Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Supplementary materials for

Huanpei LYU, Libin ZHANG, Dapeng TAN, Fang XU, 2023. A collaborative assembly for low-voltage electrical apparatuses. *Front Inform Technol Electron Eng*, 24(6):890-905. https://doi.org/10.1631/FITEE.2100423

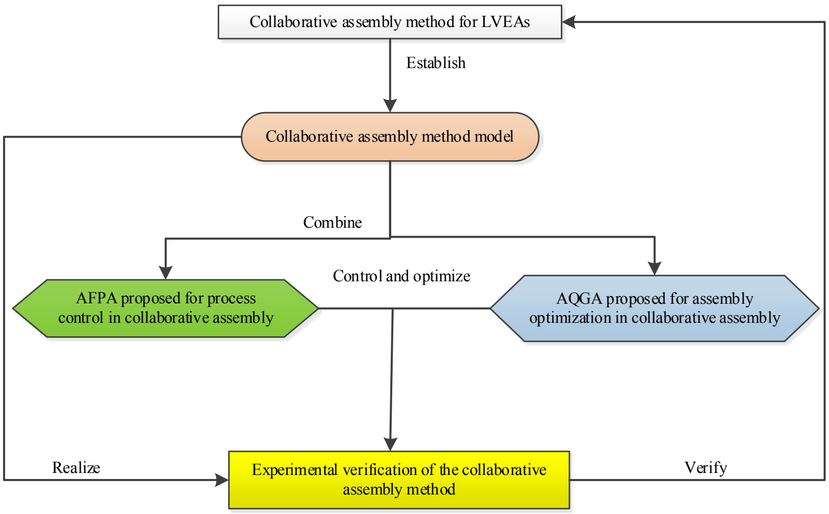

Collaborative assembly method for LVEAs
Establish
Collaborative assembly method model
Combine
Control and optimize
AFPA proposed for process control in collaborative assembly
AQGA proposed for assembly optimization in collaborative assembly
Realize
Experimental verification of the collaborative assembly method
Verify

Fig. S1 Main content of this paper

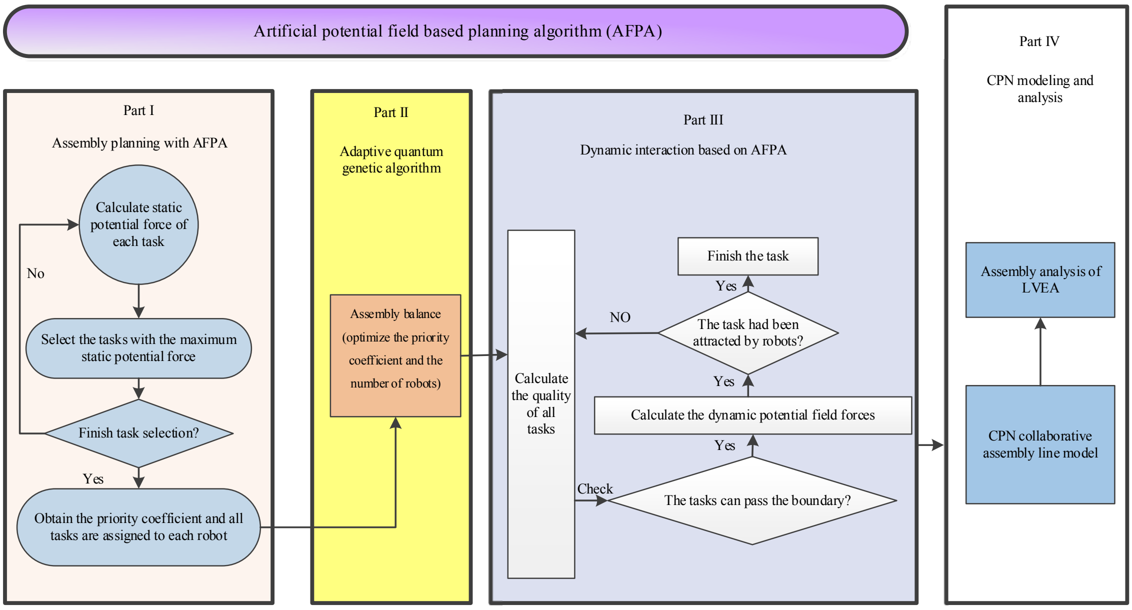

Artificial potential field based planning algorithm (AFPA)
Part I
Assembly planning with AFPA
Calculate static potential force of each task
No
Select the tasks with the maximum static potential force
Finish task selection?
Yes
Obtain the priority coefficient and all tasks are assigned to each robot
Part II
Adaptive quantum genetic algorithm
Assembly balance (optimize the priority coefficient and the number of robots)
Part III
Dynamic interaction based on AFPA
Calculate the quality of all tasks
Finish the task
Yes
NO
The task had been attracted by robots?
Yes
Calculate the dynamic potential field forces
Yes
Check
The tasks can pass the boundary?
Part IV
CPN modeling and analysis
Assembly analysis of LVEA
CPN collaborative assembly line model

Fig. S2 Architecture of the collaborative assembly methodology

If $\boldsymbol{X}(x, y)$ is the position of the robots, the attracted potential between a robot and task $\boldsymbol{U}_{\text{att}}$ is calculated as follows:

$$\boldsymbol{U}_{\text{att}} = \frac{1}{2} k_{\text{att}} (\boldsymbol{X} - \boldsymbol{X}_{\text{g}})^2, \tag{S1}$$

where k_{att} is an attracted potential scaling factor and $\boldsymbol{X}_{\text{g}}$ is the position of the goal. Thus, the attracted force $\boldsymbol{F}_{\text{att}}$ can be defined as follows:

$$\boldsymbol{F}_{\text{att}} = -\nabla \boldsymbol{U}_{\text{att}}(\boldsymbol{X}) = -k_{\text{att}} \left\| \boldsymbol{X} - \boldsymbol{X}_{\text{g}} \right\|. \tag{S2}$$

