Journal of Applied Sciences

ISSN 1812-5654

Journal of Applied Sciences 13 (13): 2429-2435, 2013
ISSN 1812-5654 / DOI: 10.3923/jas.2013.2429.2435

Research on Engine Assembly Line Balancing Based on an Improved Genetic Algorithm

[1]Fan Shu, [1]Zi-Qi Xu, [2]Chao Mi and [1]Xiao-Ming Yang
[1]Logistics Engineering College, Shanghai Maritime University, 201306, Shanghai, China
[2]Container Supply Chain Technology Engineering Research Center, Shanghai Maritime University, 201306, Shanghai, China

Abstract: This study, pertinent to the problem that the traditional process optimization of assembly line optimizes only for single target but neglects other important elements, builds the mathematical model for the multi-objective, including maximizing the capacity, balancing the load and minimizing the cost, process optimization of the assembly line of the passenger car engine. Based on the features of the passenger car engine assembly, such as complex steps, too many stations, expensive tools and great market demands, this paper adopts the GASA algorithm combining the genetic algorithm and the simulated annealing algorithm for solution. Experiments show that such model and algorithm can solve the process optimization problems of the passenger car engine assembly line effectively.

Key words: Engine assembly line, multi-objective, process optimization, GA, GASA algorithm

INTRODUCTION

Along with the fast development of modern industry, the passenger car ownership increases daily and the manufacturers are attaching more and more importance to improve the management quality while pursuing for better quality. Since, the engine assembly line is a technology-intensive manufacturing step of the passenger car enterprise and the engine quality is crucial to the overall car quality, the process of the engine assembly line turns to be a hot research item (Erel and Sarin, 1998).

The engine process is to distribute the general production flow of the engine into different stations on the assembly line reasonably and effectively. It involves in a lot of elements, including the load balance of the engine station, the comprehensive balance of the human-machine efficiency and the configuration of the tools and frocks (Douma *et al*., 2011).Thus the process optimization is actually the portfolio optimization and the changes of the sequence of different work steps decided by product design, process design and the manufacturing techniques make the process problem rather complex (Peng *et al*., 2011).

At present, the researches on the process reasonability focus on the balance of the assembly lines with the target to make the stand-by time of different stations on the assembly line as close to each other as possible. However, the manufacturers, while pursuing for high balance rate, may choose to realize such target at the cost of capacity loss and increasing cost. Therefore, it is urgently needed to evaluate the process performance from multiple angles. To build the multi-objective process optimization model and work out solution then turns into a key research direction of the assembly line of passenger cars.

RELATED WORK

The process optimization problems of the engine assembly line lies at the researches on the assembly line balance issues. The assembly line balance issues contain two types (Xiao, 2010). Targeting to minimize the work station quantity, Mao and Zheng (2010), Fan *et al*. (2010) carried out researches on the first type balance issues, trying to reduce the construction cost of the production line and such researches are greatly used in the process planning stage. The second type of balance issues had been analyzed (Jian-Sha *et al*., 2010), aiming to minimize the production steps and such analysis exists along with the ongoing optimization of the production line.

These two types of researches focus on single objectives from different angles. However, the assembly line of the passenger car engine is featured with complex process procedures, great many of work stations, great market demands and too many tools and frocks. Just take the frocks and tools for example, if the process design can save the configuration of some tools and frocks, considerable cost can be saved. Thus, it is not enough for the process design and optimization of the engine assembly line to consider only the balance issues. All influential elements should be considered

Corresponding Author: Chao Mi, Container Supply Chain Technology Engineering Research Center, Shanghai Maritime University, 201306, Shanghai, China

comprehensively in order to study the multi-objective optimization of the assembly line of the passenger car engine(Wang., 2012).

As for the portfolio optimization issues, including the process optimization, of the assembly line, (Klein and Scholl, 1996) use the branch and bound method for solution. However, this method belongs to the mathematical programming approach and is applicable only to solve small-scale issues and can hardly apply to the large-scale multi-objective assembly lines of the passenger car engines that have too many process procedures; Fan *et al.* (2010) argues to use the genetic algorithm for solution, but it may easily lead to partial optimization since the process procedures of the passenger car is too many, the priority of the procedures are always changing and the genetic algorithm has a fast convergence rate.

