Research on Evolution Balancing for Product Family Assembly Line in Big Data Environment

Yongming Wu[1,2], Yanxia Xu[1], Lifei Luo[1] ,Han Zhang[1], Xudong Zhao[1]
[1]Modern Manufacturing Key Laboratory of Guizhou University, Guizhou Guiyang 550025, China. E-mail: wu20811055@163.com
[2]Public Big Data Key Laboratory of Guizhou Province, Guizhou, Guiyang, 550025, China.

Aiming at the problem of product family assembly line (PFAL) evolution balancing, an evolution balancing model for PFAL is established and an improved algorithm based on NSGA_II is also proposed. Firstly, the product family evolution and assembly line characteristics are researched and analyzed in big data environment. Tasks on PFAL are divided into platform and personality tasks, and the stability of assembly tasks is mainly considered especially. In the optimization process, a chromosome encoding based on TOP sorting algorithm is adopted, and a new density selection and decoding algorithm is proposed to make up for the deficiencies in traditional algorithms. Finally, an example of PFAL planning is given to verify the effectiveness and feasibility of the improved NSGA_II.

Keywords: Big Date, Product Family Assembly Line, Evolution Analyzing, Balancing Optimization

1 Introduction

In Mass Customization (MC) mode, developing a high-quality, high-efficiency, low-cost and environment-friendly product platform design and assembly manufacturing system are challenges and must be solved for MC enterprises [1]. With the further research and application of product family technology, PFAL needs to meet the product diversity, assembly technology and equipment update and other dynamic changes, and constantly adjusting, or even re-planning, namely assembly line evolution problem. On the other hand, in order to improve the assembly efficiency, the assembly line balancing problem should also be considered. Therefore, in MC mode, PFAL needs to face the problems with evolution and balancing at the same time, which is called assembly line evolution balancing (dynamic balancing) .

Nowadays, how to implement dynamic balancing for product requirements, design and assembly technology and enterprise resources change is a difficult problem in the product family strategy [2]. In big data environment, the data ,too, including manufacturing data, is featured by huge scale (Volume), various types (Variety) , low value density (Value) and fast processing speed (Velocity). The revolution of big data age is not restricted to the technical level, namely a whole new way of thinking. The strategic significance of big data technology lies not in huge data information, but in the specialization of these meaningful data. Therefore, the ability to process unstructured and semi-structured data in manufacturing system needs to be improved. Simultaneously, the data value is finally realized through data mining, and assembly line evolution balancing should be studied from the data-driven perspective.

2 Literature review

2.1 The application of big data analysis in manufacturing system

With the violent development of technologies such as the internet, internet of things, cloud computing and big data, the rapid growth of data scale and the high complexity of data models have become severe tests and valuable opportunities for many industries. How to quickly and effectively collect, process, and analyze large-scale and diversified data has become a hot issue in recent scientific research and practical application [3]. In terms of data flowing management, data flowing processing technology makes it possible to obtain approximate query results in real-time and efficiently by constantly updating the structure of a representative data, which set in the memory space [4,5]. However, the evolution clustering problem mainly focuses on the clustering accuracy and ignores the real-time. In the process of studying data flowing mining, initially assuming that the data is evenly distributed, and focusing on how to solve the problem of big data samples in the data flowing. Many researchers have proposed a classification technique to solve the problem of conceptual drift in the data flowing [6]. Data mining research is still in its infancy, and has a huge space for development. Currently, many researches mainly focus on data flowing clustering, classifying, and frequent pattern mining, literature manage anonymous datasets [7], which provided a comprehensive conceptual basis for the privacy protection in big data, also to coordinate the system performance.

2.2 Product family evolution in MC mode

As the main implementation methods and means of MC, technologies such as modularization, product family and product platform are generally concerned and adopted by enterprises, in particular, are regarded as an important means to enhance the enterprises core capabilities. Product family is a series of derivative products, which can share general technology and be located in the related market, especially, can add or delete different personalized modules based on product platform to meet different special characteristics and personalized functional needs [8]. Therefore, the contradiction balancing process in the product family optimization often appears as an evolution balancing process that changes with the running time history.