The definition of the repelled potential between a robot and an obstacle is calculated as follows:

$$\boldsymbol{U}_{\text{rep}} = \begin{cases} \frac{1}{2} k_{\text{rep}} \left(\frac{1}{\left\| \boldsymbol{X} - \boldsymbol{X}_{\text{o}} \right\|} - \frac{1}{\rho_{\text{o}}} \right)^2, & \left\| \boldsymbol{X} - \boldsymbol{X}_{\text{o}} \right\| \geq \rho, \\ 0, & \left\| \boldsymbol{X} - \boldsymbol{X}_{\text{o}} \right\| \geq \rho_{\text{o}}, \end{cases} \tag{S3}$$

where k_{rep} is the repelled potential scaling factor, $\boldsymbol{X}_{\text{o}}$ is the position of the obstacle, and ρ_{o} is the interference distance of the obstacle. Hence, the repelled force $\boldsymbol{F}_{\text{rep}}$ can be defined as follows:

$$\boldsymbol{F}_{\text{rep}} = -\nabla \boldsymbol{U}_{\text{rep}}(\boldsymbol{X}) = \begin{cases} \frac{1}{2} k_{\text{rep}} \left(\frac{1}{\left\| \boldsymbol{X} - \boldsymbol{X}_{\text{o}} \right\|} - \frac{1}{\rho_{\text{o}}} \right) \frac{1}{\left\| \boldsymbol{X} - \boldsymbol{X}_{\text{o}} \right\|^2} \frac{\partial(\left\| \boldsymbol{X} - \boldsymbol{X}_{\text{o}} \right\|)}{\partial \boldsymbol{X}}, & \left\| \boldsymbol{X} - \boldsymbol{X}_{\text{o}} \right\| \leq \rho_{\text{o}}, \\ 0, & \left\| \boldsymbol{X} - \boldsymbol{X}_{\text{o}} \right\| \geq \rho_{\text{o}}. \end{cases} \tag{S4}$$

Therefore, the global potential of robot $\boldsymbol{U}_{\text{glob}}$ is calculated as follows:

$$\boldsymbol{U}_{\text{glob}} = \boldsymbol{U}_{\text{att}} + \boldsymbol{U}_{\text{rep}}. \tag{S5}$$

The total force $\boldsymbol{F}_{\text{total}}$ acting on a robot is calculated as follows:

$$\boldsymbol{F}_{\text{total}} = \boldsymbol{F}_{\text{att}} + \boldsymbol{F}_{\text{rep}}. \tag{S6}$$

Fig. S3 Definition of the artificial potential field (APF)

CPN is a multivariate group, where CPN=(Σ, P, T, A, N, C, G, E, IN) satisfying the following conditions:

(1) Σ is a type of non-empty finite set called a color set;

(2) P is a finite set of places;

(3) T is a finite set of transitions;

(4) A is a finite set of arcs, where $P \cap T = P \cap A = T \cap A = \varnothing$ and $\varnothing$ is an empty set;

(5) N is a node function, where N: $A \rightarrow (P \times T) \cup (T \times P)$;

(6) C is a color function, where C: $P \rightarrow \Sigma$;

(7) G is a guard function, defined from T into expressions such that $\forall t \in T$, Type($G(t)$)=Bool$\wedge$Type (Var($G(t)$))$\subseteq \Sigma$, and has its own threshold, which is used to judge whether the input value is satisfied;

(8) E is an arc expression function defined from E into expressions such that $\forall a \in A$, Type($E(a)$)= $C(p(a))_{\text{MS}}$$\wedge$Type(Var($E(a)$))$\subseteq \Sigma$, where $p(a)$ is the place of $N(a)$;

(9) IN is an initial function defined from P into expressions such that $\forall p \in P$, Type(IN(p))=$C(p)_{\text{MS}}$$\wedge$ Var(IN(p))=$\varnothing$.

Fig. S4 Formal definition of the colored Petri net (CPN)

Table S3 (which will be given in the following) shows the setting of places in the model. According to the theoretical model of the assembly, Ptask: {task|task n=(s_n, t_n, b_n, sq$_n$, add$_n$, f_n), $s_n \in$Ptname, $t_n \in$Ptnum, sq$_n \in$Ptchng, add$_n \in$Ptchng, $f_n \in$Ptslect, $n \in \mathbb{N}$}. Probot: {robot|robot n=(t_n, b_n, sq$_n$, add$_n$, f_n), $t_n \in$Ptkind, $b_n \in$Ptnum, sq$_n \in$Ptchng, add$_n \in$Ptchng, $f_n \in$Ptslect, $n \in \mathbb{N}$}. Psignal: {tsignal|tsignal n=(su$_n$ and f_n), add$_n \in$Ptchng,$f_n \in$Ptslect, su$_n \in$Ptchng, $n \in \mathbb{N}$}. Thus, in software CPN Tools, some places can be defined as follows:

Assembly task place colset:
Ptask=product Ptname*Ptkind*Ptnum*Ptsequ*Ptadd*Ptslect;
Task caching place colset:
Tbuffer=product Ptname*Ptkind*Ptnum*Ptsequ*Ptadd;
Completed task place colset:
Tskfinish=product Ptname*Ptkind*Ptslect;
Assembly robot place colset:
Probot=product Ptkind*Ptnum*Ptsequ*Ptadd*Ptslect;
Task information place colset:
Psignal=product Ptkind*Ptnum*Ptsequ*Ptadd*Ptslect.

Fig. S5 Setting of places in the CPN collaborative assembly model

1. Transitions between the task place and the task buffer place

The variable ts=<s, b, t, su, add, f>, and there is a guard function G(ts):

$$G(\mathrm{ts}) = \begin{cases} 1, & f = \mathrm{Prio}, \\ 0, & \text{otherwise}, \end{cases} \tag{S7}$$

where Prio is the assembly priority coefficient, Prio∈{1, 2, 3, 4, 5, 6}. That is, when f is equal to the corresponding Prio, the task proceeds, and Prio is applied as a threshold in function G(ts). In Fig. 2, IN1, IN2, IN3, IN4, IN5, and IN6 belong to the transitions between the task place and the task buffer place.

2. Transitions between the task buffer place and the assembly robot place

The variable tr_n=<b_n, t_n, sq_n, add_n, f_n>, and there is a guard function $G(\mathrm{tr}_n)$:

$$G(\mathrm{tr}_n) = \begin{cases} 1, & b_n = \mathrm{rk},\ t_n = \mathrm{sq}_{\mathrm{next}}, \\ 0, & \quad \text{otherwise}, \end{cases} \tag{S8}$$

where rk is the robot-type coefficient, rk∈{1, 2, 3}, and n is the serial number of the tasks, n∈{1, 2, …, }. When the value of b_n is equal to rk and the signal of t_n=$\mathrm{sq}_{\mathrm{next}}$ is obtained, the task is accepted by the robot. In the function $G(\mathrm{tr}_n)$, rk and $\mathrm{sq}_{\mathrm{next}}$ are used as two thresholds. In Fig. 2, Asm_Rp012, Asm_Rs011, Asm_Rs012, Asm_Rp011, and Asm_Rw011 belong to the transitions between the task buffer place and the assembly robot place.

