International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 5 Issue 5 ǁ May. 2017 ǁ PP. 55-61

An Improved Adaptive Multi-Objective Particle Swarm Optimization for Disassembly Line Balancing Problem

Xiao Shan-li[1], Nie Shan-kun[1]
([1].School of Electric and Electronic Engineering, Shanghai University of Engineering Science, Shanghai 201620, China)

Article Info
Article History

Received:
01 Month Year

Accepted:
01 Month Year

Keywords

disassembly line
multi-objective
fuzzy classification
information entropy

Abstract
With the development of productivity and the fast growth of the economy, environmental pollution, resource utilization and low product recovery rate have emerged subsequently, so more and more attention has been paid to the recycling and reuse of products. However, since the complexity of disassembly line balancing problem (DLBP) increases with the number of parts in the product, finding the optimal balance is computationally intensive. In order to improve the computational ability of particle swarm optimization (PSO) algorithm in solving DLBP, this paper proposed an improved adaptive multi-objective particle swarm optimization (IAMOPSO) algorithm. Firstly, the evolution factor parameter is introduced to judge the state of evolution using the idea of fuzzy classification and then the feedback information from evolutionary environment is served in adjusting inertia weight, acceleration coefficients dynamically. Finally, a dimensional learning strategy based on information entropy is used in which each learning object is uncertain. The results from testing in using series of instances with different size verify the effect of proposed algorithm.

I. Introduction

DLBP is a multi-objective problem that was first described by Gungor and Gupta [1] and has been mathematically proven to be NP-complete by McGovern and Gupta [2]. Exhaustive search works well enough in obtaining optimal solutions for small-sized instance. However, its exponential time complexity limits its application to large sized instances [3]. Although some researchers have formulated the DLBP using mathematical programming techniques [4-5], they quickly become unsolvable for a practical-sized problem due to their combinatorial nature [3]. According to the above reasons, heuristic approaches to attain a (near) optimal condition with respect to objective functions are widely used to solve DLBP.

The DLBP model description
Problem definition and formulation

This paper deals with the most extensively studied disassembly line balancing problem known as a deterministic and single-model of product that undergoes complete disassembly. Based on concept and assumptions made by Gungor and Gupta [6], the disassembly line balancing problem can be defined that a discarded product P is composed of n parts and a part removal is considered as a task OP_i.The tasks sequence is then utilized to group the tasks into workstations of which the number is m. The precedence relationships considered are of AND type and are represented using the immediately preceding matrix $\left[X_{ij}\right]_{n\times n}$ where:

$$X_{ij}=\begin{cases}1, & \text{if task } OP^i \text{ immediately preccdes task } OP^j\\ 0, & \text{otherwise}\end{cases}$$

In order to state the partition of total tasks, we use the assignment matrix $\left[Y_{jk}\right]_{n\times m}$, where:

$$Y_{jk}=\begin{cases}1, & \text{if mth task}(OP^j)\text{is assigned to the kth station}\\ 0, & \text{otherwise}\end{cases}$$

In this paper, the multi-objective DLBP mathematical programming model seeks to fulfill four objectives:

A major constraint is to minimize the number of workstations for a given cycle time CT (the maximum time available at each workstation), this objective is represented as:

$$\min f_1 = m \tag{1}$$

The balance measure is represented as:

$$\min f_2 = \sum_{i=1}^{m} \left[CT - t_i^{'} \right]^2 \tag{2}$$

The third objective is given as follows:

$$\min f_3 = \sum_{i=1}^{n} (i \times h_{PS_i}),\ h_{PS_i} = \begin{cases} 1 & \text{hazardous} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

A demand measure was developed to removal of high demand parts early in the part removal sequence. This measure is represented as:

$$\min f_4 = \sum_{i=1}^{n} (i \times d_{PS_i}),\ d_{PS_i} \in N, \forall PS_i \tag{4}$$

