ELSEVIER

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 29 (2012) 3340 – 3347

Procedia Engineering

www.elsevier.com/locate/procedia

2012 International Workshop on Information and Electronics Engineering (IWIEE)

Integrated Optimization of Production Planning and Scheduling in Mixed Model Assembly Line

QiaoYing Dong[a*], JianSha Lu[a], YuanKun Gui[b]

[a]*Coll. of Mech. Eng., Zhejiang Univ. of Technol., Hangzhou, 310000, P.R.China*
[b]*Key Lab. for Mech. Manuf. & Autom. of Ministry of Educ., Zhejiang Univ. of Technol., Hangzhou, 310000, P.R.China*

Abstract

In order to solve the separation in the traditional serial production planning and scheduling in mixed model assembly line, the integrated optimization complete model of production planning and scheduling based on multiple objectives and constraints was constructed. Since the integrated optimization complete model is difficult to solve, the heuristic approach was adopt, and the modified discrete particle swarm optimization(MDPSO) was presented to solve the model. The experiments verifies the presented model and algorithm can realize the simultaneously optimization of production planning and scheduling in mixed model assembly line and contribute to performance improvement and the application scope expand of the new intelligent optimization.

Keyword: Mixed model assembly line; Production planning; Scheduling; Integrated optimizaiton model; Intelligent algorithm

1. Introduction

In the actual production environment, production planning and scheduling are closely linked and interacted. However, in the traditional serial production planning and scheduling, the two levels are split. In order to eliminate the separation, the integrated optimization modelling of production planning and scheduling is presented. As the final production unit, the mixed model assembly line takes a very important role in manufacturing system. Therefore, in this paper, the integrated optimization of production planning and scheduling in he mixed model assembly line was studied.

[*] * Corresponding author.
E-mail address: dqy0910@163.com

doi:10.1016/j.proeng.2012.01.491

The integrated optimization of production planning and scheduling can be divided into 4 categories[1]. The complete model is one of them, namely construct the integrated optimization model of production planning and scheduling directly and can describe problems precisely. Therefore, in this paper, the complete integrated optimization model of production planning and scheduling in mixed model assembly line will be proposed.

Complete integrated model is very difficult to solve, we select the heuristic method to solve it. The studies on integrated optimization in assembly line are seldom. Yan et al. [2] presented three heuristic methods to achieve simultaneously optimization of production planning and scheduling in assembly line. L. Chen et al[3] used the heuristic method to optimize the production planning and scheduling simultaneously in assembly shops. Particle swarm optimization(*PSO*) is an evolutionary computation algorithm based on populations[4] in 1995. *PSO* is an excellent optimization algorithm in continuous space which has many advantages. The integrated optimization of production planning and scheduling is discrete combination optimization. However, *PSO* should be modified to adapt the discrete combination optimization. Currently, the discretization strategies of *PSO* can be divided into three categories: (1) take the speed as the probability of position changes[5,6]. (2)redefine the *PSO* operations [7]. (3)apply in discrete situation [8] directly. Different from the above methods, we will present new encoding and decoding scheme to achieve algorithm discretization and improve the algorithm performance. The modified discrete particle swarm optimization(*MDPSO*) will be proposed to solve the integrated optimization model.

2. Problem Description

The integrated optimization of production planning and scheduling in mixed model assembly line focuses on two topics: (1)construct the production plan and conduct the batch split or merge; (2) determine the MPS production set of products. Assuming that there are $M(m=1,\dots,M)$ type of products, and the ith type of product have $MF(i)$ workstations. M types of products are produced on the assembly line according to the sequence and flow to each workstation in sequence. The production planning and scheduling can be divided into $NTF(T=1,2,\dots,NTF)$ cycles and a cycle consists of nt time unit.