3. Transitions between the completed task place colset and the task or product name place

The variable tf_n=<s_n, b_n, f_n>, and there is a guard function $G(\mathrm{tf}_n)$:

$$G(\mathrm{tf}_n) = \begin{cases} 1, & \mathrm{tf}_n = \mathrm{rk},\ t_n = \mathrm{sq}_{\mathrm{next}}, \\ 0, & \quad \text{otherwise}, \end{cases} \tag{S9}$$

where rk and $\mathrm{sq}_{\mathrm{next}}$ are the thresholds of $G(\mathrm{tf}_n)$.

In Fig. 2, T_Rp, T_Rs, T_Rw, Check1, Check2, and Check3 belong to the transitions between the completed task place colset and the task or product name place.

4. Transitions between the input information place and the output information place

The variable Inform_n=<sq_n, add_n, f_n>, and there is a guard function

$$G(\mathrm{Inform}_n) = \begin{cases} 1, & \mathrm{sq}_n = \mathrm{add}_n, \\ 0, & \text{otherwise}. \end{cases} \tag{S10}$$

In $G(\mathrm{Inform}_n)$, the threshold is the input value Inform_n(<sq_n, add_n, f_n>), which is used for comparison with the next input value Inform_{n+1}, when $\mathrm{sq}_n = \mathrm{sq}_{n+1}$, $\mathrm{add}_n \neq \mathrm{add}_{n+1}$, $\mathrm{sq}_n \neq \mathrm{add}_n$, and $\mathrm{sq}_n \neq \mathrm{add}_n$, the threshold is equal to <sq_n, add_n+add_{n+1}, f_n>.

In Fig. 2, ADD1, ADD2, Send_signal, Selc, and Initial belong to the transitions between the input information place and the output information place.

Accordingly, the corresponding assembly unit is designed in CPN Tools by combining the places and transitions mentioned above, as shown in Fig. 2.

Fig. S6 Setting of model transitions in the CPN collaborative assembly model

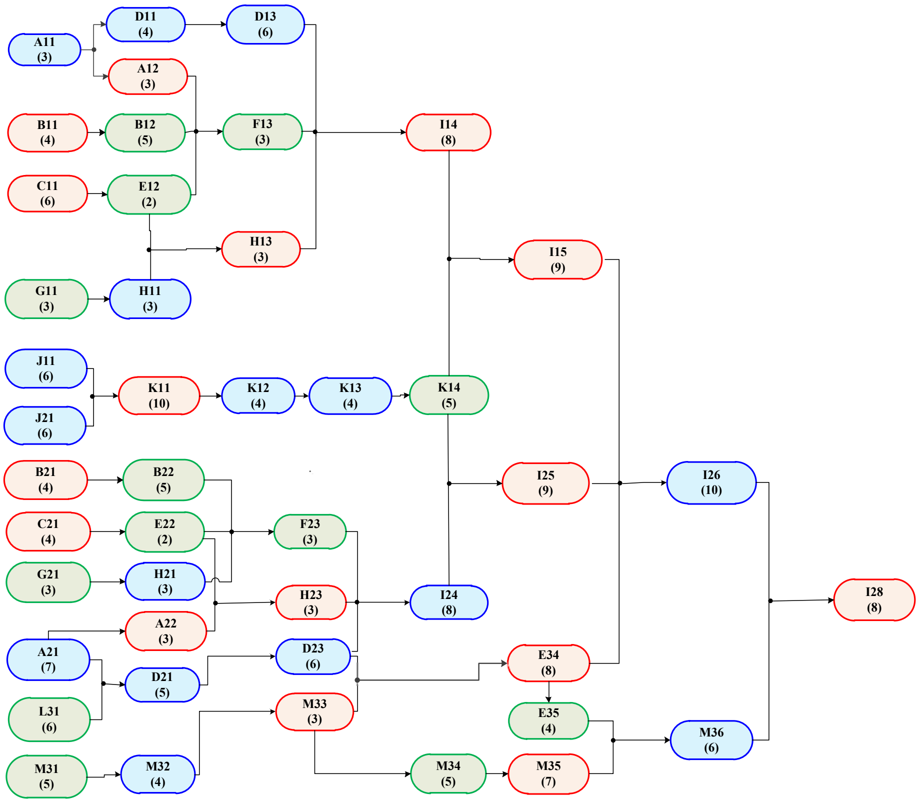
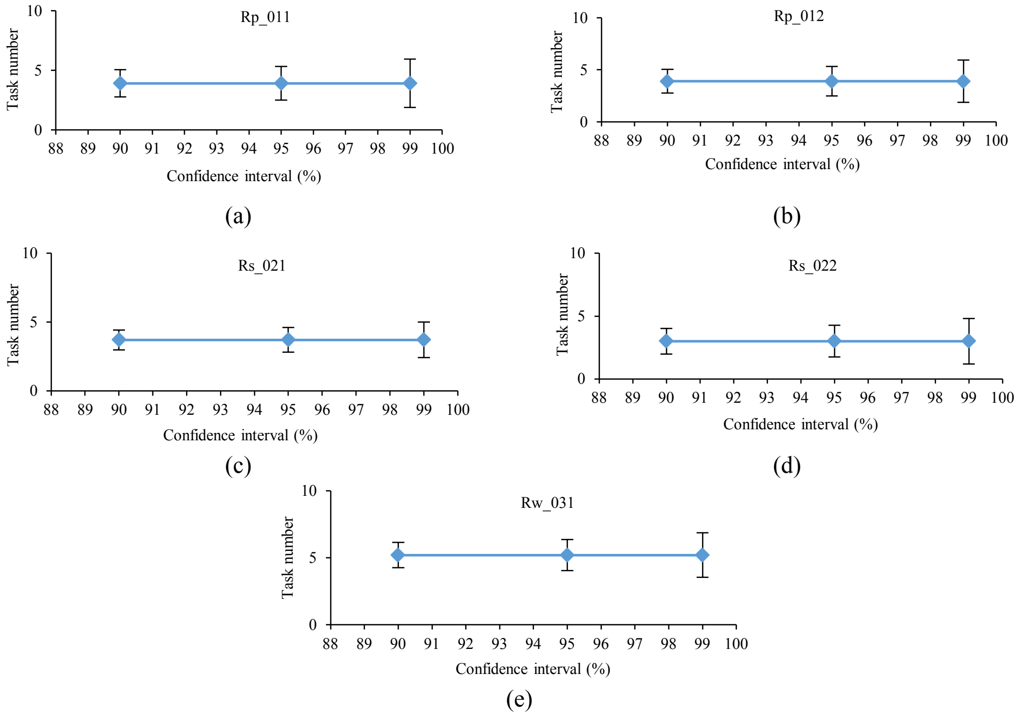