Subject to:

$$\sum_{k=1}^{m} Y_{jk} = 1,\ j = 1, 2, \cdots n \tag{5}$$

$$\left[\frac{\sum_{i=1}^{n} t_i}{CT} \right] \le m^* \le n \tag{6}$$

$$Y_{ik} \le \sum_{k=1}^{m} Y_{jk}, \forall (i, j) \in IP \tag{7}$$

Equation (5) indicates that each task must be assigned to only one station. Constraint (6) guarantees that the number of stations with a workload does not exceed the permitted number. Constraint (7) imposes the restriction that all the disassembly precedence relationships between tasks should be satisfied.

II. IAMOPSO approach for the DLBP

Arithmetic coding and decoding

According to the permutation of n numbers, a priority based zero in-degree topological sorting technique [7] is used to decode the permutation to FDS. This algorithm selects the task from the candidate tasks based on priority and zero in-degree. In the procedure, a task is an available task only if it has not already been assigned to a workstation and all of its predecessors have already been assigned to a workstation.

Defined evolutionary states

Ensured the evolution factor parameter

The evolution factor parameter can be introduced to reflect the diversity and evolutionary status of the population, as detailed in the following equations.

The mean distance of each particle i to all the other particles is as follows.

$$d_i = \frac{1}{NUM - 1} \sum_{j=1, j \ne i}^{NUM} \sqrt{\sum_{k=1}^{D} (x_i^k - x_j^k)^2} \tag{8}$$

Using the p_u represents the evolution factor parameter by the following equation:

$$p_u = \frac{d_g - d_{\min}}{d_{\max} - d_{\min}} \in [0,1] \tag{9}$$

Where, the $d_{\min}$ and $d_{\max}$ are minimum and maximum mean distance, respectively. d_g is the mean distance of globally best particle to all the other particles.

Fuzzy classification

This paper partitions the p_u into four sets, including exploration, exploitation, convergence, and jumping out in each generation.

Exploration: the set is denoted by S_1 and the fuzzy membership function is defined as follows in detail.

$$\mu S_1(p_u)=\begin{cases}0,\ 0\le p_u\le 0.4\\ 5*p_u-2,\ 0.4\prec p_u\le 0.6\\ 1,\ 0.6\prec p_u\le 0.7\\ -10*p_u+8,\ 0.7\prec p_u\le 0.8\\ 0,\ 0.8\prec p_u\le 1\end{cases} \tag{10}$$

Exploitation: the set is denoted by S_2 and the fuzzy membership function is defined as follows in detail.

$$\mu S_2(p_u)=\begin{cases}0,\ 0\le p_u\le 0.2\\ 10*p_u-2,\ 0.2\prec p_u\le 0.3\\ 1,\ 0.3\prec p_u\le 0.4\\ -5*p_u+3,\ 0.4\prec p_u\le 0.6\\ 0,\ 0.6\prec p_u\le 1\end{cases} \tag{11}$$

Convergence: the set is denoted by S_3 and the fuzzy membership function is defined as follows in detail.

$$\mu S_3(p_u)=\begin{cases}1,\ 0\le p_u\le 0.1\\ -5*p_u+1.5,\ 0.1\prec p_u\le 0.3\\ 0,\ 0.3\prec p_u\le 1\end{cases} \tag{12}$$

Jumping-out: the set is denoted by S_4 and the fuzzy membership function is defined as follows in detail.

$$\mu S_4(p_u)=\begin{cases}1,\ 0\le p_u\le 0.7\\ 5*p_u-3.5,\ 0.7\prec p_u\le 0.9\\ 1,\ 0.9\prec p_u\le 1\end{cases} \tag{13}$$

Adjust adaptively the control of PSO parameter

Adjust adaptively the control of PSO parameter is given in Table1.