Firstly, the batch split or merge on ***demand***(i,T) are conducted to obtain the production planning ***production***(i,T); And then aiming at an equipment set and a job set $\boldsymbol{D}_{\mathrm{m}}$ $(m=1,\dots, M)$, the mixed model assembly line scheduling is conducted. We use the Minimum Part Set(*MPS*) to describe a complete production process. *MPS* is a vector representing a product mix, such that $(d_1, \dots, d_M) = (D_1/H, \dots, D_M/H)$, where M is the total number of models, D_m $(m=1,\dots, M)$ is the number of products of model type m that need to be assembled during the entire planning horizon and H is the greatest common divisor or highest common factor of $D_1, D_2, \dots, D_m$. This strategy operates in a cyclical manner. The number of products produced in one cycle is given by D: $D=\sum_{m=1}^{Nm} d_m$. Assuming that the total demand for products is H and demand for models is M in the planned period. Obviously, H/D times the repetition of producing the MPS products can meet the total demand in the planning horizon.

3. Integrated Optimization Modeling of Production Planning and Scheduling in Mixed Model Assembly Line

3.1. Assumptions

According to characteristics of mixed model assembly line, we make the following assumptions:

(1) each working process content and production time of each product are known before planning; (2) the products are produced in sequence and have same production sequence on each workstation;(3) the products arrives at the same time in a cycle and the pre-emptive production is not permitted; (4) a

workstation only can deal with one product at the same time and the different working process of a product cannot be dealt with simultaneously; (5) the corresponding product can be processed right now when the needed equipment become idle; (6) the assembly line has been balanced;(7) different products with similar production characteristics are produced on the assembly line which has a finite number of workstations; (8) the model sequences considered in the study is based on the Minimum Part Set (*MPS*) principle; (9) the travel time of workers are ignored; (10) the rework is ignored.

3.2. Objetive functions

The primary goal of the integrated optimization model of production planning and scheduling in mixed model assembly line is to meet the production demands. In order to reduce the total cost, the following objectives are put forward.

3.2.1. Minimizaing the shortage and overtime cost:

$$f_1 = \min \sum_{T=1}^{NTF} \sum_{m=1}^{M} [over\cos tF(m) \cdot nF\_over(m,T) + short\cos tF(m) \cdot nF\_short(m,T)] \tag{1}$$

where *NTF* is the total cycle number of production plan; *M* is the total type number of demand; ***overcostF***(*m*) is the penalty cost per piece of the redundant products; ***nF_over***(*m*,*T*) is product *m* exceeds the demand in quantity in cycle *T*; ***shortcostF***(*m*) is the penalty cost per piece of product *m* for shortage; ***nF_short***(*m*,*T*) is the shortage quantity in cycle T for product *m*.

3.2.2. Minimizing the total setup cost. In many industries, sequence-dependent setups are considered as an important item in assembly operations. The model considering sequence-dependent setups developed by Kara et al [9] is considered in this paper.

$$f_2 = \min \sum_{T=1}^{NTF} \sum_{j=1}^{D(T)} \sum_{m=1}^{M} \sum_{r=1}^{M} \sum_{k=1}^{Ns(m)} S1(j,m,r,T) \cos t(k,m,r) \tag{2}$$

$$s.t. \quad \sum_{m=1}^{M} \sum_{r=1}^{M} S1(j,m,r,T) = 1, \quad \sum_{m=1}^{M} S1(j,m,r,T) = \sum_{p=1}^{M} S1((j+1),r,p,T), \quad \sum_{m=1}^{M} S1(j,m,r,T) = \sum_{p=1}^{M} S1((j+1),r,p,T) \tag{2.1}$$

$$\sum_{m=1}^{M} S1(D(T),m,r,T) = \sum_{p=1}^{M} S1(1,r,p,T), \sum_{j=1}^{D(T)} \sum_{r=1}^{M} S1(j,m,r,T) = d_m(T) \tag{2.2}$$

where ***D***(*T*) is the MPS total product amount in cycle *T*; ***N*s**(*m*) is the total workstation amount of model *m*; ***S1***(*j*,*m*,*r*,*T*) is a sign function, ***S1***(*j*,*m*,*r*,*T*) is *1* if model m and r are assigned, respectively, at position *j* and *j+1* in a sequence in cycle *T*; otherwise it is *0*; ***cost***(*k*,*m*,*r*) is the setup cost required when the model type is changed from *m* to *r* at station *k*.