Fig. S7 Relationship between the tasks required for a TPCLP



(a)

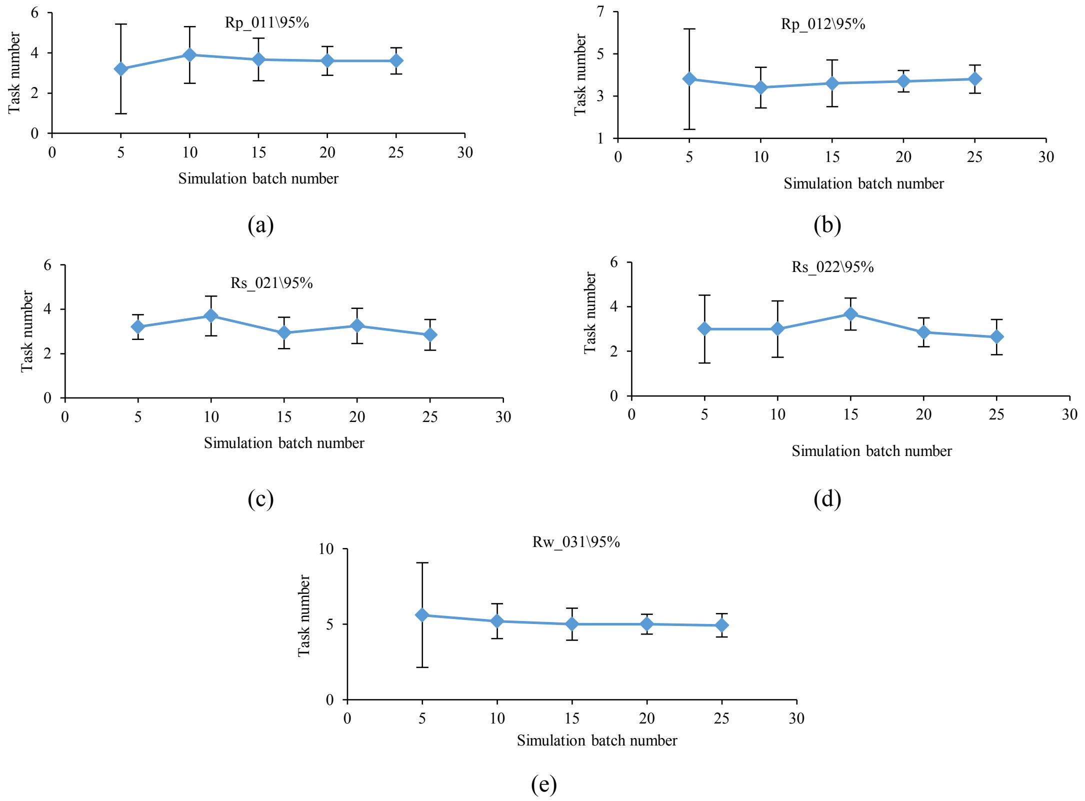
(b)

(c)

(d)

(e)

Fig. S8 Average number of assembly tasks of the assembly robots for different confidence intervals: (a) Rp_011 robot; (b) Rp_012 robot; (c) Rs_021 robot; (d) Rs_022 robot; (e) Rw_031 robot



(a)

(b)

(c)

(d)

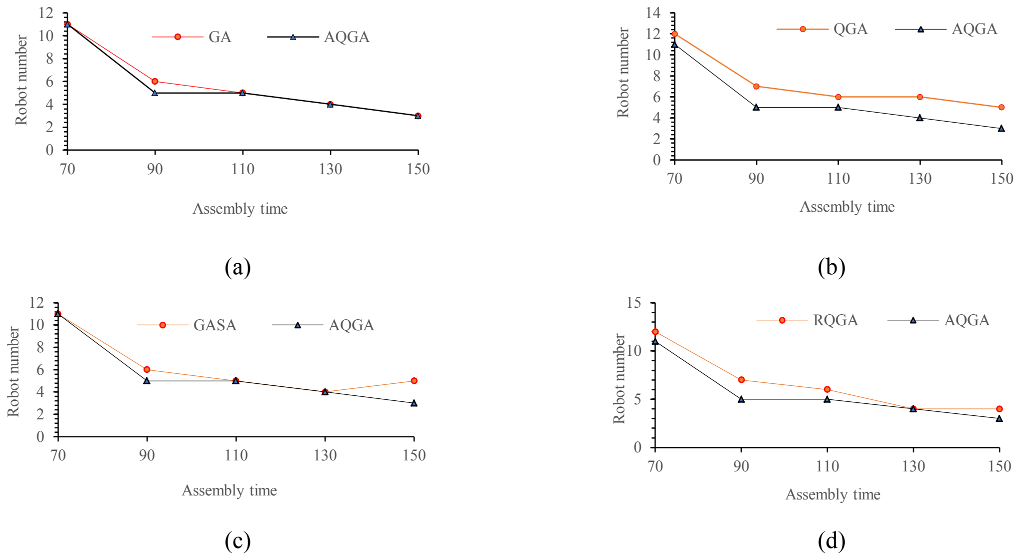
(e)

Fig. S9 Number of assembly tasks for each assembly robot for different numbers of simulation batches: (a) Rp_011 robot; (b) Rp_012 robot; (c) Rs_021 robot; (d) Rs_022 robot; (e) Rw_031 robot

GA and GASA: The population size is 40, the maximum genetic generation is 20, the individual variable dimension is 45, the crossover probability is 0.75, and the mutation rate is 0.04.

QGA, RQGA, and AQGA: The population size is 40, the maximum genetic generation is 20, the individual variable dimension is 45, and the quantum rotation angle is 0.05.