Table 1. Adjust adaptively the control of PSO parameter

Evolution state	c_1 ($c_{\max}=2.5,\ c_{\min}=1.5, c_1=2$)	c_2 ($c_{\max}=2.5,\ c_{\min}=1.5, c_1=2$)	w
Exploration	$c_1=c_1+a,(a\in[0.05,0.1])$	$c_2=c_2-a,(a\in[0.05,0.1])$	$w=\frac{1}{1+1.5*e^{-2.6*p_u}}$
Exploitation	$c_1=c_1+0.5*a,(a\in[0.05,0.1])$	$c_2=c_2-0.5*a,(a\in[0.05,0.1])$	
Convergence	$c_1=c_1+0.5*a,(a\in[0.05,0.1])$	$c_2=c_2+0.5*a,(a\in[0.05,0.1])$	
Jumping-out	$c_1=c_1-a,(a\in[0.05,0.1])$	$c_2=c_2+a,(a\in[0.05,0.1])$	

Information entropy

Suppose that the number of FDS is NUM, in which includes element of D numbers, R_j represents a set including the j^{th} element of all FDSs, b_{ij} is a number of repeated values in the R_j. P_{ij} represents the occupancy of the j^{th} element of i^{th} FOS in the R_j. The information entropy of the j^{th} element in the all FDSs can be calculated by following formula:

$$H_i = \sum_{j=1}^{D} \frac{P_{ij}}{b_{ij}} \log_2 \frac{1}{P_{ij}} \tag{14}$$

The diversity of the FDSs can be expressed by the average information entropy of all FDSs.

$$H = \frac{1}{NUM} \sum_{i=1}^{NUM} \sum_{j=1}^{D} \frac{P_{ij}}{b_{ij}} \log_2 \frac{1}{P_{ij}} \tag{15}$$

Selection particle of good diversity

In order to further enhance the convergence speed and precision of the algorithm , σ_i was introduced to assess the diversity of corresponding particle whether has good diversity by following formula and the smaller the σ_i is the more diversities the corresponding particle has. The particle with good diversity is denoted by $pbest_{bin}$.

$$\sigma_i = H - H_i \tag{16}$$

Proposed MOPSO Algorithm Procedure for DLBP

The aforementioned procedure is iteratively followed until the maximum number of cycles ($T_{\max}$), which is adopted as the termination criteria, has been traced. The program flowchart of the procedure is shown in Figure1.