3.2.3. Minimizing the total idle-overload cost. It is assumed that the stations are all closed types which have boundaries workers cannot cross. Such a closed station is often found in reality in which the use of facilities is restricted within a certain boundary. A possible situation in the assembly line in a launch cycle is idle phenomenon: after a product is processed, the next product has not been launched onto the line, and the worker has to return the starting point to wait. Another possible situation is overload phenomenon: the worker does not complete the processing of a product at the station and the remained work need the assisting worker to complete.

The idle phenomenon[10] and overload phenomenon will cause the waste of time and cost in production. In order to minimize the total idle-overload cost, we designed the objective function of mixed model assembly line as follows:

$$f_3 = \min \sum_{T=1}^{NTF} \sum_{m=1}^{M} \sum_{k=1}^{Ns(m)} \sum_{j=1}^{D(T)} (idl(k,j,T) \cdot idl\cos t + ovt(k,j,T) \cdot over\cos t) \tag{3}$$

$$idl(k,j) = \max\{0, C - ET(k,j)\}, \quad ovt(k,j) = \max\{0, ST(k,j) + T(k,j) - L(k)/V_c\} \tag{3.1}$$

$$\mathrm{ET}(k,j) = \min\{ST(k,j) + T(k,j), L(k)/Vc\}, \quad ST(k,j+1) = \max\{0, ET(k,j) - C\}, \quad T(k,j) = \sum_{m=1}^{M} S2(m,j) \cdot t(m,k) \tag{3.2}$$

where ***idl***(k,j), ***ovt***(k,j) is the idle and overload time caused by the product j in the sequence at station k respectively; ***T***(k,j) represents the operating time of model j in station k; ***L***(k) is the length of station k; ***ST***(k,j) and ***ET***(k,j) are starting time and ended time of product j at station k. The conveyor system moves at a constant speed V_c, the launch interval of products is constant C; ***t***(m,k) represents the operating time of model m in station k, ***S2***(m,j) is 1 if the product j in the sequence is model m, otherwise ***S2***(m,j) is 0.

3.3. The Complete Integrated Optimization Model of Production Planning and Scheduling in Mixed model assembly line

There are conflict and competition among objectives of the multi-objective optimization problem. We transform the objective into cost and present the following integrated optimization model:

$$f=\min\sum_{T=1}^{NTF}\{\sum_{j=1}^{D(T)}\sum_{m=1}^{M}\sum_{r=1}^{M}\sum_{k=1}^{Ns(m)}S1(j,m,r,T)\cos t(k,m,r)+\sum_{m=1}^{M}[over\cos tF(m)\cdot nF\_over(m,T)+short\cos tF(m)\cdot nF\_short(m,T)] \tag{4}$$

$$+\sum_{m=1}^{M}\sum_{k=1}^{Ns(m)}\sum_{j=1}^{D(T)}(idl(k,j,T)\cdot idl\cos t+ovt(k,j,T)\cdot over\cos t)\}$$

s.t. equation (1)~(3) .

The complete integrated optimization model fully considers the working characteristics of mixed model assembly line and integrated the objectives and constraints of production planning and scheduling.

4. The Algorithm Design of the Integrated Optimization Model of Production Planning and Scheduling in Mixed Model Assembly Line

4.1. Encoding and decoding design

The key of the evolution algorithm is the encoding and decoding of the solution. It will influence the efficiency of the algorithm. The integrated optimization in Mixed model assembly line refers to the production plan and production mix. We use the segmented encoding mode: ***X*1** represents the production plan, ***X*2** is the corresponding production mix. Assuming that there are 3 type of products to assemble: *A*, *B*, *C*, then the problem description are shown in Fig. 1.

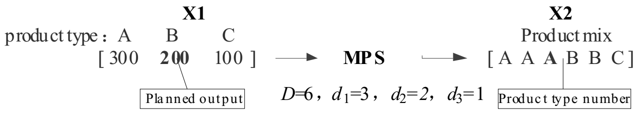

Fig.1. Problem description

JIT mode is common in assembly, ***X*1** usually is decided by demand and ***X*2** consists of different type of product mix. Clearly, in mixed model assembly line sequencing, D products of M models need to be launched onto the line in a MPS. A particle (possible solution) should be a sequence with D products of M models.

