An Assembly Line Multi-Station Assembly Sequence Planning Method Based on Particle Swarm Optimization Algorithm

Shuan-Jun Song[1,2*], Cheng-Hong Qiu[1], Long-Guang Peng[1], Sheng Hu[1]

[1] School of Mechanical & Electrical Engineering, Xi'an Polytechnic University, Xi'an 710048, China
songshuanjun@126.com, 1067296865@qq.com, 275149881@qq.com, husheng564@163.com
[2] Xi'an key laboratory of modern intelligent textile equipment, Xi'an 710048, China

Abstract. Aiming at the problem that the existing assembly sequence planning methods are difficult to meet the multi-station assembly requirements of assembly line, an assembly sequence planning method of assembly line considering the constraints of station sequence and station capability is proposed. The multi-station assembly sequence model is established to describe the allocation scheme and assembly sequence of parts. The conditions and generating rules of feasible assembly sequence are given. The assembly time variance of each station is used as the fitness function, and the particle swarm optimization (PSO) algorithm is designed. Taking an engineering vehicle assembly as an example, the optimal integration solution of multi-station assembly sequence and job assignment is obtained by using this algorithm, and the validity of the model is verified.

Keywords: assembly line, multi-station assembly model, particle swarm optimization, assembly sequence planning, job assignment

1 Introduction

1.1 Background and Related Work

Assembly sequence planning (ASP) is the process of designing and evaluating the proper assembly sequence of components on an assembly. As for the complex assembly such as aircraft, automobiles, it is usually need to complete the assembly through an assembly line. This kind of complex assembly includes a large number of parts, complex structure and complex assembly relationship between parts. The assembly time of its products accounts for about 50% of the whole production time. Assembly line is a typical multi-station serial structure. The assembly tasks performed on the station have a sequential relationship and the station has selectivity to the assembly ability of parts. In order to improve the operation efficiency of the assembly, the assembly time of each station is usually required to be as consistent as possible. Therefore, for complex assemblies assembled on the assembly line, ASP should not only consider the assembly relationship constraints between parts, but also consider the sequence constraints of each station, station capacity constraints and production line balance, which is of great significance for enterprises to shorten the product production cycle.

ASP problem belongs to NP-hard problem. The general solution is the following three steps. Firstly, the expression of the assembly relationship is constructed (relation sequence diagram, component connection diagram, direction matching diagram and space constraint diagram, assembly priority diagram, reachability matrix, interference matrix). Then a feasible assembly sequence is generated by some method and optimize a feasible assembly sequence to a certain goal by exact calculation method or heuristic calculation method. Finally, the optimal assembly sequence scheme or near-optimal scheme is obtained.

In recent years, many scholars have carried out research on ASP and made great progress. Hsu YY describes the relationship between components through relational sequence diagram [1]. Zhou use Petri net to describe the geometric relationship between components [2]. Lin Use component connection diagram, direction matching diagram, and space constraint diagram to express the priority constraint relationship of components [3]. Wang Establishing the priority relationship of aircraft landing gear assembly by using assembly priority map [4]. The literature [5] uses the interference matrix to represent the assembly interference of the components in the x, y, z direction. The assembly sequence can be generated by Disassembly method [6], priority constraint method [7]. In the assembly sequence optimization, accurate calculation methods include cut set method [8], knowledge reasoning method [9], geometric reasoning method [10], assembly subset detection method [11]. However, as the number of assembly parts increases, combined explosions often occurs. To solve such problems, heuristic algorithms such as genetic algorithm [12], ant colony algorithm [13], particle swarm algorithm [14-15], firefly

---

algorithm [16], harmony search algorithm [17], discrete artificial bee colony algorithm [18], discrete fireworks algorithm [19], improved harmony search algorithm [20] and the Genetic algorithm [21] are used to complete the problem. Ab Rashid proposes an integrated optimization method for multi-objective assembly sequence planning and assembly line balancing based on particle swarm optimization algorithm [15]. Defersha integrates balancing and sorting into consideration, proposes a mixed integer linear programming model for mobile conveyors, and designs an embedded genetic algorithm for multi-stage linear programming [21]. A genetic algorithm based on disassembly matrix is proposed by Pedraza to minimize the number of assembly operations [22]. To minimize assembly cost, Sasiadek proposes an assembly sequence planning method [23].