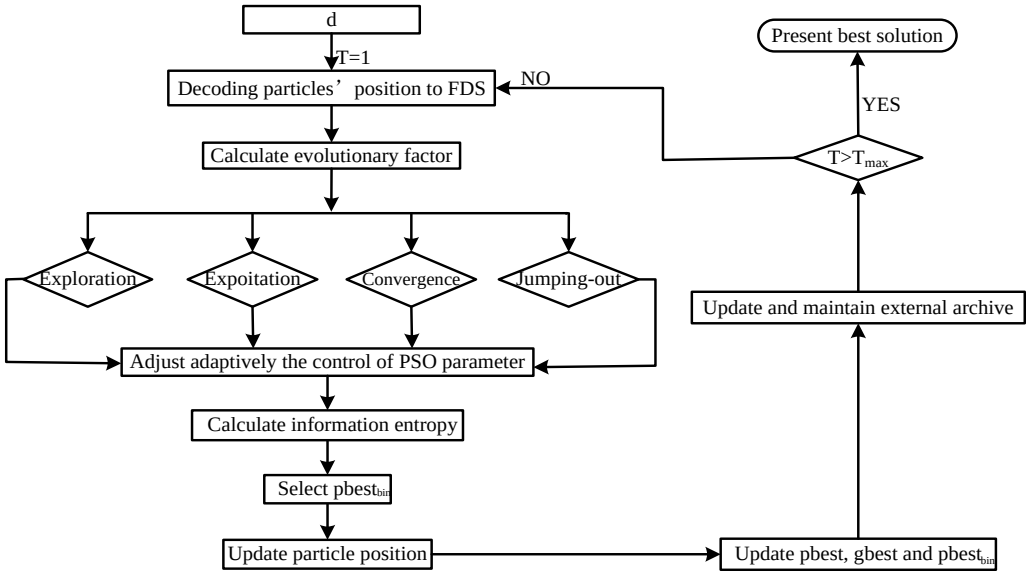

Figure 1. The procedure of improved MOPSO for DLBP

III. Numerical results

The number of the population NUM is set to be 100. The size of the external non-dominated NP is set to 50, a maximum number of iteration $T_{\max}$ is set equal to 300, other parameters: w is set to 0.729, p_c is set to 0.1, p_m is set to 0.4, while p_{u1} is 0.4 and p_{u2} is 0.05. We use a practical example to confirm the improved algorithm's effectiveness for DLBP. The example consists of 25 subassemblies and working cycle of the workstation is 600 seconds (CT=600). The precedence relationship and knowledge database are given in Figure2 and Table 2.

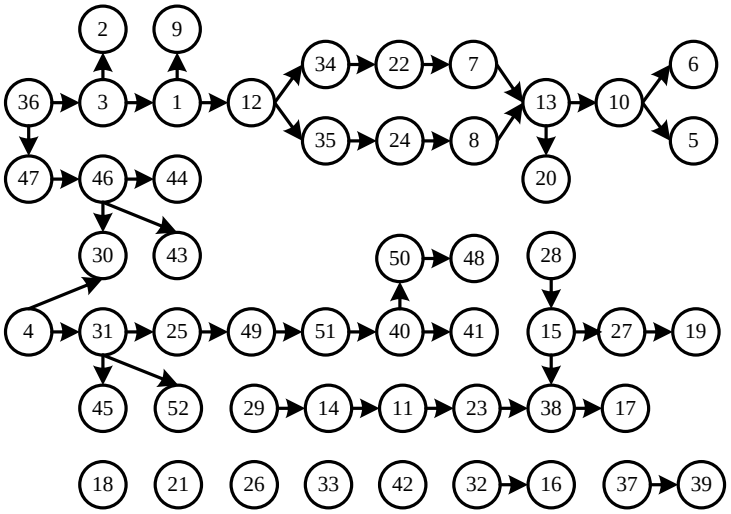

Figure 2. Precedence relationships for the components

Table 2. Knowledge base for 52-part DLBP instance

Part	Removal time(second)	Hazardous	Demand	Part	Removal time(second)	Hazardous	Demand
1	172.95	0	4	27	86.54	0	19
2	45.32	0	2	28	65.49	0	8
3	85.02	0	9	29	72.61	0	13
4	63.47	0	3	30	153.68	0	6
5	235.36	0	15	31	71.92	0	11
6	145.63	1	17	32	68.34	0	15
7	32.44	0	20	33	121.08	1	18
8	32.44	0	6	34	63.98	0	14
9	202.01	0	14	35	71.23	0	15
10	55.28	0	4	36	112.65	0	4
11	28.49	0	19	37	55.32	0	11
12	185.32	0	10	38	35.63	0	8
13	25.41	0	5	39	9.14	1	17
14	17.02	0	14	40	23.25	0	6
15	25.63	0	16	41	14.25	0	17
16	10.22	0	6	42	86.32	1	11
17	52.68	0	16	43	53.61	0	12
18	86.35	0	5	44	35.68	0	5
19	66.38	0	15	45	15.04	0	18
20	122.34	1	13	46	21.46	0	13
21	107.05	0	7	47	44.37	0	8
22	74.85	0	4	48	36.91	0	3
23	194.64	0	12	49	54.73	0	4
24	201.57	0	19	50	48.95	0	5
25	86.54	0	8	51	78.69	0	15
26	65.49	1	3	52	26.58	0	18

Two Pareto optimal solutions obtained by DLBP IAMOPSO on different runs are given in Figure3 and Figure4, the function values are as follows: $f_1 = 7$, $f_2 = 8454.675$, $f_3 = 13966$, $f_4 = 136$ and $f_1 = 7$, $f_2 = 11130.72$, $f_3 = 14686$, $f_4 = 126$. The results of computer simulation indicate that the DLBP IAMOPSO has performance and application prospect involving multiple objectives that are optimized simultaneously.