As shown in Fig. 1, we have three models represented by *A, B* and *C*, let *A, B* and *C* corresponding to number *1, 2,* and *3*, $d_1=3$, $d_2=2$, $d_3=1$, $D=6$, a possible sequence in a *MPS* is: *111223*. Obviously, the dimension of solution is D, and the code is consists of discrete numbers. This is a discrete permutation that the standard PSO cannot be used directly since the real-valued positions of particles. So, we proposed the mapping scheme as follows: Firstly, let the discrete number *1, 2, 3, 4, 5, 6* to represent the products, 1,2,3 represent product *A*, 4,5 represent product *B*, 6 represent product *C*. Secondly, in order to mapping the discrete coded sequence with the real-valued position of particle, we chose the random key representation proposed by Bean [11]. The random key representation encodes a particle (solution) $\boldsymbol{X}_i$ with real random numbers. These values are used as sort keys to decode a solution. In our sequencing problem, each particle's component corresponds to a product of a certain model. To form a sequence, we generate a random number in (*0*, *1*) for each dimension. The mapping to the permutation's space is accomplished by sequencing these numbers in ascending order.

Through this encoding and decoding schemes, we can not only take full advantages of the *PSO* algorithm but also achieve the mapping between the real-valued position of particle and discrete representation of solution. The presented encoding and decoding method can discretize the *PSO* successfully and make it can be used in discrete problem directly without unfeasible solution. So the complexity of the presented algorithm will be reduced and the efficiency will be improved.

4.2. The improvement scheme for algorithm performance

Name the algorithm which only achieve the discretization by encoding and decoding as Basic Modified Particle Swarm Optimization(*BMDPSO*). In the purpose of improving the algorithm performance, the further improvement on evolution scheme is introduced and then we obtain the *MDPSO*.

4.2.1. Self-adaptive Escape Scheme. In *PSO*, lack of diversity of the swarm, particularly during the latter stages of the optimization, was understood as the dominant factor for the convergence of particles to local optimum solutions prematurely. So we introduced a self-adaptive escape scheme [12] to improve the searching ability and diversity of swarms. The self-adaptive escape scheme uses the reference of migrating habit of species when population grows too dense in biome, and conducts an escape when the flying velocity of particle is relatively low so that to confirm the effective global search.

The self-adaptive escape scheme is a direction-preconcerted and time-preconcerted mutation operator. A mutation operation on the velocity is conducted when the velocity smaller than a threshold T_j:If ($v_{ij}<T_j$) then $v_{ij}=rand\times V_{max}$, where $T_j>0$ is the current threshold velocity of *j*th position of the particle, rand is a random number between (*0*,*1*). The self-adaptive escape scheme is formulated as:

$$F_j(t)=F_j(t-1)+\sum\nolimits_{i=1}^{Np}b_{ij}(t),\quad b_{ij}(t)=\begin{cases}0, & v_{ij}(t)>T_j\\ 1, & v_{ij}(t)<T_j\end{cases},\ If\quad F_j(t)>k_1\quad then\quad F_j(t)=0;\ \ T_j=T_j/k_2 \tag{5}$$

where frequency $F_j(t)$ represents the escaping times of velocity *j*, constant k_1 is the condition value for regulating T_j and $F_j(t)$; constant k_2 is used to regulate the decreasing range of threshold.

This scheme regulates the relation between local search and global search and improves the diversity of the swarm effectively by adjusting each velocity of particle.

4.2.2. The improvement of searching ability. In order to make the particle swarm searching in the whole solution space at the early stages and converging to the optimal solution rapidly, let $w(t)$=1.2-0.8×t/Ng, where t is the evolution generation, Ng is the total evolution generation. Accompanying the increase of the evolution generation, the weight $w(t)$ decreasing gradually and the particle swarm will achieve local adjustment in this area.