However, most of the current researches are mostly on the ASP problem in a single-station environment. In reality, the assembly of complex assemblies needs to be carried out in a multi-station assembly line environment. The constraints for ASP issues in this environment can be summed as: 1) Assembly priority relationship and interference relationship between assembly components; 2) The number of conversions of assembly tools and fixtures; 3) The sequence of each station on the assembly line; 4) Assembly capacity of each station; 5) The workload of each station on the assembly line is balanced to the expected ideal situation, while maximizing the productivity of the assembly line.

It can be seen that, compared with the assembly line ASP problem in multi-station environment, single-station ASP problem is not considered the effect of multi-station order constraint, station capacity difference and station allocation on production line productivity. And it is prone to pseudo-optimal solutions that do not meet the station requirement. Wang [4] and Yuan [24] used particle swarm optimization and fruit fly optimization algorithms respectively to assigned and optimized the aircraft landing gear assembly sequence at five stations. However, the problem of station order is not considered in the optimization, and the pseudo-optimal solution that does not meet the requirements of the station is prone to occur. For the multi-station assembly sequence planning, Liu [5] proposed an improved particle swarm optimization algorithm, but assembly sequence planning and assembly line balance integration is not considered, so it still has some limitations.

1.2 Research Goal and Purposes

Aim at the features for multi-station assembly environments, an assembly line multi station ASP method based on PSO algorithm is proposed in this paper. This method comprehensively considers the feasibility of assembly sequence, station capacity constraints, station sequence constraints, station assembly time balance and so on. Firstly, the assembly sequence model in the assembly line multi-station environment is established, and the interference matrix and the station capacity matrix are constructed to judge the feasibility of assembling sequence model. Secondly, the balance of the task completion time (assembly operation time, tool conversion time, direction change commutation time) of each station is taken as the fitness function. Based on this, the particle swarm optimization algorithm for multi-station assembly line ASP is designed. Finally, the multi-station ASP method is used to optimize the assembly process of a product (17 parts and 8 work stations) and the optimal assembly sequence and station distribution solution is obtained. Result shows that the proposed method effectively improves the assembly production effectiveness.

2 Assembly Sequence Model

2.1 Assembly Priority Relationship

In the assembly operation, the assembly priority of each component can be represented by a multi-level hierarchical directed graph and a reachable matrix. Suppose the directed graph D = (V, A), which $V = \{v_1, v_2, ..., v_m\}$ represents a collection of parts, v_i represents a part labeled i, $i \in [1, m]$, m represents the number of parts; $A = \{a_1, a_2, ..., a_n\}$ represents a set of directed arcs with part assembly operations, n is the number of directed arcs.

The formation of assembly priority relationship can be summarized as follows. Firstly, the directed graph of the assembly is established, and the assembly relationship reachable matrix M is obtained according to the directed graph. Then use the Interpretative Structural Modeling Method (ISM) method to obtain the assembly relationship priority map. Take an engineering vehicle model as an example (Fig. 1). The engineering vehicle model has 17 main parts, which are indicated by v_i, $i = 1, 2, ..., 17$. The connection relationship of each part can be represented by a directed graph shown in Fig. 2. Then according to the directed graph, the ISM method is used to obtain

the component assembly priority relationship diagram with the hierarchical relationship of the parts, as shown in Fig. 3. When analyzing and evaluating the assembly sequence, the assembly priority map can be transformed into an associable precedence matrix (APM) with the following conversion rules.

$$M_{APM} = \{b_{ij}\}, i, j = 1, 2, ..., m \qquad (1)$$

Among them, $b_{ij} = \begin{cases} 1, v_i \text{ to } v_j \text{ is reachable} \\ 0, v_i \text{ to } v_j \text{ is not reachable} \end{cases}$.
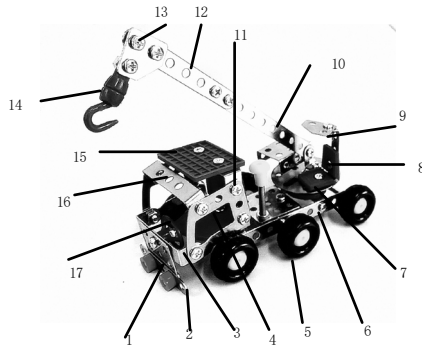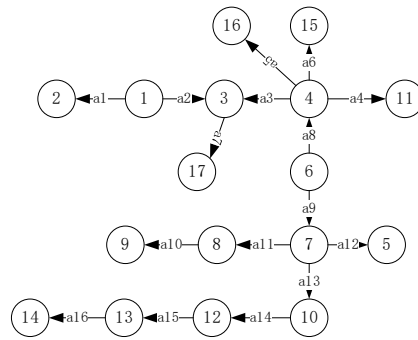