Therefore, considering the process design of the engine corporate and the actual optimization needs and targeting on three elements including the capacity, balance and cost, this paper builds the multi-objective mathematical algorithm of the assembly line of passenger cars. Meanwhile, pertinent to the features of the assembly line, this paper adopts the GASA algorithm which inherits not only the fast convergence rate of the genetic algorithm but also the strong solving ability of the simulated annealing algorithm. Finally, in order to enhance the production efficiency and cut down the production cost, this paper provides feasible solutions for the assembly line problems of the passenger car engine.

MULTI-OBJECTIVE OPTIMIZATION MATHEMATICAL MODEL

Model definitions: It is assumed that there is one assembly line equipped with one engine and requires totally m procedures distributed to n stations. Then we define:

- I: Indicate the procedure No. i (I = {1,2, ..., m}, i∈I)
- j: Indicate the station No. J : (j = {1, 2,..., n}, J∈J)
- t_i: Indicate the working hours of the procedure No. i to reflect the working speed of this procedure, i∈I
- e_i Indicate the quantity of equipments required by the procedure No. i. i∈I

x_{ij} to indicate whether the procedure i is assigned to the station j, it is the decision variable of the process optimization:

$$X_{ij} = \begin{cases} 1, \text{the procedure has been assigned to station} \\ 0, \text{the procedure hasn't been assigned to station} \end{cases}$$

O_j to indicate the working hours of the working hours of the station j, that is, the working cycle, $Oj = t_i X_{ij}$

P_j to indicate the calculated value of the equipment demands on the station j, $P_j = e_i X_{ij}$

D_{ab} to indicate the precedence relationship of the procedures a and b, $P_j = e_i X_{ij}$:

$$D_{ab} = \begin{cases} 1, \text{the procedure a goes before the b} \\ 0, \text{there is no limitation on the precedence of procedures a and b} \end{cases}$$

Thus, we can model for the process optimization according to the above mentioned variables and intermediate variables.

Model objectives: According to the features of the engine assembly, that is, complex procedures, too many work stations, expensive frocks and tools and great market demands, the mathematical model of the process optimization should take the capacity, balance, cost and other elements into consideration to build the multi-objective process optimization model for the engine assembly line (Fattahi *et al.*, 2012). Focusing on the dimension difference among different objectives, this paper adopts the self-adaption weighting method to adjust the weight of different objectives at real time and the objective function is shown in the following formula (1).

$$\text{Min } Z = \alpha\phi_1+\beta\phi_2+\gamma\phi_3 \tag{1}$$

In which, α, β, γ is the weight of different indexes, respectively, ϕ_1 the capacity index as shown in Formula (2);ϕ_2 is the balance index as shown in Eq. 3 while ϕ_3 is the cost index as shown in Eq. 4:

$$\varphi_1 = \frac{n\times \max(t_i X_{ij})}{\sum_{i=1}^{m} t_i} \tag{2}$$

$$\varphi_2 = \sqrt{\sum_{j=1}^{n}(\max(O_j)-O_j)^2} = \sqrt{\sum_{j=1}^{n}(\max(t_i X_{ij})-t_i X_{ij})^2} \tag{3}$$

$$\varphi_3 = \sum_{j=1}^{n} Q_j \tag{4}$$

Considering that the most common equipment used in the engine assembly is the electric gun or the pneumatic gun used to tighten the bolt, which then result in the consequence that the calculated value on the station j needing more than one equipment is always reflects that the actual need is only one equipment, thus in Eq. 4, e_j reflects the actual value of the equipment needs on the station j and its values are: $e_j = 1$ (when $P_j>0$) or $Q_j = 0$ (when $P_j = 0$). Finally the multi-objective function of the mathematical model for process optimization is built.