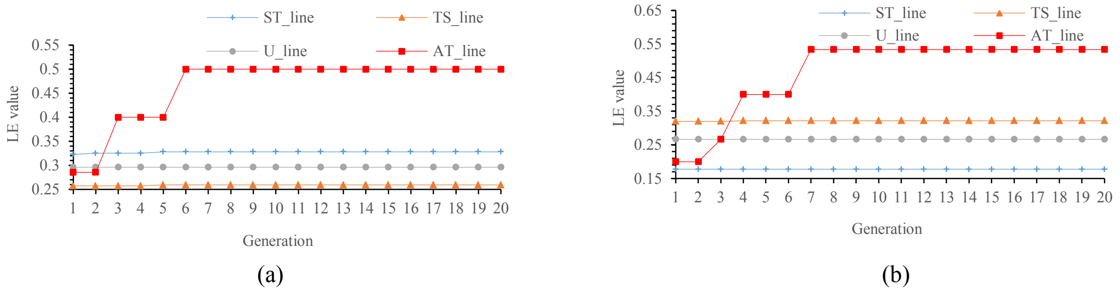
Fig. S10 Setting of relevant parameters of the compared algorithms



(a) (b)

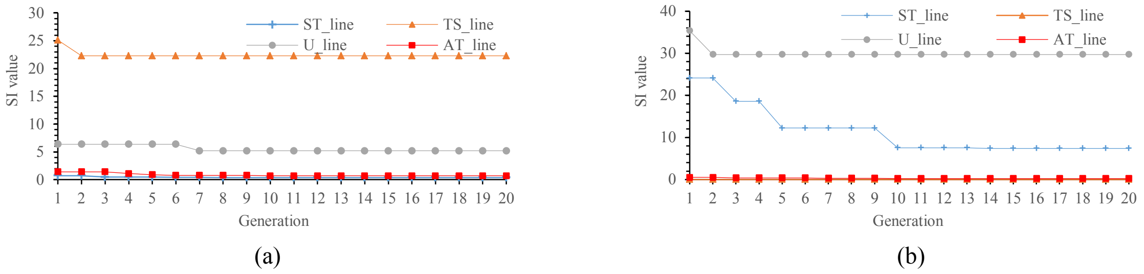
(c) (d)

Fig. S11 Comparison of the number of robots required for the different algorithms: (a) GA vs. AQGA; (b) QGA vs. AQGA; (c) GASA vs. AQGA; (d) RQGA vs. AQGA



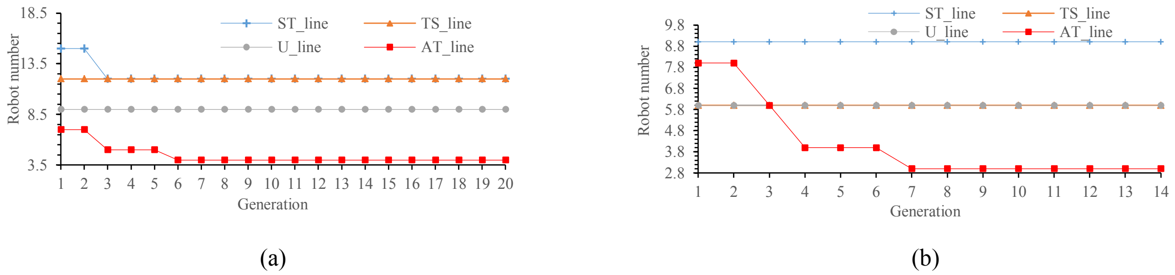
(a) (b)

Fig. S12 LE values of each assembly mode for different assembly cycles: (a) 90 time units; (b) 150 time units



(a) (b)

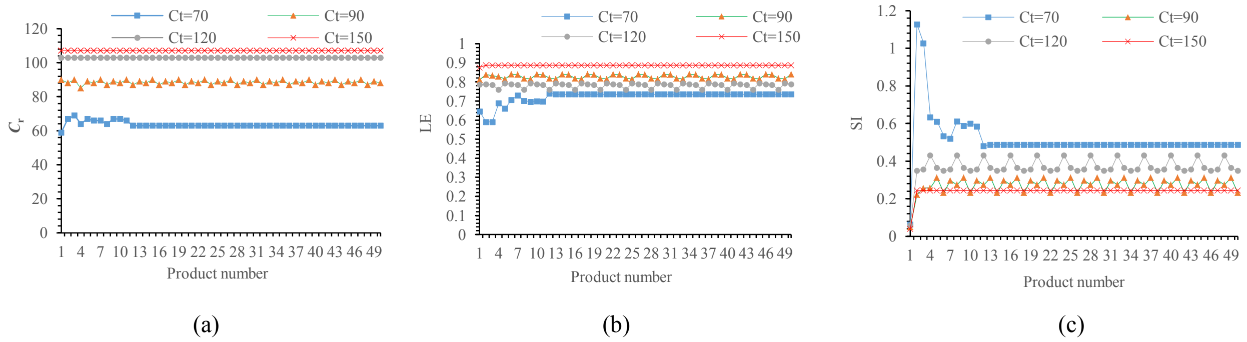
Fig. S13 SI values of each assembly mode for different assembly cycles: (a) 90 time units; (b) 150 time units



(a)

(b)

Fig. S14 Number of robots required for each assembly mode for different assembly cycles: (a) 90 time units; (b) 150 time units



(a)

(b)

(c)

Fig. S15 Analysis of the robot assembly performance with dynamic balance of AFPA: (a) actual assembly cycle C_r; (b) LE value; (c) SI value