Fig. 1. The engineering vehicle model

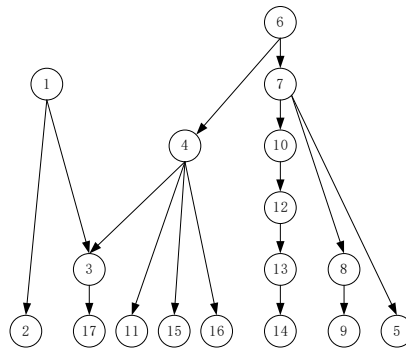**Fig. 2.** The directed graph of assembly

Fig. 3. The assembly priority map of Engineering vehicle model

2.2 Assembly Interference Relationship

When assembling components, in addition to considering the assembly priority relationship of the assembled parts, consideration should also be given to the interference of the components in the assembly space. The mutual interference of the components during assembly can be expressed by the interference matrix AM:

$$A_M = \begin{array}{c} \\ v_1 \\ v_2 \\ ... \\ v_m \end{array} \begin{array}{cccc} v_1 & v_2 & ... & v_m \\ \left[\begin{array}{cccc} a_{d11} & a_{d12} & ... & a_{d1m} \\ a_{d21} & a_{d22} & ... & a_{d2m} \\ ... & ... & a_{dij} & ... \\ a_{dm1} & a_{dm2} & ... & a_{dmm} \end{array}\right] \end{array} . \qquad (2)$$

Where, d is the directions of x, y, z in three dimensions, a_{dij} is the space interferometry relations between v_i and v_j. If it has interferometry relations, the $a_{dij}=1$, otherwise 0.

$$D_i = \bigcap_{j=1}^{i} A_M (v_i, v_j) \ . \tag{3}$$

Whether there is spatial interference in the assembly sequence, and it can be judged whether the assembly direction set of the parts to be assembled is an empty set. The specific solving process is shown in formula (3). When $D_i = \varnothing$, The part to be assembled has no interfere with the assembled sequence. If $D_i \neq \varnothing$, it shows that an interference relationship occurs and the assembly sequence is not feasible.

2.3 Station Ability and Assembly Tool Information

In a multi-station assembly line environment, the assembly activities that can be completed at each station vary depending on factors such as assembly tools and fixtures. A viable assembly sequence needs to meet the requirements of the station ability, which can be represented as C_M.

$$C_M = \begin{array}{c} \\ v_1 \\ v_2 \\ \cdots \\ v_m \end{array} \begin{array}{cccc} s_1 & s_2 & \cdots & s_q \\ \left[\begin{array}{cccc} c_{11} & c_{12} & \cdots & c_{1q} \\ c_{21} & c_{22} & \cdots & c_{2q} \\ \cdots & \cdots & \cdots & c_{ik} & \cdots \\ c_{m1} & c_{m2} & \cdots & c_{mq} \end{array} \right] \end{array} . \tag{4}$$

Where $c_{ik} = 0$ or 1, $c_{ik} = 0$ represents v_i cannot be assembled on s_k. $c_{ik} = 1$ represents v_i can be assembled on s_k. Suppose the assignment matrix corresponding to an assembly sequence is $\tilde{C}_M$, then

$$\tilde{C}_M = \begin{array}{c} \\ v_1 \\ v_2 \\ \cdots \\ v_m \end{array} \begin{array}{cccc} s_1 & s_2 & \cdots & s_q \\ \left[\begin{array}{cccc} \tilde{c}_{11} & \tilde{c}_{12} & \cdots & \tilde{c}_{1q} \\ \tilde{c}_{21} & \tilde{c}_{22} & \cdots & \tilde{c}_{2q} \\ \cdots & \cdots & \cdots & \tilde{c}_{ik} & \cdots \\ \tilde{c}_{m1} & \tilde{c}_{m2} & \cdots & \tilde{c}_{mq} \end{array} \right] \end{array} . \tag{5}$$

Where q is the number of stations, $\tilde{c}_{ik} = 0$ represents that the part is not assigned to station s_k for assembly, $\tilde{c}_{ik} = 1$ represents that the part is assigned to station s_k for assembly. Taking into account the effective use of resources in actual production, it assumes that each part can only be assigned to one station assembly and the station does not have free time. Therefore, the station allocation matrix element $\tilde{c}_{ik}$ should be satisfy following constraints.