Product family is a series of derivative products, which can add or delete different personalized modules based on product platform to meet different functional needs [9]. Product family and product platform are two different concepts but are related to each other. The development of product family is based on product platform and is an important way to achieve mass customization design [10]. Simpson et al [11] proposed establishing a product conceptual platform through similarity analysis between functional and conceptual structure and generalized design, and then set up an example platform through sensitivity analysis and generalized design of demand changes. Aiming at the master-slave architectures between platform parameters and personalized parameters in product platform architecture, as well as global and local design relationships between product platform and each product in platform, Simpson et al [12] established a master-slave multi-tier optimization design model for product platform architecture.

2.3 Research on PFAL evolution

PFAL is a mixed model assembly line (MMAL) that can assemble a group of similar products simultaneously [13], which has its own characteristics and balancing planning theories. Hou et al [14] studied assembly line manufacturing systems for MC, and proposed a PFAL balancing model and an optimization algorithm. The assembly line balancing can be divided into two categories: initial balancing and rebalancing. However, most of studies only focus on the initial balancing for the assembly line, while less on rebalancing. Many researchers point out the importance of assembly line rebalancing [15, 16]. In a diversified and personalized environment, manufacturing enterprises need to continuously introduce new products to meet ever-changing market demands, which is also the key to win the market, and this new products often have new functions and structures.

Currently, assembly line re-balancing needs to consider the production efficiency and adjustment cost at the same time, which is regard as the indirect balancing problem under job allocation constraint. Yang et al [17] studied the evolution balancing caused by the seasonal demand, in the process of assembly line rebalancing, mainly considered two costs: the cost of equipment movement and the labor training costs of task reallocation. Aiming at solving assembly line rebalancing caused by the random change of task time, Gamberini et al [16] studied the adjustment cost and assembly balancing efficiency. However, in the process of product evolution, not only the change of work time but also the assembly line balancing research needs to face various problems.

3 Data-driven evolution balancing model

Driven by customer demand, core technology and market competition, product family constantly to a higher level development, reflecting into the product family hierarchical model, all this is a change in functional requirements driven by core technology, which eventually leads to the products changes in physical structure, forces the corresponding physical module changes. And then gradually improves the product platform or module, updates and the continuous improvement. The relationship between dynamic product demand and assembly task planning is shown in Fig. 1.

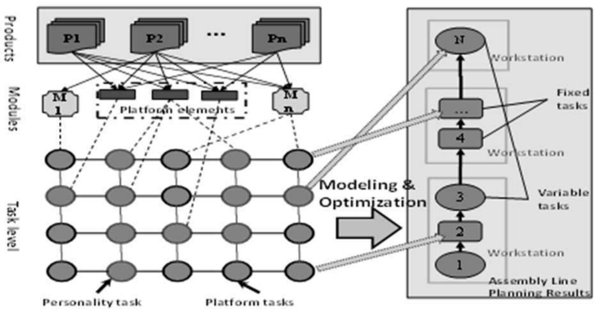

Fig. 1 *Dynamic product demand and assembly task planning*

The PFAL balancing goal is to find a plan for optimal task distribution to workstations, to balance the load among the workstations while meeting the product order, to minimize idle time at each workstation within *CT*. The mathematical model related parameters are as follows.

T_{total} : The total processing time; S : Number of workstations divided, where $k = 1,2,\cdots S$; J: Number of products that can be assembled simultaneously in PFAL, where $j = 1,2,\cdots J$; N : Total number of tasks, where $i = 1,2,\cdots N$; D_j : Demand for the j-th product in the product family; CT: Cycle time; q_j : The proportion of the *J*-th product in the product family; t_{ij} : The task time of the j-th product task i in the product family;

$$Q_{ij} = \begin{cases} 1 & \text{if product } j \text{ has the task } i; \\ 0 & \text{otherwise.} \end{cases} \quad T_{ki} = \begin{cases} 1 & \text{if task } i \text{ assigned to station } k; \\ 0 & \text{otherwise.} \end{cases}$$

$$P_{mn} = \begin{cases} 1 & \text{if task } m \text{ and } n \text{ belong to the same type of task;} \\ 0 & \text{otherwise.} \end{cases}$$

$$C_{kmn} = \begin{cases} 1 & \text{if task } m \text{ and } n \text{ are distributed to the } k\text{-th workstation at the same time;} \\ 0 & \text{otherwise.} \end{cases}$$