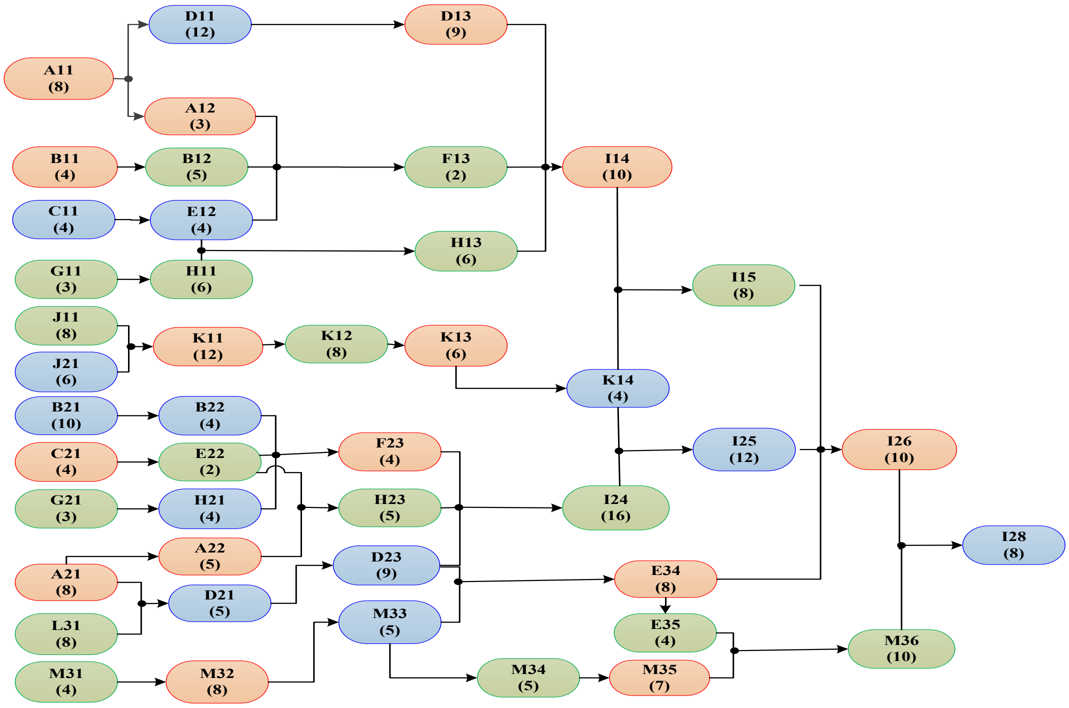


Fig. S16 Relationship between the assembly tasks required for type 2

Table S1 Studies of assembly line construction and balance optimization

Reference	Line type	Solution	Assembly problem
Yu and Yang, 2018	Flexible assembly	Dynamic management of a flexible assembly process based on actual information	Agility requirements of multi-variety and small-batch production
Çil et al., 2017	Robotic parallel assembly line	Iterative beam search (IBS), best search method based on IBS (BIBS), and cutting BIBS (CBIBS) algorithms	Minimum joint cycle time
Tavakoli, 2020	Multi-product assembly line	Tabu search and simulated annealing algorithms	Multi-criterion optimization of multi-product assembly line
Yan and Jiao, 2011	Parallel assembly line	An improved particle swarm optimization - scatter search (PSO-SS) co-evolution algorithm	Multi-robot parallel assembly optimization
Li et al., 2021	Flexible assembly	Hybrid POS algorithm	Scheduling problem of flexible assembly systems (FASs) without intermediate buffers
Samouei1 et al., 2016	Fuzzy mixed-model assembly line	Hierarchical-cyclic heuristic approach	Fuzzy mixed-model assembly line balancing problem
Özcan and Toklu, 2009	Simple straight and U-type assembly line	Adaptive learning approach and simulated annealing	Simple straight and U-type assembly line balancing problems
Rizwan et al., 2020	Human–robot collaborative assembly	Hybrid conditional planning, answer set programming	Human–robot collaborative assembly planning
Johannsmeier and Haddadin, 2017	Human–robot collaborative assembly	A hierarchical human–robot interaction-planning framework	Assembly task allocation
He et al., 2021	Robot assembly	Improved deep deterministic policy gradient reinforcement learning algorithm	Axle-hole assembly

Table S2 Assembly tasks corresponding to each assembly robot

Robot type	Corresponding assembly tasks
R_p	A11, J11, J21, A21, D11, H11, H21, D21, M32, D13, K12, D23, I14, K13, I24, I15, M36, I26
R_s	A12, B11, C21, C11, B21, K11, H13, H23, M33, E34, M35, I25, I28, A22
R_w	G11, A11, G21, L31, M31, B12, E12, E22, F13, F23, M34, K14, E35

Table S3 Relationship between the assembly cycle and the robot number

Assembly cycle (time unit)	Robot number
70	11
90	5
110	5
130	4
150	3

Table S4 Number of assembly tasks for each type of robot

System status	Monitoring point	Status	Number of tasks completed	Number of products completed
All robots are functioning normally	Asm_Rp011	Normal	166	10
	Asm_Rp012	Normal	164	
	Asm_Rs011	Normal	98	
	Asm_Rs012	Normal	82	
	Asm_Rw011	Normal	210	
	T_Rp	Normal	330	
	T_Rs	Normal	180	
	T_Rw	Normal	210	
One robot fails	Asm_Rp011	Normal	330	10
	Asm_Rp012	Trouble	0	
	Asm_Rs011	Normal	90	
	Asm_Rs012	Normal	90	
	Asm_Rw011	Normal	210	
	T_Rp	Normal	330	
	T_Rs	Normal	180	
	T_Rw	Normal	210	
Two robots fail	Asm_Rp011	Normal	330	10
	Asm_Rp012	Trouble	0	
	Asm_Rs011	Normal	180	
	Asm_Rs012	Trouble	0	
	Asm_nh011	Normal	210	
	T_Rp	Normal	330	
	T_Rs	Normal	180	
	T_Rw	Normal	210	
Three robots fail	Asm_ Rp011	Normal	160	0
	Asm_Rp012	Trouble	0	
	Asm_Rs011	Normal	100	
	Asm_Rs012	Trouble	0	
	Asm_Rw011	Trouble	210	
	T_Rp	Normal	160	
	T_Rs	Normal	100	
	T_Rw	Trouble	0	

Table S5 Value range of the critical parameters

Parameter	Population size	Genetic generation	Crossover probability	Mutation rate
Value range	20–100	20–500	0.4–0.99	0.0001–0.1