$$s.t \begin{cases} \sum_{k=1}^{q} \tilde{c}_{ik} = 1 \\ \sum_{i=1}^{m} \tilde{c}_{ik} \geq 1 \end{cases} . \tag{6}$$

Similarly, assembly tool information can be represented by the assembly tool matrix T, if parts v_1 needs tool T_j assembly, $t_{ij} = 1$, otherwise $t_{ij} = 0$.

$$T = \begin{matrix} & T_1 & T_2 & ... & T_e \\ v_1 & \begin{bmatrix} t_{11} & t_{12} & ... & t_{1e} \\ v_2 & t_{21} & t_{22} & ... & t_{2e} \\ ... & ... & ... & t_{ij} & ... \\ v_m & t_{m1} & t_{m2} & ... & t_{me} \end{bmatrix} \end{matrix} . \qquad (7)$$

2.4 Multi-station Assembly Sequence Model

In a multi-station assembly line environment, a feasible assembly sequence should contain part assembly sequence information and job assignment information, and there are order constraints between the stations, as shown in Fig. 4.
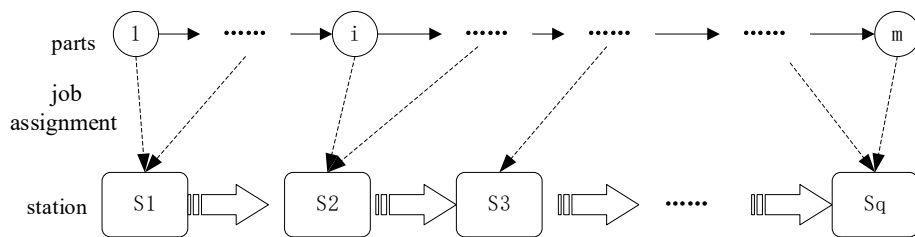

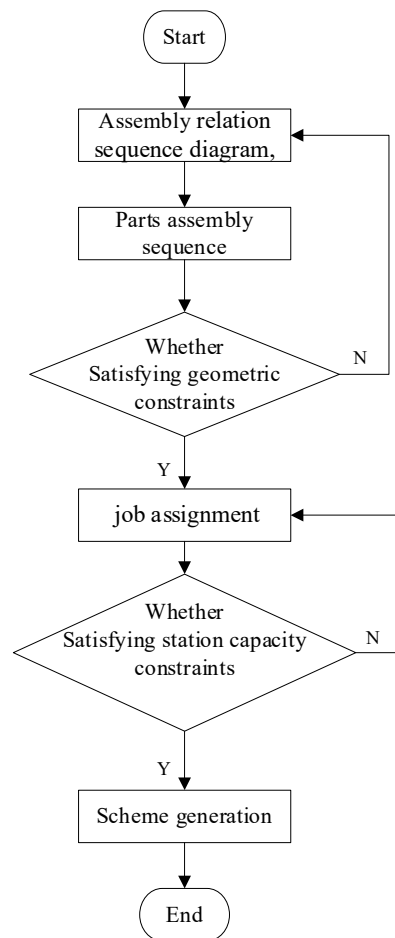**Fig. 4.** Multi-station assembly line assembly sequence

Fig. 5. Generation process of possible assembly sequence

The feasible assembly sequence generation process is shown in Fig. 5. (1) the feasible part assembly sequence is generated based on the part assembly priority relationship; (2) Check the interference of the part; (3) The parts assembly station is randomly assigned according to the order of the stations, and the work station is assigned to check whether the work capacity is satisfied. (4) The multi-station assembly sequence scheme is generated.

Based on the above analysis, the assembly line multi-station assembly sequence model can be represented as the following matrix.

$$
P = \begin{array}{c} \\ v_1 \\ v_2 \\ \dots \\ v_m \end{array}
\begin{array}{cccccc} s_1 & s_2 & \dots & s_q & p \\ \left[\begin{array}{ccccc} c_{11} & c_{12} & \dots & c_{1q} & p_1 \\ c_{21} & c_{22} & \dots & c_{2q} & p_2 \\ \dots & \dots & c_{ik} & \dots & \dots \\ c_{m1} & c_{m2} & \dots & c_{mq} & p_m \end{array} \right] \end{array} . \tag{8}
$$

Where the value of c_{ik} is following formula (4) and should satisfy the restrictions (6). p_i is the assembly order of the parts v_i, $p_i \in [1, m]$ and $p_1 \neq p_2 \neq p_3 \neq \dots \neq p_m$.