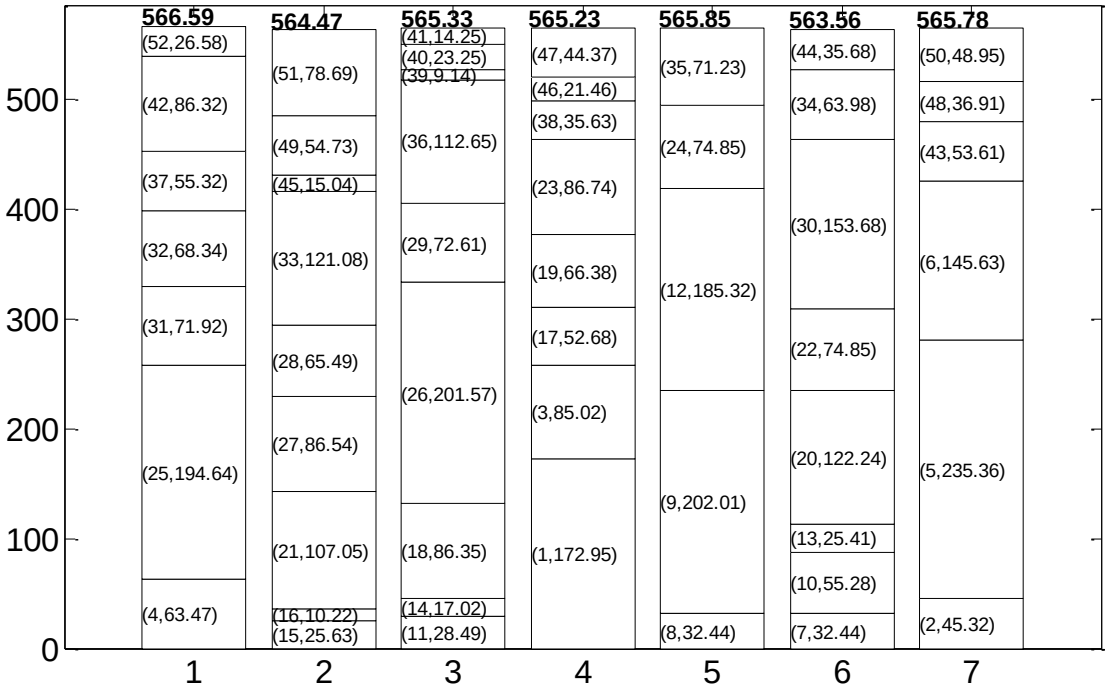

Figure 3. Solution (a) ($f_1 = 7$, $f_2 = 8454.675$, $f_3 = 13966$, $f_4 = 136$)