Table S6 Comparison of optimal target values for each algorithm

C_t	Algorithm	Min. SI	Max. LE	Min. Fitness	Min. *S*
70	QGA	0.6627	0.0790	1.133	12
	GA	0.6760	0.0857	1.132	11
	GASA	0.6830	0.0857	1.123	11
	RQGA	0.7780	0.2857	1.132	12
	AQGA	0.6830	0.3117	1.130	11
90	QGA	0.6210	0.1290	1.115	7
	GA	0.5490	0.1670	1.113	6
	GASA	0.6090	0.1620	1.113	6
	RQGA	0.6020	0.3810	1.112	7
	AQGA	0.4930	0.5330	1.108	5
110	QGA	0.5064	0.1379	1.113	6
	GA	0.4311	0.1745	1.111	5
	GASA	0.4595	0.1745	1.112	5
	RQGA	0.6005	0.3636	1.110	6
	AQGA	0.4364	0.4928	1.109	5
130	QGA	0.4152	0.1167	1.113	6
	GA	0.3784	0.2212	1.110	4
	GASA	0.3832	0.2308	1.110	4
	RQGA	0.3678	0.4615	1.107	4
	AQGA	0.3253	0.4615	1.107	4
150	QGA	0.3898	0.1773	1.110	5
	GA	0.2798	0.3200	1.108	3
	GASA	0.4253	0.1533	1.111	5
	RQGA	0.4000	0.2000	1.106	4
	AQGA	0.2411	0.5333	1.106	3

Table S7 Statistical data of different algorithms

No.	RBN					SI				
	AQGA	GA	GASA	QGA	RQGA	AQGA	GA	GASA	QGA	RQGA
1	5	6	7	6	6	0.608	0.608	0.764	0.594	0.609
2	6	7	6	6	6	0.539	0.735	0.609	0.508	0.518
3	5	5	6	7	6	0.463	0.505	0.609	0.583	0.530
4	5	6	6	6	6	0.505	0.602	0.609	0.563	0.529
5	5	5	6	6	7	0.492	0.431	0.609	0.609	0.431
6	5	7	6	7	7	0.478	0.583	0.609	0.596	0.583
7	5	5	7	6	6	0.500	0.431	0.735	0.492	0.508
8	5	6	5	6	6	0.414	0.598	0.434	0.492	0.609
9	5	6	7	7	6	0.508	0.608	0.763	0.554	0.508
10	5	6	7	6	6	0.459	0.603	0.735	0.589	0.463

No.	LE					Fitness				
	AQGA	GA	GASA	QGA	RQGA	AQGA	GA	GASA	QGA	RQGA
1	0.444	0.161	0.138	0.161	0.444	1.110	1.113	1.116	1.113	1.110
2	0.444	0.124	0.161	0.161	0.444	1.110	1.115	1.111	1.113	1.110
3	0.533	0.198	0.161	0.138	0.444	1.108	1.112	1.113	1.115	1.110
4	0.533	0.167	0.161	0.161	0.444	1.108	1.113	1.113	1.113	1.110
5	0.533	0.200	0.161	0.161	0.381	1.108	1.111	1.113	1.113	1.113
6	0.533	0.124	0.161	0.129	0.381	1.108	1.115	1.113	1.115	1.113
7	0.533	0.200	0.124	0.161	0.444	1.108	1.111	1.115	1.113	1.110
8	0.533	0.161	0.200	0.161	0.444	1.108	1.113	1.111	1.113	1.110
9	0.533	0.161	0.138	0.138	0.444	1.108	1.113	1.116	1.115	1.111
10	0.533	0.167	0.124	0.161	0.444	1.108	1.113	1.115	1.113	1.110

Table S8 Descriptive statistics of different algorithms

Variable	Algorithm	Mean	Std.	Min.	Max.	P25	P50	P75
RBN	AQGA	5.1	0.316 228	5	6	5	5	5.00
	GA	5.9	0.737 865	5	7	5	6	6.25
	GASA	6.3	0.674 949	5	7	6	6	7.00
	QGA	6.3	0.483 046	6	7	6	6	7.00
	RQGA	6.2	0.421 637	6	7	6	6	6.25
SI	AQGA	0.496 60	0.051 803	0.414	0.608	0.462 00	0.496 00	0.515 75
	GA	0.570 40	0.091 920	0.431	0.735	0.486 50	0.600 00	0.608 00
	GASA	0.647 60	0.102 858	0.434	0.764	0.609 00	0.609 00	0.742 00
	QGA	0.558 00	0.044 944	0.492	0.609	0.504 00	0.573 00	0.594 50
	RQGA	0.528 80	0.058 385	0.431	0.609	0.496 75	0.523 50	0.589 50
LE	AQGA	0.515 20	0.037 526	0.444	0.533	0.510 75	0.533 00	0.533 00
	GA	0.166 30	0.027 769	0.124	0.200	0.151 75	0.164 00	0.198 50
	GASA	0.152 90	0.022 757	0.124	0.200	0.134 50	0.161 00	0.161 00
	GQA	0.153 20	0.012 796	0.129	0.161	0.138 00	0.161 00	0.161 00
	RQGA	0.431 40	0.026 563	0.381	0.444	0.428 25	0.444 00	0.444 00
Fitness	AQGA	1.108 40	0.000 843	1.108	1.110	1.108 00	1.108 00	1.108 50
	GA	1.112 90	0.001 370	1.111	1.115	1.111 75	1.113 00	1.113 50
	GASA	1.113 60	0.001 838	1.111	1.116	1.112 50	1.113 00	1.115 25
	QGA	1.113 60	0.000 966	1.113	1.115	1.113 00	1.113 00	1.115 00
	RQGA	1.110 70	0.001 252	1.110	1.113	1.110 00	1.110 00	1.111 50

N=10

Table S9 Ranks for different algorithms

Algorithm	Mean rank			
	RBN	SI	LE	Fitness
AQGA	1.35	1.80	4.90	1.10
GA	2.95	3.05	2.40	3.60
GASA	3.60	4.55	1.80	4.05
QGA	3.65	2.85	1.80	4.10
RQGA	3.45	2.75	4.10	2.15

Table S10 Friedman test statistics of different algorithms

Variable	Value			
	RBN	SI	LE	Fitness
N	10	10	10	10
Chi-square	19.148	16.123	33.583	30.292
df	4	4	4	4
p	0.001	0.003	0.000	0.000

Table S11 Statistical data of different assembly lines

No.	RBN				SI			
	ST_line	TS_line	U_line	AT_line	ST_line	TS_line	U_line	AT_line
1	12	12	9	5	0.633	3.808	1.291	0.412
2	12	12	9	6	0.894	4.123	2.582	0.608
3	12	12	9	6	0.633	8.515	8.042	0.608
4	12	12	9	5	1.000	3.606	1.291	0.508
5	12	12	9	5	0.633	2.916	4.203	0.533
6	12	12	9	5	0.633	4.301	7.594	0.533
7	12	12	9	5	1.155	2.916	1.291	0.533
8	12	12	9	5	0.633	7.106	1.291	0.505
9	12	12	9	5	1.000	2.916	1.291	0.434
10	12	12	9	6	0.633	4.301	1.291	0.608