3 Particle Swarm Optimization for Multi-station Assembly Sequence

3.1 Particle Coding

In the particle swarm algorithm, each particle flies in an N-dimensional space. Each particle has a fitness value determined by the objective function and a speed that determines their flight direction and distance. Each particle adjusts its next speed and position based on its best position (pbest) and the best position (gbest) found by the group. In the multi-station assembly sequence of assembly line, each particle represents a feasible assembly sequence, so the position of each particle can be represented by equation (8). Similarly, the speed of each particle can be expressed as.

$$
V = \begin{array}{c} \\ v_1 \\ v_2 \\ \dots \\ v_m \end{array}
\begin{array}{cccccc} s_1 & s_2 & \dots & s_q & p \\ \left[\begin{array}{ccccc} cv_{11} & cv_{12} & \dots & cv_{1q} & pv_1 \\ cv_{21} & cv_{22} & \dots & cv_{2q} & pv_2 \\ \dots & \dots & cv_{ik} & \dots & \dots \\ cv_{m1} & cv_{m2} & \dots & cv_{mq} & pv_m \end{array} \right] \end{array} . \tag{9}
$$

Where $cv_{ik} = 0$ or 1, and should satisfy the restrictions (10), $pv_i \in [0, m]$.

$$
s.t \begin{cases} \sum_{k=1}^{q} cv_{ik} = 1 \\ \sum_{i=1}^{m} cv_{ik} \geq 1 \end{cases} . \tag{10}
$$

3.2 Fitness Function

On the assembly line, different assembly sequences and station assignments will result in different assembly operation times for each station, and assembly line balance rates will vary greatly. Therefore, the minimum variance of the assembly time of each station is selected as the fitness function.

$$
f = \sqrt{\frac{\sum_{k=1}^{q}(T_k - \overline{T})^2}{q}} . \tag{11}
$$

Where, T_k is the assembly time occupied by the assembly task on the kth station, which includes assembly time and tool conversion time of part assigned to the station. $\overline{T}$ is the mean value of total assembly time for all stations, and q is the number of stations.

3.3 Algorithm Flow

The particle swarm algorithm first initializes a group of random particles (random solutions), then the particles follow the current optimal particle and search in the solution space. That is, the optimal solution is searched by iteration. The position and velocity of each particle correspond to a matrix, such as equations (8) and (9). Specific steps are as follows:

    (1) Basic parameters setting
    1) Set the number of populations;
    2) Set the number of parts m;
    3) Set the number of stations q;
    4) Set a priority relationship matrix;
    5) Set the work capacity matrix;
    6) Set the interference matrix;
    7) Set the number of iterations D, the number of particles L, the inertia weight, and the acceleration factor.
    (2) Randomly generate particle position and velocity matrix
    1) Randomly generate the particle position and velocity matrix for q+1th column. that randomly generate part assembly order vector $p = (p_1, p_2, ..., p_m)^{\mathrm{T}}$, $pv = (pv_1, pv_2, ..., pv_m)^{\mathrm{T}}$.

    2) According to the feasibility of the part assembly priority relationship matrix test, if possible, go to step 3), otherwise return to step 1).

    3) According to the feasible component assembly order vector p, the station pre-allocation is randomly performed in sequence according to the station order. The allocation principle satisfies the formula (5) to generate a station allocation matrix $\tilde{C}_M$.

    4) Check the feasibility of the station $\tilde{C}_M$. If possible, go to step 5), otherwise return to 3).

    5) Generate position matrix of particle i according to p and $\tilde{C}_M$. Similarly, the velocity matrix V_i of particle i can be generated.

    (3) Calculate the fitness value f for each particle and the p_{best}, g_{best};

    (4) Update the position and velocity of the particle according to formula (12)(13). For specific operators and calculation rules, reference [5] for details.

$$V_i^{t+1} = \omega \otimes V_i^t \oplus (c_1 \times r_1 \otimes (p_{best} \Theta P_i^t)) \oplus (c_2 \times r_2 \otimes (g_{best} \Theta P_i^t)) . \qquad (12)$$

$$P_i^{t+1} = P_i^t \oplus V_i^{t+1} . \qquad (13)$$

Check if the number of iterations meets the termination condition, and return step (2) if it does not satisfy, otherwise go to step (6).

Output g_{best} and decode the optimal position of the particle, get the optimal solution.