In the balancing of this paper, the time required for task i can be calculated by Eq. 1.

$$t_i = \sum_{j=1}^{J} (t_{ij} \cdot q_j) / \sum_{j=1}^{J} (Q_{ij} \cdot q_j) \tag{1}$$

PFAL balancing goal is to keep the whole assembly process in a continuous and steady state. Therefore, PFAL balancing is a multi-objective optimization problem. In the optimal balancing process, there are three objectives in this paper can be expressed by Eq. 2, Eq. 3 and Eq. 4.

$$Obj_1 = \left(\frac{1}{S-1}\sum_{k=1}^{S}\left(\sum_{j=1}^{J} D_j . \sum_{i=1}^{N} Q_{ij}.T_{ki}t_i - \overline{T}_s\right)^2\right)^{\frac{1}{2}} \quad (2)$$

Where, $\overline{T}_s = \sum_{k=1}^{S}\sum_{j=1}^{J}\sum_{i=1}^{N} q_j \cdot T_{ki} t_{ij}$ represents the average between workstations.

$$Obj_2 = \frac{1}{S}\sum_{k=1}^{S}\left(\frac{1}{J-1}\cdot\sum_{j=1}^{J}\left(D_j\cdot\sum_{i=1}^{N}Q_{ij}.T_{ki}t_i - \overline{T}_k\right)^2\right)^{\frac{1}{2}} \quad (3)$$

Where, $\overline{T}_k = \sum_{j=1}^{j}\sum_{i=1}^{N} q_j \cdot T_{ki}t_{ij}$ represents the average in the workstations.

$$Obj_3 = \left(\sum_{k=1}^{S}\sum_{m=1}^{N}\sum_{n=1}^{N} C_{kmn}\cdot p_{mn}\right)^{-1} \quad (4)$$

4 Improved multi-objective NSGA_II algorithm

The improved NSGA - II flowchart is shown in Fig. 2.

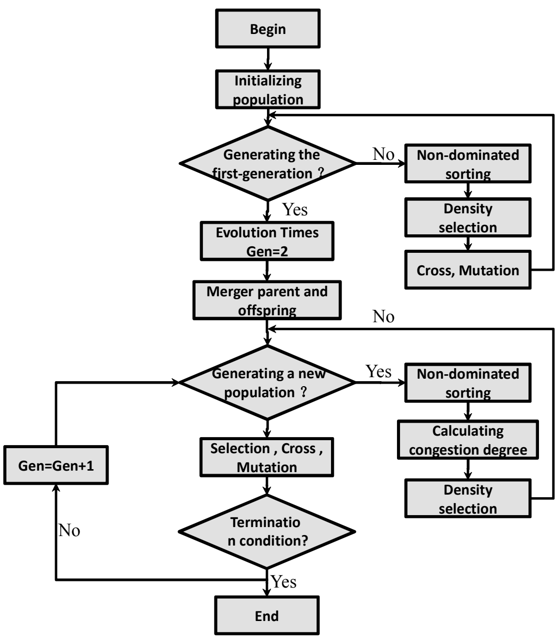

***Fig. 2** Improved NSGA - II Flowchart*

4.1 Encoding and decoding strategy

In order to make the generated chromosomes satisfy the precedence constraint relation between tasks, and prevent unreasonable chromosome generation, an improved topological sorting algorithm is used to encode all tasks to generate chromosomes, and eventually generate chromosomes that can satisfy the constraints. Detailed steps are illustrated as follows.

Step1: Initializing priority graph *G*, sets *S*= {1, 2, ... *N*}; Step2: Selecting job i with the in-degree of 0 in the set *S*, and writing it to the sorted set *I*; Step3: Deleting the job i from the set *S*, and subtracting 1 from the immediate successor to task i ; Step4: Determining whether the number of elements in the set *S* is 0, if so, then ending the algorithm; otherwise, returning to Step2.