No.	LE				Fitness			
	ST_line	TS_line	U_line	AT_line	ST_line	TS_line	U_line	AT_line
1	0.332	0.319	0.296	0.533	1.333	1.150	1.121	1.107
2	0.332	0.319	0.296	0.444	1.400	1.163	1.108	1.110
3	0.332	0.311	0.296	0.444	1.400	1.185	1.108	1.110
4	0.331	0.324	0.296	0.533	1.400	1.155	1.121	1.108
5	0.332	0.324	0.296	0.533	1.400	1.176	1.150	1.108
6	0.332	0.320	0.296	0.412	1.333	1.155	1.108	1.107
7	0.331	0.317	0.296	0.431	1.400	1.184	1.108	1.108
8	0.332	0.311	0.296	0.533	1.333	1.184	1.121	1.108
9	0.331	0.324	0.296	0.533	1.400	1.145	1.121	1.108
10	0.332	0.317	0.296	0.444	1.333	1.174	1.121	1.110

Table S12 Descriptive statistics of different assembly lines

Variable	Assembly line	Mean	Std.	Min.	Max.	P25	P50	P75
RBN	ST_line	12	0.0	12	12	12	12	12
	TS_line	12	0.0	12	12	12	12	12
	U_line	9	0.0	9	9	9	9	9
	AT_line	5.3	0.483 046	5	6	5	5	6
SI	ST_line	0.784 70	0.205 441	0.633	1.155	0.633 00	0.633 00	1.000 00
	TS_line	4.450 80	1.883 078	2.916	8.515	2.916 00	3.965 50	5.002 25
	U_line	3.016 70	2.701 231	1.291	8.042	1.291 00	1.291 00	5.050 75
	AT_line	0.528 20	0.068 715	0.412	0.608	0.487 25	0.533 00	0.608 00
LE	ST_line	0.331 70	0.000 483	0.331	0.332	0.331 00	0.332 00	0.332 00
	TS_line	0.318 61	0.004 818	0.311	0.324	0.315 53	0.319 00	0.324 00
	U_line	0.296 00	0.000 000	0.296	0.296	0.296 00	0.296 00	0.296 00
	AT_line	0.484 03	0.052 522	0.412	0.533	0.440 75	0.488 50	0.533 00
Fitness	ST_line	1.373 23	0.034 560	1.333	1.400	1.333 00	1.400 00	1.400 00
	TS_line	1.167 10	0.015 308	1.145	1.185	1.153 75	1.168 50	1.184 00
	U_line	1.118 70	0.012 755	1.108	1.150	1.108 00	1.121 00	1.121 00
	AT_line	1.108 40	0.001 174	1.107	1.110	1.107 75	1.108 00	1.110 00

N=10

Table S13 Ranks of different assembly lines

Assembly line	Mean rank			
	RBN	SI	LE	Fitness
ST_line	3.5	2.0	3	4.00
TS_line	3.5	3.8	2	3.00
U_line	2.0	3.2	1	1.75
AT_line	1.0	1.0	4	1.25

Table S14 Friedman test statistics of different assembly lines

Variable	Value			
	RBN	SI	LE	Fitness
N	10	10	10	10
Chi-square	30.000	30.000	28.080	28.030
df	3	3	3	3
p	0.000	0.000	0.000	0.000

Table S15 Comparison between the collaborative assembly and basic assembly lines

Theoretical cycle, C_t	Compared parameter	Value			
		One-sided linear assembly, simulated annealing	Double-sided linear assembly, tabu search	U-shaped assembly, genetic algorithm	Collaborative assembly, AFPA-AQGA
70	SI	1	2.2361	3.74	0.49
	LE	0.33	0.35	0.29	0.74
	S	12	12	9	5
90	SI	1.41	26.7067	2.45	0.23–0.31
	LE	0.33	0.2574	0.3	0.82–0.84
	S	9	12	9	4
120	SI	1.18	64.3623	1.29	0.35–0.43
	LE	0.33	0.1986	0.22	0.76–0.79
	S	9	12	6	3
150	SI	1.2	0	1.41	0.24
	LE	0.32	0.3067	0.27	0.89
	S	5	6	6	3

References

Çil ZA, Mete S, Özceylan E, et al., 2017. A beam search approach for solving type II robotic parallel assembly line balancing problem. *Appl Soft Comput*, 61:129-138. https://doi.org/10.1016/j.asoc.2017.07.062

Johannsmeier L, Haddadin S, 2017. A hierarchical human-robot interaction-planning framework for task allocation in collaborative industrial assembly processes. *IEEE Robot Autom Lett*, 2(1):41-48. https://doi.org/10.1109/LRA.2016.2535907

Li XL, Xing KY, Lu QC, 2021. Hybrid particle swarm optimization algorithm for scheduling flexible assembly systems with blocking and deadlock constraints. *Eng Appl Artif Intell*, 105:104411. https://doi.org/10.1016/j.engappai.2021.104411

Özcan U, Toklu B, 2009. A new hybrid improvement heuristic approach to simple straight and U-type assembly line balancing problems. *J Intell Manuf*, 20(1):123-136. https://doi.org/10.1007/s10845-008-0108-2

Rizwan M, Patoglu V, Erdem E, 2020. Human robot collaborative assembly planning: an answer set programming approach. *Theory Pract Log Program*, 20(6):1006-1020. https://doi.org/10.1017/S1471068420000319

Samoueil P, Fattahi P, Ashayeri J, et al., 2016. Bottleneck easing-based assignment of work and product mixture determination: fuzzy assembly line balancing approach. *Appl Math Modell*, 40(7-8):4323-4340. https://doi.org/10.1016/j.apm.2015.11.011

Tavakoli A, 2020. Multi-criteria optimization of multi product assembly line using hybrid tabu-SA algorithm. *SN Appl Sci*, 2(2):151. https://doi.org/10.1007/s42452-019-1863-8

Yan P, Jiao MH, 2011. Research of multi-robot parallel assembly optimization base on PSO-SS. Proc 1st Int Conf on Robot, Vision and Signal Processing, p.220-223. https://doi.org/10.1109/RVSP.2011.31

Yu MY, Yang JJ, 2018. Research on flexible assembly system for multi-variety and small-batch products. *Mech Eng Autom*, (6):39-41 (in Chinese).