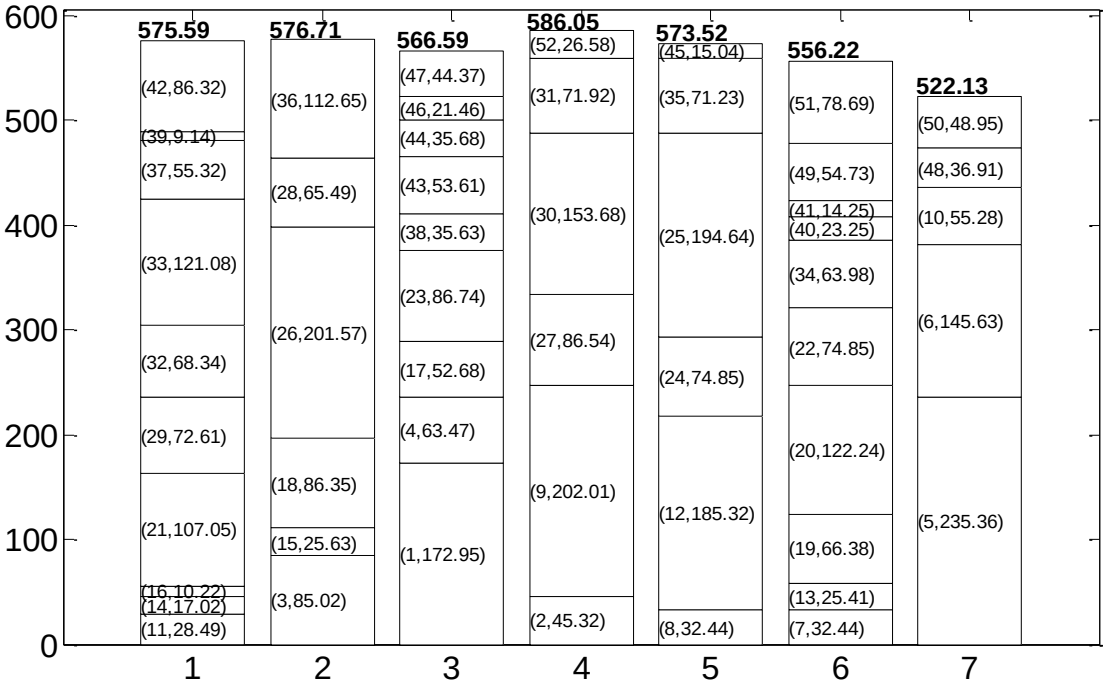

Figure 4. Solution (a) ($f_1 = 7$, $f_2 = 11130.72$, $f_3 = 14686$, $f_4 = 126$)

IV. Conclusion

The main intent of this paper is to build a particle dimension of the particle swarm of the total number of tasks, and through particle swarm optimization method obtains the best disassembly sequence. Both theoretical proof and numerical experiments indicate that this algorithm is superior to the other algorithms in terms of the objective values.

References

[1]. Aşkiner Güngör, Surendra M. Gupta. Disassembly line in product recovery[J]. International Journal of Production Research, 2002, 40(11):2569-2589.
[2]. Mcgovern S M, Gupta S M. A balancing method and genetic algorithm for disassembly line balancing[J]. European Journal of Operational Research, 2007, 179(3):692-708.
[3]. Kalayci C B, Polat O, Gupta S M. A hybrid genetic algorithm for sequence-dependent disassembly line balancing problem[J]. Annals of Operations Research, 2016, 242(2):321-354.
[4]. Özceylan, Eren, Paksoy T. Fuzzy mathematical programming approaches for reverse supply chain optimization with disassembly line balancing problem[J]. Journal of Intelligent & Fuzzy Systems, 2014, 26(4):1969-1985.
[5]. Mete S, Çil Z A, Özceylan E, et al. Resource Constrained Disassembly Line Balancing Problem[J]. IFAC-PapersOnLine, 2016, 49(12):921-925.
[6]. Mcgovern S M, Gupta S M. Ant colony optimization for disassembly sequencing with multiple objectives[J]. The International Journal of Advanced Manufacturing Technology, 2006, 30(5):481-496.
[7]. Dou J, Su C, Li J. A Discrete Particle Swarm Optimization Algorithm for Assembly Line Balancing Problem of Type 1[C]// Third International Conference on Measuring Technology and Mechatronics Automation. IEEE Computer Society, 2011:44-47.

Author Information

Shanli Xiao
Shanghai University of Engineering Science
Longteng road 333, songjiang district, Shanghai
Contact e-mail:582564269*@qq.com*

Shankun Nie
Shanghai University of Engineering Science
Longteng road 333, songjiang district, Shanghai