4.2 Population initialization and fitness calculation

This paper takes the Eq. 2, 3 and 4 as the optimization goals, namely, in the process of assembly line balancing, minimizing the number of workstations, inter-station and in-station load index, and maximizing assembly line flexibility index.

4.3 Non-dominated sorting strategy

In order to avoid losing the population diversity, Pareto rank and density information are considered. The improved non-dominated sort (INDS) algorithm is as follows.

for each $p \in P$
$S_p = \phi; n_p = 0; P_{rank} = 1;$
for each $q \in P$
if $(p \prec q)$ then $S_p = S_p \cup \{q\};$
elseif $(q \prec p)$ then $n_p = n_p + 1;$
if $n_p = 0$ then
$p_\tau = 1;$
$F_1 = F_1 \cup \{p\};$
$i = 1$
while $F_i = \phi$
$F_{i+1} = \phi;$
for each $p \in F_i$
$n_q = n_q - 1; P_{rank} = P_{rank} + p_\tau;$
if $n_q = 0$ then
$p_\tau = p_\tau + 1; F_{i+1} = F_{i+1} \cup \{q\};$
$i = i + 1;$

4.4 Evolution operator

1) Selection strategy

If individuals p and q have the same Pareto ranking, the individual with lower density is selected, which will be helpful to increase population diversity. In particular, the chromosome with the best fit in the population can be replicated directly to the next generation without crossover and mutation, thus speeding the algorithm search.

2) Crossover strategy

Crossover operation plays a key role in the genetic evolution process, which can make the best individuals in the parental generation to get inherited and reserved as far as possible in the offspring. A single point crossover way is adopted in this paper, which is shown in Fig. 3. The parental chromosomes are divided into two parts by a randomly generated point, and the genetic information is exchanged after the cross position. In order to ensure the legitimacy of cross-offspring chromosomes, the exchanged sequence 6, 9, 8 and 10 is found in parent 1. And then, searching for the arrangement of genes 6, 9, 8 and 10 in parent chromosome 2, and which is used as new genes rearrangement in offspring 1, that is arranged as 8, 9, 6 and 10.

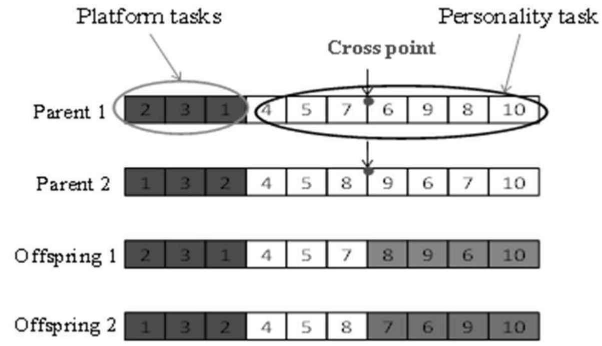

Fig. 3 *Chromosome mutation process*

3) Mutation strategy

Two mutation points are randomly generated. Then the gene fragment between the mutation points is also recombined, and the specific operations are as follows: the gene fragment between the variation points are deleted, and a feasible gene fragment is regenerated according to encoding strategy. So combining the gene fragments at both ends of mutation points with the regenerated feasible gene fragments together, the new chromosome is created. The specific mutation process is shown in Fig. 4.

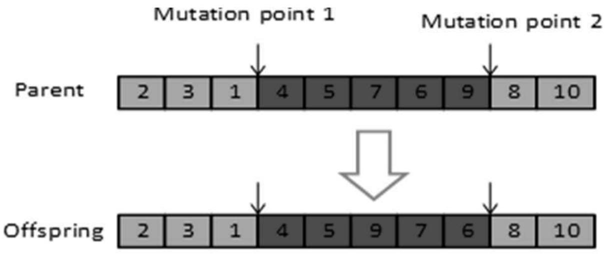

Fig. 4 *Chromosome mutation process*

5 Case study

5.1 Improving NSGA-II to optimize PFAL

In this section, a small sized mechanical PFAL evolution balancing problem in an enterprise is taken as the research object, to verify the effectiveness and feasibility of evolution balancing model and improved NSGA_II proposed in this paper. The product family includes four series of products: VG216, VG218, VG220 and VG220T, which are assembled on the same production line. The detailed sales are shown in Fig. 5 and the number of tasks contained on PFAL is 60.