Model constraints: If the station j can complete i procedures, then the model should meet following constraints:

$$\sum_{j=1}^{n} X_{ij} = 1(\forall i \in I) \quad (5)$$

Equatoin 5 ensures that each procedure i can and can only be assigned to one station j, Eq. 5:

$$\sum_{i=1}^{m}\sum_{j=1}^{n} X_{ij} = m \quad (6)$$

Equation 6 guarantees that the total quantity of the assembly operations of each station j is the equal to the total quantity (m) of procedures:

$$X_{kh} \le \sum_{j=1}^{h} X_{lj} \quad (7)$$

If the procedure l goes before the procedure k, that is, when D_{lk} = 1 and if the procedure k is assigned to the station h, then the procedure l will for sure be assigned to station h or afterwards. Equation 7 then ensures that procedures ∀k, l can meet the precedence relationship for assembly of procedures k, l.

SOLUTIONS TO THE MULTI-OBJECTIVE PROCESS OPTIMAZATION

GASA algorithm design: A lot of experts and scholars use the heuristic algorithms, such as the ant colony algorithm (Mao and Zheng, 2010), the genetic algorithm (Mutlu *et al*., 2013), the particle swarm algorithm and the simulated annealing algorithm (Suresh and Sahu, 1994), to solve the assembly line problems. In all these algorithms, the Genetic Algorithm (GA) has advantages such as fast convergence rate and strong parallel capacity, but it has the defect of "early maturing". And since the engine assembly line has too many procedures (Chica *et al*., 2012), the traditional GA will omit the optimum solution. Thus, its search capacity shall be enhanced. The Simulated Annealing algorithm (SA) has similar thoughts with the physical annealing, that is, it has features of weak dependence on initial solution and strong search capacity. However, it runs slowly (Wang *et al*., 2005). Therefore, this paper mixes the SA into GA and uses SA to strengthen the solution rang while ensuring the overall speed of the algorithm is reasonable and finally obtains the optimum solution (Wang *et al*., 2012). Figure 1 for the GASA algorithm flow.

The left half part of Fig. 1 shows the process of GA algorithm, the step "generate station division" is produced dynamically under the feasible process sequence and the dynamic allocation method is adopted. Fig. 2.

In Fig. 2, the "generate station division" method is as follows:

- **Step 1:** Initialize cycle: calculate the average processing time T_{avg} and choose the bigger one from T_{avg} and max (t_i) as the minimum theoretical cycle:

$$T_{avg} = \frac{\sum_{i=1}^{m} t_i}{n} \quad (8)$$

  In which, m is the procedures quantity, t_i is the time consumed by the procedure i and n is the station quantity
- **Step 2:** Take the minimum theoretical cycle as the basis to assign the procedures to n stations and calculate the average time consumption of each station T_{j1} (j1 = 1, 2,..., n) and solve the cycle CT.
- **Step 3:** Calculate the time consumption of each station after the potential increment treatment: $T_{j2} = T_{j1}+t_{ji}$ (j1 = 1, 2, ..., n) is the time of the first procedure assigned to the station (j1+1) and work out the cycle CT'
- **Step 4:** Check whether CT ≤CT, if so, then CT is the minimum cycle and such allocation scheme is the optimum, or otherwise take CT' as the minimum theoretical cycle and return to Step 2

After selecting the variant chromosome and then imbed the SA algorithm into GA (Gurevsky *et al*., 2013). You can refuse the offspring whose gene doesn't meet requirements to ensure that the optimum solution can be saved to the final. The realization flow is shown in the right part of Figure 1 and the steps are as follows:

- **Step 1:** Judge the outer circulation: whether the appointed temperature is reached? End if so, or carry out the next step
- **Step 2:** judge the inner circulation: whether the appointed times are reached? Forward to Step 8 if so or carry out the next step
- **Step 3:** Variation. Generate a variation point randomly and cut the parent chromosome into two parts. Save the front part and realign and combine the back part chromosome according to the precedence matrix and then get the offspring chromosome. Since the generation of the back half chromosome depends