4 An Example Analysis

An engineering vehicle model is analyzed to validate the feasibility of the proposed model and algorithm. The engineering vehicle model is mainly composed of 17 parts. The assembly line has 8 stations, and the assembly capacity of each station is shown in formula (14). Firstly, according to the BOM of engineering vehicle, the component assembly priority map is obtained and shown in Fig. 3. Assembly tool matrix is shown as formula (15). Parts 11 and 15 have interference in the x and y directions, and 8, 10 have interference in the x direction, and the

remaining parts have no interference. Thus, in the assembly interference matrix, $a_{d11,15} = 110$, $a_{d8,10} = 100$, others are 000.

$$
C_M = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \\ v_{10} \\ v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \\ v_{15} \\ v_{16} \\ v_{17} \end{array}
\begin{array}{c} s_1\ s_2\ s_3\ s_4\ s_5\ s_6\ s_7\ s_8 \\ \begin{bmatrix}
1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\
1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\
1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\
1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 0
\end{bmatrix} \end{array} \quad (14)
$$

$$
C_M = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \\ v_{10} \\ v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \\ v_{15} \\ v_{16} \\ v_{17} \end{array}
\begin{array}{c} T_1\ T_2\ T_3\ T_4\ T_5 \\ \begin{bmatrix}
1 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 \\
1 & 1 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 \\
1 & 1 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 0
\end{bmatrix} \end{array} \quad (15)
$$

Parameter setting in particle swarm algorithm are as follows: T=100, The number of population is 30, Inertia weight $\omega = 1$, acceleration factor $c_1 = 2$, $c_2 = 2$, The time for replacing the tool is 1seconds. The assembly time of parts is shown in Table 1.

Table 1. Assembly time of parts

Part number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Time (Second)	12	15	25	13	6	5	6	20	22	10	6	12	5	20	12	8	5

The calculation results are as shown in the Fig. 6, the algorithm converges in the 49th generation, and the fitness value is 3.24. The position matrix optimized by the engineering vehicle model assembly sequence is as shown in formula (16).
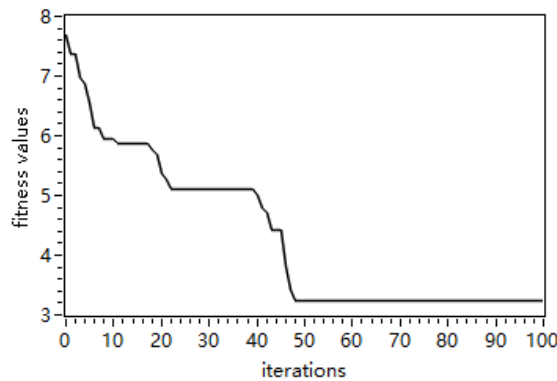

Fig. 6. Calculation result based on PSO

$$P=\begin{array}{c}\\ v_1\\ v_2\\ v_3\\ v_4\\ v_5\\ v_6\\ v_7\\ v_8\\ v_9\\ v_{10}\\ v_{11}\\ v_{12}\\ v_{13}\\ v_{14}\\ v_{15}\\ v_{16}\\ v_{17}\end{array}\begin{array}{ccccccccc}s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & s_7 & s_8 & p\\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2\\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 13\\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 10\\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 5\\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 9\\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1\\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3\\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 8\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 17\\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 4\\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 7\\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 6\\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 11\\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 12\\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 14\\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 15\\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 16\end{array} \qquad (16)$$

The optimization result of assembly sequence for engineering vehicle is 6-1-7-10-4-12-11-8-5-3-13-14-2-15-16-17-9，The station allocation is：part 6, 1, 7 is assigned at S1 station. 10, 4 is assigned at S2 station. 12, 11 is assigned at S3 station. 8, 5 is assigned at S4 station. 3, 13 is assigned at S5 station. 14, 2 is assigned at S6 station. 15, 16, 17 is assigned at S7 station. 9 is assigned at S8 station.
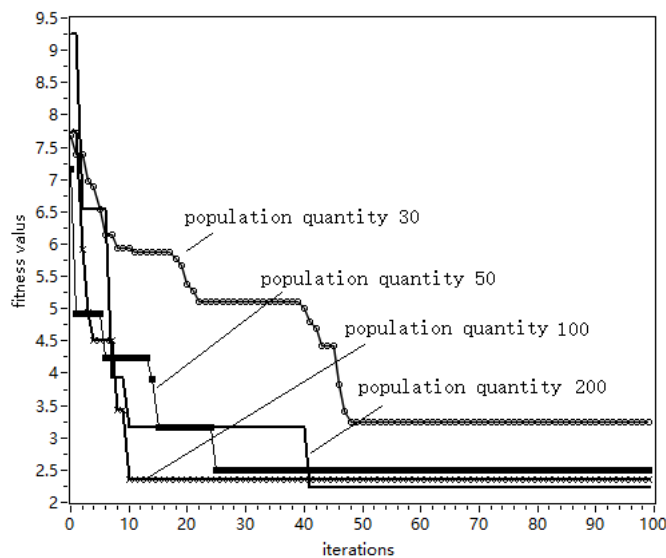