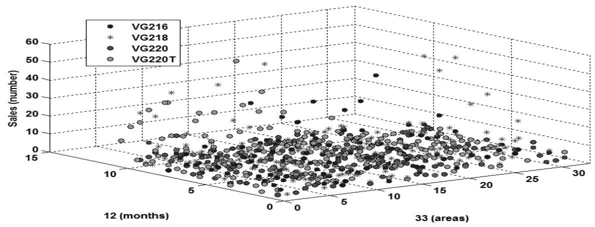

Fig. 5 *Sales diagrams of four products*

5.2 Algorithm performance and optimization results analysis

According to the improved NSGA_II algorithm, PFAL evolution planning is optimized and solved, in the process of optimization, population size *Pop*=100, crossover probability *Pc*=0.6, evolution generation *Gen_num*=100. PFAL of four kinds mechanical products are optimized by INSGA_II, to obtain the Pareto boundary as shown in Fig. 6. At the same time, according to the actual situation and Pareto boundary to determine the optimal evolution planning results are shown in Fig. 7.

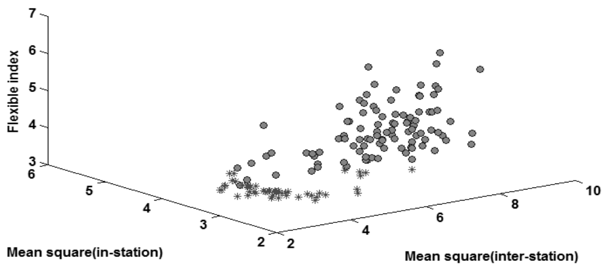

Fig. 6 *Pareto boundary diagram*

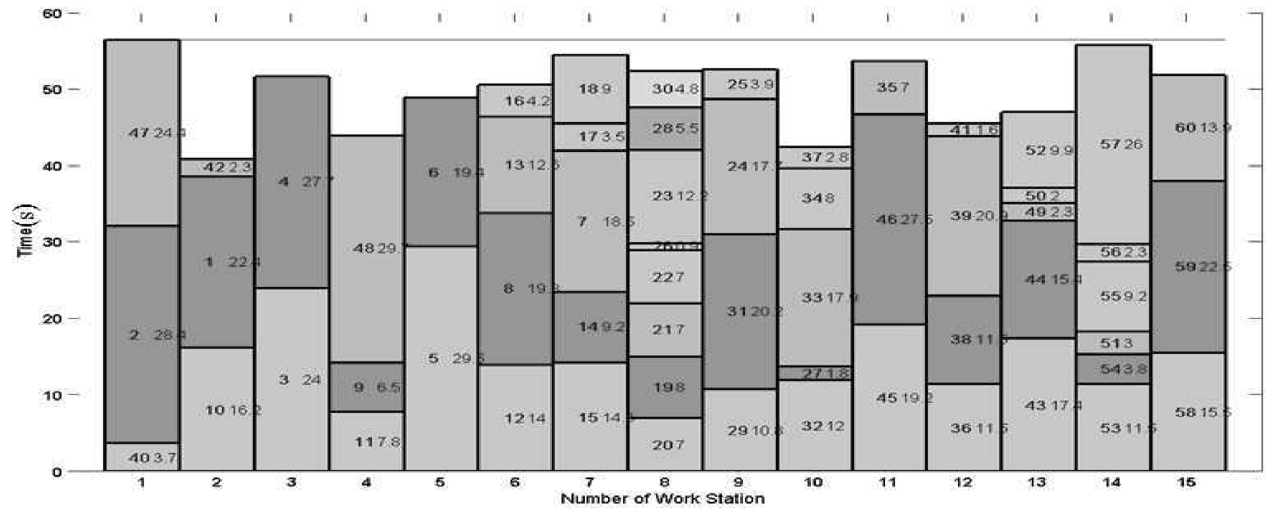

Fig. 7 *Station planning results*

In order to further prove the advancement for the INSGA_II algorithm in this paper, the traditional NSGA_II and multi-objective PSO algorithm are respectively selected for comparison, and the evolution planning results are shown in Tab. 1. More importantly, the ptimization times and performance of INSGA_II, which are shown Tab. 2 and Fig. 8, are significantly less than the other two algorithms, so INSGA_II proposed in this paper is more efficient.