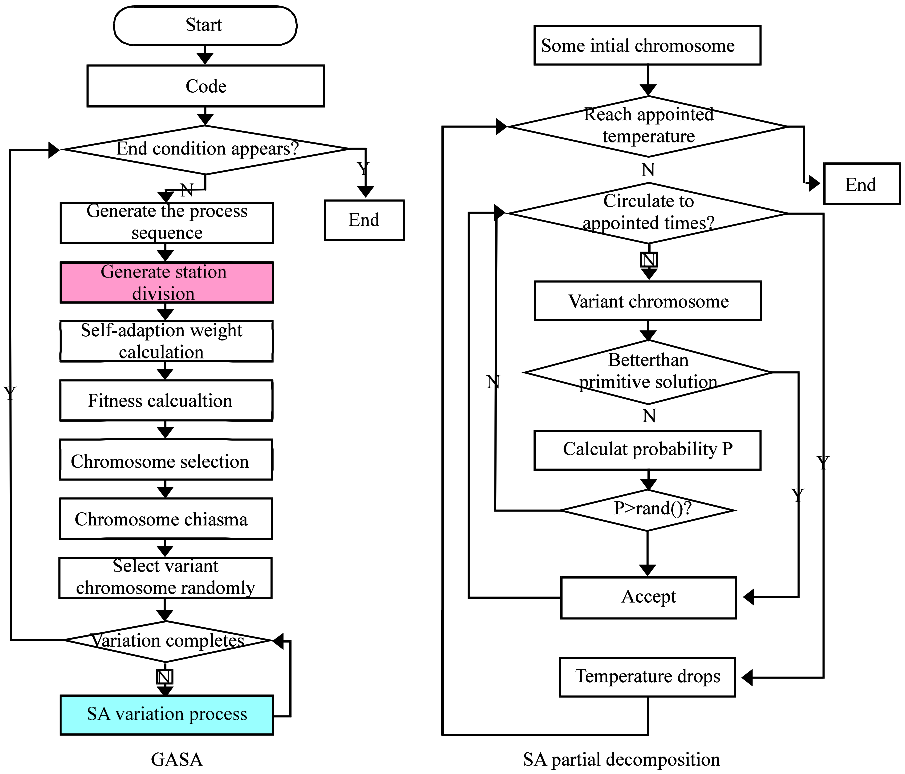

Fig. 1: Flow of GASA algorithm

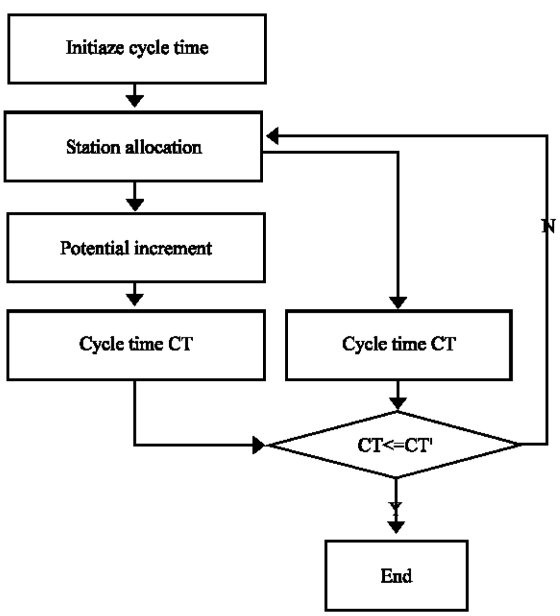

Fig. 2: Logic flow chart of "Generate Station Division"

on the precedence matrix, thus the feasibility of the chromosome is ensured

- **Step 4:** Judgment. Calculate the energy value of both the parent and the offspring chromosome and mark them as E1and E2. Forward to Step 7 if $\Delta E = E_2 - E_1 < 0$ or carry out the next step
- **Step 5:** Probability calculation. According to the Metropolis law, the probability of the particles towards balance under temperature T is P= e-ΔE/(KT), here K is a Boltzmann constant
- **Step 6:** judgment. Generate a digit between 0-1 randomly and carry out the next step if this digit is bigger than the above probability, otherwise transfer to Step 2
- **Step 7:** Accept the variant offspring chromosome to replace the parent chromosome and transfer to step 2 until the inner circulation end conditions appear
- **Step 8:** Cool down the temperature. T' = K*T, here T is the cooling constant. Transfer to Step 1 until the outer circulation end conditions appear

GASA-based model solution process

Code: Firstly, carry out traditional algorithm code to the process optimization object. The process optimization mainly solves problems at two aspects: arrange the procedures and allocate the stations. We code to solve the presentation of the procedure arrangements.