4.2.3. The evolution equation of MDPSO. To sum up, the evolution equation of *MDPSO* is shown as:

$$v_{ij}(t+1)=w(t)v_{ij}(t)+c_1r_1(p_{ij}(t)-x_{ij}(t))+c_2r_2(g_{ij}(t)-x_{ij}(t)) \tag{6}$$

$$x_{ij}(t+1)=x_{ij}(t)+v_{ij}(t+1) \tag{6.1}$$

$$w(t)=1.2-0.8\times t/Ng,\ v_{ij}=rand\times V_{\max},\quad s.t.\ v_{ij}<T_j \tag{6.2}$$

5. Computational Results and Discussions

5.1. The experimental parameter

In order to valid the effectiveness of the proposed integrated optimization model and the modified algorithms, we conduct a series of experiments.

Considering an assembly line with four models and five work stations, the yield demand for four models and corresponding *MPS* are shown in Table 1(a), the assembly time of four models at six work stations is listed in Table 1(b). We construct the integrated optimization model of production planning and scheduling in this mixed model assembly line, and employ the presented *MDPSO*, *BDPSO* and the classic Genetic algorithm (*GA*) to solve it.

Table 1. (a)The demand of four models and corresponding MPS (b)The assembly time (unit: s) of four models AT six work stations (for the front suspension assembly)

Cycle T	Product number				MPS
	A	B	C	D	
1	400	200	100	50	d_1=8,d_2=4,d_3=2,d_4=1
2	120	60	80	40	d_1=6,d_2=3,d_3=4,d_4=2

Product name	Product number	Workstation number					
		1	2	3	4	5	6
A series	A	58	52	73	42	72	50
B series	B	45	65	42	45	62	52
C series	C	48	55	78	68	80	48
D series	D	32	40	41	52	42	36

The running horizon of algorithm is: matlab6.5, *CPU*: Intel(R) Core(TM)2 6320,1.86GHz, and 1GB memory. The algorithm parameters are as follows:(1) *MDPSO*: learning factor C_1=2, C_2=2, weight w=[0.4,1.2], K_1=5， K_2=5, particle swarm size *N*p=20, evolution generation *N*g=100. (2) *BMDPSO*: learning factor C_1=2, C_2=2， weight w=0.9, particle swarm size *N*p=20, evolution generation *N*g=100. (3) *GA*: crossover factor pc=0.85, variation factor pm=0.1, Chromosome size *N*p=20, evolution generation *N*g=100.

5.2. The algorithm performance and model solution analysis

5.2.1. The algorithm evolution performance comparison. In order to analyze the influence of the improving scheme, we compared the MDPSO and BDPSO. For the purpose of validating the performance of *MDPSO*, we compared *MDPSO*, *BDPSO* and the classic *GA*. The chromosome representation of *GA* is digital string; the objective function of *GA* is same as *MDPSO*; the selection scheme is roulette; single point crossover and two variants. The evolution process comparison of these algorithms is shown in Fig. 2.

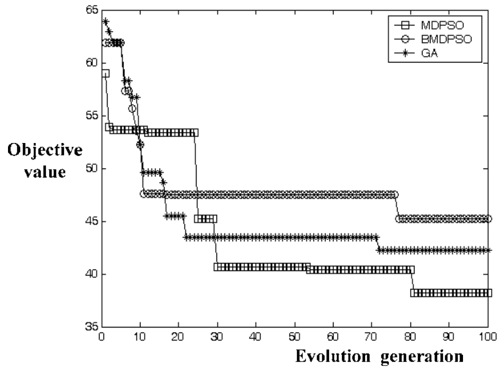

Fig. 2. The evolution process of 3 algorithms

(a) *GA* stop evolving at generation 72, *BMDPSO* found the optimal solution at generation 77 and *MDPSO* at 80. This phenomenon proves the optimizing performance of all the three algorithms is quite good;(b) But the diversity and the unti-premature performance of *MDPSO* is better than *BDPSO* and *GA*; (c) The evolution speed and the optimal solution of *MDPSO* are better than *BDPSO* and *GA*; (d) The optimal objective value of *MDPSO*, *GA* and *BDPSO* are 38.211, 43.1958 and 45.1958 respectively. It means *MDPSO* has the best performance in 3 algorithms.

