. (2009). Balancing of mixed-model two-sided assembly lines, *Computers & Industrial Engineering*, Vol. 57, No. 1, 217-227, doi:10.1016/j.cie.2008.11.012

[8] Jin, M.; Wu, S. D. (2003). A new heuristic method for mixed model assembly line balancing problem, *Computers & Industrial Engineering*, Vol. 44, No. 1, 159-169, doi:10.1016/s0360-8352(02)00190-0

[9] Hwang, R. K.; Katayama H. (2009). A multi-decision genetic approach for workload balancing of mixed-model U-shaped assembly line systems, *International Journal of Production Research*, Vol. 47, No. 14, 3797-3822, doi:10.1080/00207540701851772

[10] Ding, F.-Y.; Tolani, R. (2003). Production planning to support mixed-model assembly, *Computers & Industrial Engineering*, Vol. 45, No. 3, 375-392. doi:10.1016/s0360-8352(03)00071-8

[11] Mendes, A. R.; Ramos, A. L.; Simaria, A. S.; Vilarinho, P. M. (2005). Combining heuristic procedures and simulation models for balancing a PC camera assembly line, *Computers & Industrial Engineering*, Vol. 49, No. 3, 413-431, doi:10.1016/j.cie.2005.07.003

[12] Rahimi-Vahed, A. R.; Rabbani, M.; Tavakkoli-Moghaddam, R.; Torabi, S. A.; Jolai, F. (2007). A multi-objective scatter search for a mixed-model assembly line sequencing problem, *Advanced Engineering Informatics*, Vol. 21, No. 1, 85-99, doi:10.1016/j.aei.2006.09.007

[13] Miltenburg, J.; Sinnamon, G. (1989). Scheduling mixed-model multi-level just-in-time production systems, *International Journal of Production Research*, Vol. 27, No. 9, 1487-1509, doi:10.1080/00207548908942636

[14] Bukchin, Y.; Rabinowitch, I. (2006). A branch-and-bound based solution approach for the mixed-model assembly line-balancing problem for minimizing stations and task duplication costs, *European Journal of Operational Research*, Vol. 174, No. 1, 492-508, doi:10.1016/j.ejor.2005.01.055

[15] Zacharia, P. T.; Nearchou, A. C. (2012). Multi-objective fuzzy assembly line balancing using genetic algorithms, *Journal of Intelligent Manufacturing*, Vol. 23, No. 3, 615-627, doi:10.1007/s10845-010-0400-9

[16] Boysen, N.; Fliedner, M.; Scholl, A. (2008). Sequencing mixed-model assembly lines to minimize part inventory cost, *OR Spectrum*, Vol. 30, No. 3, 611-633, doi:10.1007/s00291-007-0095-2

[17] Supsomboon, S.; Vajasuvimon, A. (2016). Simulation model for job shop production process improvement in machine parts manufacturing, *International Journal of Simulation Modelling*, Vol. 15, No. 4, 611-622, doi:10.2507/IJSIMM15(4)3.352