Considering that the engine has too many procedures, for the purpose of convenient expression, this study adopts the natural numbers for coding. That is, match each procedure to one natural number and the sequence of the natural number is the precedence order of the processing procedures. For example, 1 3 2 4 6 8 7 5 is a series of chromosome and it indicates that the procedure 1 comes first and then procedures 3, 2, 4, 6, 8, 7 and finally 5. According to the coding method and the relationship matrix, the computer then generates a group randomly and each chromosome string inside the group is a feasible solution to the problem.

Fitness calculation and chromosome selection: The establishment of fitness function depends on the objective function. According to the above contents, the objective function is made up of three parts, thus the fitness function should also be cut into three pieces:

$$\text{Fitness} = \alpha_n/\phi_1+\beta_n/\phi_2+\gamma_n/\phi_3 \quad (9)$$

In which, ϕ_1, ϕ_2, ϕ_3 are the same with those in the objective function in Eq. 1; α_n, β_n, γ_n are the dynamically calculated index weights under the self-adaption algorithm. Weights are the standards to reflect the importance level of the indexes. Under normal conditions, the indexes need to be compared belong to the same dimension and the weight sum of all the indexes is 1. Pertinent to the model built in this study, the adaptability weight method is used since the artificial weight allocation is not objective because the fitness value ranges of these three parts differ a lot from each other.

Later, the roulette approachFan *et al.* (2010) is used. The selective probability of each individual is calculated according to their fitness value:

$$P_i = f_i / \sum_{j=1}^{n} f_i$$

Here P_i is the fitness value selective probability, f_i the fitness value of some chromosome and N the group size.

Crossover and variation: The purpose of crossover is to produce better chromosome and the key is how to keep the feasibility of the chromosome, that is, the sequence of the procedures after the crossover will still meet the precedence relationship matrix of the procedures.

This study adopts the two-point crossover method and the crossover process is as follows:

- **Step 1:** Generate two strings of parent chromosome randomly
- **Step 2:** Generate two crossover points randomly and cut each string of parent chromosome into three sections
- **Step 3:** Keep the 1st and 3rd chromosome sections and arrange the procedures at the middle part of one parent chromosome string according to the sequence of the same in the other parent chromosome string to form the offspring chromosome

It can be seen from Step 3 that the offspring chromosome is the feasible chromosome since the arrangement sequence of the offspring chromosome after the crossover is stemmed from the pre-crossover parent chromosome.

Then, carry out the variation operation of the chromosome by using the simulated annealing algorithm according to the above mentioned algorithm design. Here the inner circulation end condition is that the iteration shall reach certain times while the end condition for the outer circulation is that the designed minimum temperature shall be met. Thus, going through the above steps, new groups will replace the old ones.

CASE STUDY

Take the core part of the assembly line of the passenger car engine of company B for example. There are totally 5 work stations requiring totally 38 procedures. Their sequence order, procedure time and assigned equipment quantities are shown in Fig. 3.

In Fig. 3, the digits in the circle represents the procedure number and the digits above the circles indicate the procedure time and those colored circles means that equipments are required to finish such procedures.

Use GASA algorithm to solve under the Matlab environment and the program running results are shown in Fig. 4 while Fig. 5 is the iteration map:

The digits at the left side of each column of Fig. 4 are the procedure number while the digits at the right pat indicate the procedure time. The process sequence is from the bottom to the top. The horizontal axis of Fig. 5 is the iteration times and the vertical axis indicates the each iteration optimum value. Figure 5 proves the good convergence of this algorithm.

The solution results by using the traditional genetic algorithm are shown in Fig. 6.

According to the above running results map, the results of the traditional GA and the GASA algorithms are shown in Table 1.