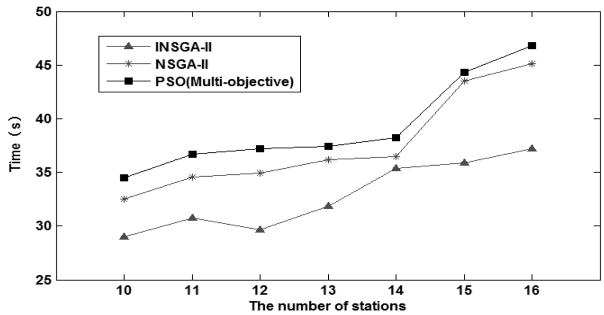

Fig. 8 *Performance comparison of three algorithms*

Tab. 1 *Evolution planning results of PFAL*

Algorithms	Product ratio	Distribution of workstations（*K*=15）（individual）
INSGA-II	2:4:2:1	(1,10,42);(2,40,47);(3,4);(5,6);(7,14,15,17,18);(8,12,13,16);(9,11,48);(19,20,21,22,23,26,28,30);(24,25,29,31);(27,32,33,34,37);(35,45,46);(36,38,39,41);(43,44,49,50,52);(51,53,54,55,56,57);(58,59,60)
NSGA-II	2:4:2:1	(1,2);(3,48);(4,10);(5,6);(7,19,20,24);(8,14,16,18,21,26);(9,11,40,42,47);(12,13,15,17);(22,23,29,30,31);(25,27,28,32,33,34);(35,36,37,38,39);(41,43,44,45);(46,49,50,51,52,54);(53,55,56,57);(58,59,60)
PSO multi-objective	2:4:2:1	(1,3,42)(2,4,7);(4,10,40);(5,6);(7,14,15,16);(8,12,13,17);(9,11,48);(18,19,20,21,24);(22,26,28,29,30,31);(23,25,32,33,34);(27,46,49,50,53,54)(35,36,37,38,39);(41,43,44,45);(51,52,55,56,57);(58,59,60)

Tab. 2 *Optimization performance analysis in the algorithm*

Algorithms	**INSGA-II**			**NSGA-II**			**PSO**		
Index	Obj_1	Obj_2	Obj_3	Obj_1	Obj_2	Obj_3	Obj_1	Obj_2	Obj_3
Mean	**3.102**	**2.795**	**3.372**	**3.630**	**3.132**	**3.397**	**3.274**	**2.782**	**3.545**
Std	**0.444**	**0.465**	**0.269**	**0.691**	**0.470**	**0.278**	**0.577**	**0.468**	**0.307**

6 Conclusions

Aiming at the evolution characteristics of PFAL, a topological method is adopted to generate the initial population that satisfies the assembly constraints, and an improved NSGA_II algorithm is also proposed. In the improved NSGA_II, a new encoding and decoding method is adopted to make up for the deficiencies of traditional decoding method, and the satisfactory results was obtained in the assembly line evolution balancing.The improved NSGA_II is applied to the actual assembly line evolution balancing, compared with the other two optimization algorithms, the results show that the evolution balancing model and the optimization method proposed in this paper can obtain a higher flexibility index.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (grant no. 51505094), the Guizhou Provincial Natural Science Foundation, China (no. [2016]1037), the Science and Technology Support Program in Guizhou (no. [2017]2029), Applied basic research program of major projects in Guizhou (JZ[2014] 2001) and talent introduction research program of Guizhou University（2014）60.