5.2.2. The solution of the integrated model of production planning and scheduling. In the first cycle, the optimal production set of *MDPSO*, *BDPSO* and *GA* is *AAABBBBCCDAAAAA*, *BCCDAAAAAAAABBB*, *AAAABBBBCCAAAAD*; and the completion time in a *MPS* is 1295S,1334S and 1615S respectively; In the second cycle, the optimal production set of *MDPSO*,*BDPSO* and *GA* is *AAADCCCCBBBDAAA*, *AAAACCCCABBBDDA*, *AABBBCDDACCCAAA*; and the completion time in a *MPS* is 1322S, 1324S and 1589S respectively. Obviously, MDPSO has the shortest completion time.

6. Conclusion

The integrated optimization model and its solving algorithm of production planning and scheduling in mixed model assembly line were presented in this paper. (1) In the model construction, the definition of the complete integrated optimization modelling of production planning and scheduling in mixed model assembly line were proposed firstly, and then the model based on multiple objectives and constraints was constructed. (2) In the model solving, in order to overcome the difficulty in the complete integrated optimization model solving, the heuristic approach in was adopt. The encoding, decoding and evolution way of *PSO* was improved and the modified discrete particle swarm optimization was put forward to solve the integrated optimization model.

Acknowledgement

Acknowledgements to the National Natural Science Foundation of China (Grant No. 70971118), Zhejiang Province Department of education scientific research project of China(Grant No. Y201121984) and Zhejiang Provincial Natural Science Foundation of China(Grant No. Y1110994) for supplied support.

References

[1] Maravelias Christos T, Sung C. Integration of production planning and scheduling: Overview, challenges and opportunities. Comput Chem Eng 2009; 33(12):1919-1930.

[2] Yan HS, Xia QF, Zhu CR. Approaches to Simultaneous Production Planning and Scheduling in Automobile Assembly Workshops. Acta Automatica Sinica 2002; 28(6):911-919. (in Chinese)

[3] Chen L, Yan HS, Liu T, Liu XL. Approaches to Integrated Optimization of Production Planning and Scheduling on Automobile Assembly Lines. Computer Technology and Development 2009; 19(1):134-138. (in Chinese)

[4] Kennedy J, Eberhart RC, Particle Swarm Optimization. Proceedings of IEEE International Conference on Neural Networks, Perth, Australia: IEEE Service Center, Piscataway, NJ; 1995, p.1942-1948.

[5] Kennedy J, Eberhart RC, A discrete binary version of the particle swarm algorithm. Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC '97), Orlando, Fla, USA: IEEE Press; 1997, p.4104-4108.

[6] Liao CJ, Tseng CT, Pin L. A discrete version of particle swarm optimization for flowshop scheduling problems. Comput Oper Res 2007; 34(10):3099-3111.

[7] Lian ZG, Gu XS, Jiao B. A novel particle swarm optimization algorithm for permutation flow-shop scheduling to minimize makespan. Chaos Soliton Fract 2008; 35(5):851-861.

[8] Salman I Ahmad, Madani SA. Particle swarm optimization for task assignment problem. Microprocess Microsy 2002; 26:363-371.

[9] Kara Y, Ozcan U, Peker A. Balancing and sequencing mixed-model just-in-time U-lines with multiple objectives. Appl Math Comput 2007, 184(2):566-588.

[10] Song HM, Han YQ, Yang H. Mixed-model assembly line balancing design. China Mechanical Engineering 2003; 14(06):475-479. (in Chinese)

[11] Bean JC. Genetic Algorithms and Random Keys for Sequencing and Optimization. ORSA Journal on Computing 1994; 6(2):154-160.

[12] He R, Wang YJ, Wang Q, Zhou J, Hu C. An improved particle swarm optimization based on self-adaptive escape velocity. Journal of Software 2005; 16(12):2036-2044. (in Chinese)