It can be seen from the "station time consumption" column of Table 1 that the cycle of

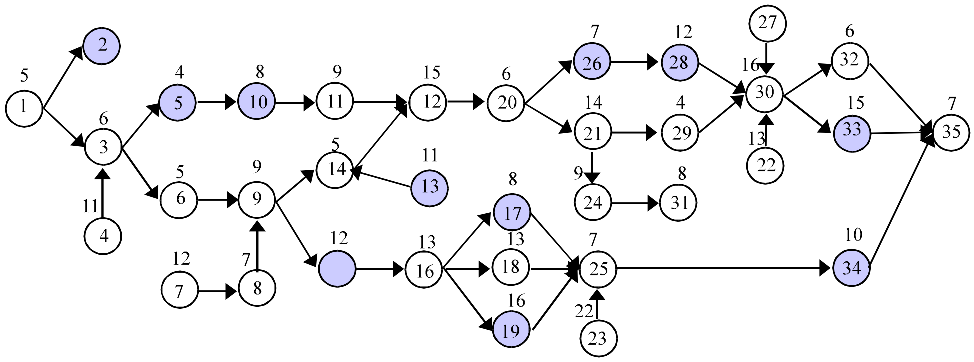

Fig.3: Map of processing procedures

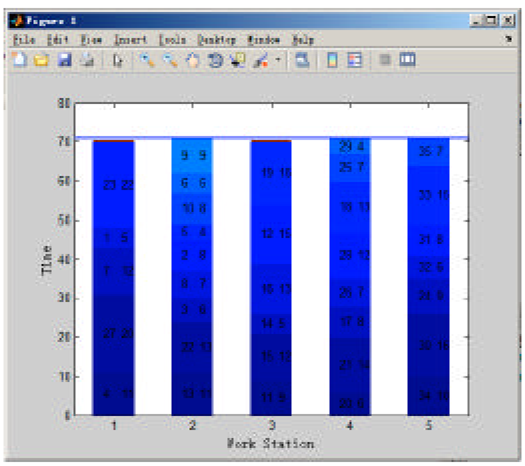

Fig. 4: Running results of GASA algorithm

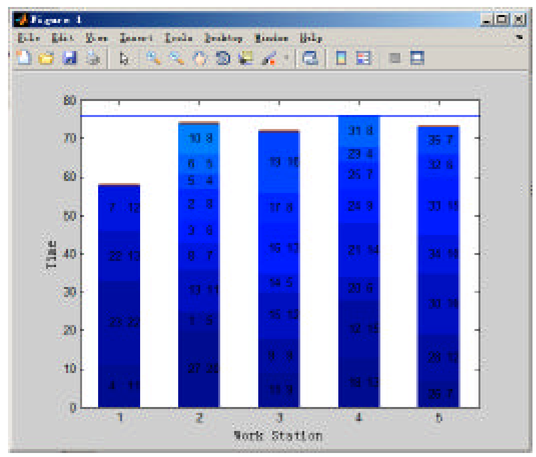

Fig. 6: Running results of traditional genetic algorithm

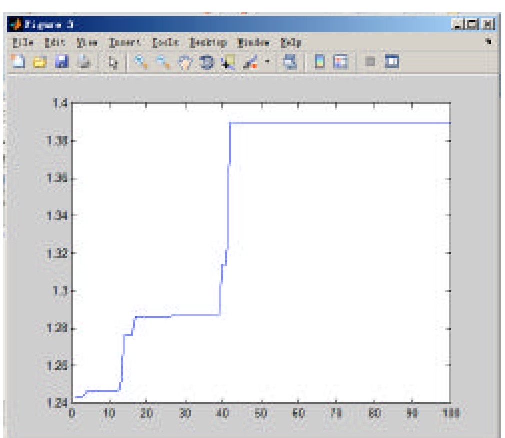

Fig. 5: GASA optimum solution iteration map

GASA is 76-71 = 5seconds less than the traditional GA and thus the capacity is enhanced greatly; it can be found from the comparison between Fig. 4 and 6 that the station load of GASA is better balanced. Therefore, GASA can obtain better solution and realize the process optimization better.