References

[1] FELFERNIG, A., & FELFERNIG, A. (2017). An introduction to personalization and mass customization. *Journal of Intelligent Information Systems*, Vol.49, no.2, pp.1-7. Elsevier Ltd, United Kingdom.

[2] RUSNÁKOVÁ S, ALEXANDER ČAPKA,FOJTL L.,MILAN ŽALUDEK, RUSNÁK V. (2016) Technology and mold design for production of hollow carbon composite parts. *Manufacturing Technology*, Vol. 16, No.4,pp.799-804. Univerzita J. E., Purkyne.

[3] BERNATIKOVÁ A.,NOVÁK P.(2016).Optimizing the production of porous alloys based on TiSi. *Manufacturing Technology*, Vol. 16, No.5,pp.871-874. Univerzita J. E., Purkyne.

[4] WERNER, M. (2017). Financial process mining - accounting data structure dependent control flow

inference. *International Journal of Accounting Information Systems*, Vol.25,pp. 57-80. Elsevier Ltd, United Kingdom.

[5] JAWERTH B, SWELDENS W.(2014). An overview of wavelet based multiresolution analyses. SIAM review, Vol.36, no.3,pp. 377-412. Elsevier Ltd, United Kingdom.

[6] HULTEN G, SPENCER L, DOMINGOS P. (2012). Mining time-changing data streams.*Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp.97-106. Univerzita J. E., Purkyne

[7] FUNG B, WANG K, CHEN R, et al. (2010). Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys (CSUR)*, Vol.42 ,no.4,pp. 14. Univerzita J. E., Purkyne

[8] TANG, M., QI, Y., & ZHANG, M. (2017). Impact of product modularity on mass customization capability: an exploratory study of contextual factors. International Journal of Information Technology & Decision Making, Vol.16.

[9] SEDLAK J,MALASEK J,ONDRA M,POLZER A.(2016). Development and production of prototype model of axial fan.*Manufacturing Technology*, Vol. 16, No.2,pp.436-444. Univerzita J. E., Purkyne.

[10] TSENG, M. M., & DU, X. (1998). Design by customers for mass customization products. CIRP Annals - *Manufacturing Technology*, Vol.47 ,no.1,pp. 103-106. Elsevier B.V., Netherlands.

[11] SIMPSON, T. W. (2004). Product platform design and customization: status and promise. *Artificial Intelligence for Engineering Design Analysis & Manufacturing*, Vol.18 ,no.1, pp.3-20. Elsevier B.V., Netherlands.

[12] SIMPSON, T. W., MAIER, J. R., & MISTREE, F. (2001). Product platform design: method and application. *Research in Engineering Design,* Vol.13, no.1,pp. 2-22. Elsevier B.V., Netherlands.

[13] BOYSEN N, FLIEDNER M, SCHOLL A. (2007). A classification of assembly line balancing problems. *European Journal of Operational Research*, Vol.183, no.2,pp. 674-693. Elsevier B.V., Netherlands.

[14] HOU LIANG,WU YONGMING, LAI RONGSHEN.(2014). An improved genetic algorithm for product family assembly line balancing. *Int J Adv Manuf Technol*, Vol. 70,pp.1775-1786. Elsevier B.V., Netherlands.

[15] WU Y.M., LIN S.Q., DAI L.Z.,LUO L.F.. (2017). Dynamic balancing for mixing assembly line based on PSO - GA cooperative optimization. *Manufacturing Technology*, Vol. 17, No.4,pp.622-627. Univerzita J. E., Purkyne.

[16] GAMBERINI R, GEBENNINI E, GRASSI A, REGATTIERI, A. A multiple single-pass heuristic algorithm solving the stochastic assembly line rebalancing problem. *International Journal of Production Research*, 2009, Vol.47,no.8,pp. 2141-2164. Elsevier Ltd, United Kingdom.

[17] YANG C, GAO J, SUN L. (2013). A multi-objective genetic algorithm for mixed-model assembly line rebalancing. *Computers & Industrial Engineering*, 2013, Vol. 65, no.1,pp. 109-116. Elsevier Ltd, United Kingdom.

10.21062/ujep/102.2018/a/1213-2489/MT/18/2/337