Fig. 7. Algorithm results for different population numbers

Fig. 7 shows the different optimization result for population numbers of 50, 100, 200. It can be seen that when the population is small, fewer feasible sequences are produced, it results a slow convergence speed for PSO algorithm. And the optimal solution obtained is caught in the local optimal solution earlier.

Table 2. Optimization results of multi-station assembly sequences under different population sizes

Population size		50	100	150	200
Optimal assembly sequence		6,7,4,1,11,15,10,12, 13,5,16,2,8,3,14,17,9	6,7,1,2,4,10,3,12, 17,13,8,9,16,15,5,14,11	6,1,2,7,4,15,3,16, 17,10,12,11,13,8,14,5,9	6,1,7,2,4,15,3,17, 16,10,11,12,13,8,5,14,9
Station allocation result	S_1	6 7 4	6 7 2 1	6 1 7 2	6,1,7,2,4
	S_2	1 11 15	4 10	4 15	4,15
	S_3	10 12	3	3	3
	S_4	13 5 2 16	12 17 13	16 17 10	17,16,10
	S_5	8	8	12 11 13	11,12,13
	S_6	3	9	8	8
	S_7	14 17	16 15 5	14 5	5,14
	S_8	9	14 11	9	9
Optimal fitness function value		2. 784	2.5	2.345	2.345

To further verify the validity of the algorithm, the population capacity is changed to 50, 100, 150, 200 and with the basic setting parameters unchanged, the obtained optimal result is decoded as shown in Table 2. It shows that in the particle swarm optimization algorithm for solving multi-station assembly optimization sequences, due to the limitation of station capacity and station allocation order, smaller populations get more infeasible solutions, which it is difficult to get a global optimal solution. As the population increases, the optimization results gradually approach the global optimal solution. In this example experiment, the global optimal solution is obtained when the population is 150.

5 Conclusions

(1) For the problem of station order constraints and station capacity constraints in a multi-station environment, the assembly line multi-station assembly sequence model is constructed, and the constraints that the feasible assembly sequence should meet are given.
(2) The minimum variance of the assembly time of each station that characterizes the assembly line balance rate is used as the fitness function, based on this, a multi-station ASP method based on particle swarm optimization is proposed.
(3) The flow of implementing the algorithm is designed, and the assembly of a vehicle model is taken as an example. Using the multi-station assembly sequence model and the designed particle swarm optimization algorithm, the optimal assembly line multi-station assembly sequence and station distribution integration solution are obtained, which the experimental result has verified the effectiveness of the proposed method.

In this paper, the multi-station assembly line model is suitable for serial assembly. The parallel assembly is not considered when building the model. Parallel assembly can further shorten the product assembly time and improve the assembly efficiency. Therefore, the problem of parallel ASP can be further studied in the future. In addition, the number of stations studied in this paper is fixed. In the actual assembly line, the number of available stations changes due to the failure of tooling and assembly manipulator. At this time, ASP should make timely adjustment, which is also the direction of future research.

6 Acknowledgments

References

[1] Y.-Y. Hsu, P.-H. Tai, M.-W. Wang, W.-C. Chen, A knowledge-based engineering system for assembly sequence planning, The International Journal of Advanced Manufacturing Technology 55(5-8)(2011) 763-782.

[2] X.-M. Zhou, P.-A. Du. Assembly modeling for assembly sequence planning, Computer Integrated Manufacturing System 13(12)(2007) 2365-2372.

[3] A.C. Lin, T.-C. Chang, 3D MAPS: Three-dimensional mechanical assembly planning system, Journal of Manufacturing Systems 12(6)(1993) 437-456.

[4] F.-C. Wang, Y.-C. Sun, N. Li, Multi station assembly sequence planning based on particle swarm optimization algorithm, Journal of Mechanical Engineering 48(09)(2012) 1-8.

[5] J.-W. Liu, Y. Guo, S.-S. Zha, F.-L. Wang, S.-C. Zhang, Multi station assembly sequence planning based on improved particle swarm optimization algorithm, Computer Integrated Manufacturing System 24(11)(2018) 2701-2711.