Table 1: Case results

Algorithms	Sta-tion	Sequence	Cycle time
GASA	1	4→27→7→1→23	70
	2	13→22→3→8→2→5→10→6→9	71
	3	11→15→14→16→12→19	70
	4	20→21→17→26→28→18→25→29	71
	5	34→30→24→32→31→33→35	71
Traditional GA	1	4→23→22→7	58
	2	27→1→13→8→3→2→5→6→10	74
	3	11→9→15→14→16→17→19	72
	4	18→12→20→21→24→25→29→31	76
	5	26→28→30→34→33→32→35	73

CONCLUSION

Considering that different objectives involved in the balance of the engine assembly line are always contradictory to each other and the optimization to one objective will then results in the degradation of other objectives, this paper builds the multi-objective process optimization mathematical model for the assembly line of passenger cars by taking the capacity, balance rate and cost into consideration and later uses the GASA algorithm for solution. Actual case proves that this model and related algorithm can optimize the process of the large-scale assembly line of passenger car engines in certain degree and thus enhance the comprehensive performance of the process and the performance of the algorithm is better than the traditional genetic algorithm.

REFERENCES

Chica, M., O. Cordon, S. Damas and J. Bautista, 2012. Multiobjective memetic algorithms for time and space assembly line balancing. Eng. Appli. Artificial Intell., 25: 254-273.

Douma, A., P. Schuur and J. Schutten, 2011. Aligning barge and terminal operations using service-time profiles. Flexible Ser. Manuf. J., 23: 385-421.

Erel, E. and S.C. Sarin, 1998. A survey of the assembly line balancing procedures. Prod. Plann. Control, 9: 414-434.

Fan, W.B., J. Zhou and Z.l. Xu, 2010. Appication of genetic algorithm to assembly line balancing. Comput. Technol. Dev., 2: 194-196.

Fattahi, P., N.B. Tavakoli, M. Fathollah, A. Roshani and M. Salehi, 2012. Sequencing mixed-model assembly lines by considering feeding lines. Inter. J. Adv. Manuf. Technol., 61: 677-690.

Gurevsky, E., O. Battaia and A. Dolgui, 2013. Stability measure for a generalized assembly line balancing problem. Dis. Applied Math., 161: 377-394.

Klein, R. and A. Scholl, 1996. Maximizing the production rate in simple assembly line balancing-a branch and bound procedure. Eur. J. Oper. Res., 91: 363-385.

Jian-Sha, L., J. Ling-ling and L. Xiu-Lin, 2010. Hybrid particle swarm optimization algorithm for assembly line balancing problem. China Mech. Eng., 21: 420-424.

Mao, L.X. and Y.Q. Zheng, 2010. Ant colony algorithm for solving the first category of assembly line balancing problem. Comput. Syst. Appli., 19: 140-143.

Mutlu, O., O. Polat and A.A. Supciller, 2013. An iterative genetic algorithm for the assembly line worker assignment and balancing problem of type-II. Comput. Oper. Res., 40: 418-426

Peng, H., K.L. Xu and Z.H. Si, 2011. Multi-objective optimization for mixed-model assembly line balancing problem based on GA. Mod. Manuf. Eng., 11: 49-53.

Suresh, G. and S. Sahu, 1994. Stochastic assembly line balancing using simulated annealing. Inter. J. Prod. Res., 32: 1801-1810.

Wang, H.B., 2012. Study on the energy consumption balance WSN routing protocol based on improved genetic algorithm. Inter. J. Adv. Comput. Technol., 4: 460-467.

Wang, S.T., M.L. Zheng, Y.H. Fan and Z. Li, 2012. Cutting parameters optimization in machining thin-walled characteristics of aircraft engine architecture based on machining deformation. Inter. J. Adv. Comput. Technol., 4: 244-252.

Wang, Y.F., M.Z. Liu and B.Z. Yu, 2005. Hybrid genetic algorithm for solving the problem of balancing an assembly line. J. Hefei Univ. Technol., 6: 616-619

Xiao, Z.H., 2010. Research in auto mobile assembly line balancing problem based on improved genetic algorithm. Master's Thesis, Wuhan University of Science and Technology.