[6] Y.-P. Ren, C.-Y. Zhang, F. Zhao, H.-J. Xiao, G.-D. Tian, An asynchronous parallel disassembly planning based on genetic algorithm, European Journal of Operational Research 269(2)(2018) 647-660.

[7] L. Laperrière, H.A. EIMaraghy, GAPP: A generative assembly process planner, Journal of Manufacturing Systems 15(4) (1996) 282-293.

[8] L.S. Homem de Mello, A.C. Sanderson, A correct and complete algorithm for the generation of mechanical assembly sequences, IEEE Transactions on Robotics and Automation 7(2)(1991) 228-240.

[9] Y. Wang, J.-H. Liu, L.-S. Li, Assembly sequences merging based on assembly unit partitioning, The International Journal of Advanced Manufacturing Technology 45(7)(2009): 808-820.

[10] R. Li, Y.-L. Fu, H.-B. Feng, Assembly sequence planning based on connected structure knowledge, Computer Integrated Manufacturing System 14(6)(2008) 1130-1135.

[11] A. Gunji, Deepak, B. B. B. V. L, C. Bahubalendruni, D. Biswal. An optimal robotic assembly sequence planning by assembly subsets detection method using teaching learning-based optimization algorithm, IEEE Transactions on Automation Science and Engineering 15(3)(2018) 1369-1385.

[12] W. Zhou, J.-R Zheng, J.-J Yan, J.-F. Wang, A novel hybrid algorithm for assembly sequence planning combining bacterial chemotaxis with genetic algorithm, The International Journal of Advanced Manufacturing Technology 52(5)(2011) 715-724.

[13] S.-C. Shi, R. Li, Y.-L. Fu, Y.-L, Ma. Assembly sequence planning based on improved ant colony algorithm, Computer Integrated Manufacturing System 16(6)(2010) 1189-1194.

[14] H.-Y Zhang, H.-J. Liu, L.-Y. Li, Research on a kind of assembly sequence planning based on immune algorithm and particle swarm optimization algorithm, The International Journal of Advanced Manufacturing Technology 71(5-8)(2014) 795-808.

[15] M.F.F.Ab Rashid, W. Hutabarat, A. Tiwari, Multi-objective discrete particle swarm optimization algorithm for integrated assembly sequence planning and assembly line balancing, Proceedings of the Institution of Mechanical Engineers Part B-Journal of Engineering Manufacture 232(8)(2018) 1444-1459.

[16] B. Zeng, M.-F. Li, Y. Zhang, J.-H. Ma, Research on assembly sequence planning based on firefly algorithm, Journal of Mechanical Engineering 49(11)(2013) 177-184.

[17] X.-Y Li, K. Qin, B. Zeng, L. Gao, L.-J Wang, A dynamic parameter controlled harmony search algorithm for assembly sequence planning, The International Journal of Advanced Manufacturing Technology 82(9-12)(2017) 3399-3411.

[18] O. Ozmen, T. Batbat, T. Ozen, C. Sinanoglu, A. Guven, Optimum assembly sequence planning system using discrete artificial bee colony algorithm, Mathematical Problems in Engineering (2018) 3407646.

[19] C. Lu, J.-Y. Li, Assembly sequence planning considering the effect of assembly resources with a discrete fireworks algorithm, International Journal of Advanced Manufacturing Technology 93(9-12)(2017) 3297-3314.

[20] X.-Y. Li, K. Qin, B. Zeng, L. Gao, J.-Z. Sun, Assembly sequence planning based on an improved harmony search algorithm, International Journal of Advanced Manufacturing Technology 84(9-12)(2016) 2367-2380.

[21] F.M. Defersha, F. Mohebalizadehgashti, Simultaneous balancing, sequencing, and workstation planning for a mixed model manual assembly line using hybrid genetic algorithm, Computers & Industrial Engineering 119(2018) 370-387.

[22] G. Pedraza, M. Diaz, H. Lombera, An approach for assembly sequence planning by genetic algorithms, IEEE Latin America Transactions 14(5)(2016) 2066-2071.

[23] M. Sasiadek, Planning and analysis of mechanical assembly sequences in design engineering - part II: application, Tehnicki Vjesnik-Technical Gazette 22(3)(2015) 643-648.

[24] W.-B. Yuan, L. Chang, Z.-B. Xu, T.-L. Gu, Multi-plant assembly sequence planning based on fruit fly optimization algorithm, Computer Science 44(4)(2017) 246-251.