Arab J Sci Eng (2014) 39:5271–5277
DOI 10.1007/s13369-014-1052-4

RESEARCH ARTICLE - SYSTEMS ENGINEERING

A Real Multi-Objective Bin Packing Problem: A Case Study of an Engine Assembly Line

B. Naderi · Mehdi Yazdani

Received: 30 September 2012 / Accepted: 15 May 2013 / Published online: 16 April 2014

Abstract This paper studies a real case of bin packing problems. The problem is inspired from a car manufacturer that aims at improving logistic activities of its engine assembly line. It is to plan transportations of parts from a warehouse to workstations. The first objective is to find the minimum possible motorized vehicles to conduct all the transportations. The second objective is to smooth the workload of vehickles to the maximum extent possible. To tackle the problem, it is first formulated in form of a mixed integer linear programming model. Then, a local-based greedy heuristic is proposed to solve the problem in large-sized cases.

Keywords Multi-objective bin packing problem · Mixed integer linear programming model · Local search-based greedy heuristic

الخلاصة

تدرس هذه الورقة العلمية حالة حقيقية من مشاكل تعبئة الصناديق وتغليفها. وتهدف المشكلة المستوحاة من شركة مصنعة للسيارات إلى تحسين الأنشطة اللوجستية لخط تجميع محركها. إنها تخطيط عمليات النقل للأجزاء من المستودع إلى محطات العمل. والهدف الأول هو إيجاد الحد الأدنى من المركبات الآلية الممكنة لإجراء جميع عمليات النقل. والهدف الثاني هو تسهيل عبء العمل للرافعات الشوكية إلى أقصى حد ممكن. ولحل المشكلة، تتم في البداية صياغتها على شكل نموذج برمجة خطية صحيح مختلط. ومن ثم تُقترح طريقة تنقيبية محلية جشعة لحل المشكلة في الحالات ذات الحجم الكبير.

B. Naderi
Department of Industrial Engineering, Faculty of Engineering, University of Kharazmi, Karaj, Iran
e-mail: bahman_naderi62@yahoo.com

M. Yazdani (✉)
Department of Industrial Engineering, Islamic Azad University, Qazvin Branch, Qazvin, Iran
e-mail: m_yazdani@qiau.ac.ir

1 Introduction

A bin packing problem (BPP) is referred to as the problem in which objects of different volumes have to be packed into a finite number of bins with limited capacity [1,2]. The objective is to minimize the number of bins needed. Some other variations of BPP are two-dimensional packing, linear packing, packing by weight, packing by cost, and so on [3–5]. BPP belongs to a special class of combinatorial optimization problems, known as NP-hard problems [6,7]. Falkenauer [8] proposes a grouping genetic algorithm to solve clustering problems. Chan et al. [9] and Kang et al. [10] propose genetic algorithms to solve BPPs. Other metaheuristics are also applied, for example, simulated annealing by Dowsland [11], particle swarm optimization by Liu et al. [12] and variable neighborhood search by Hemmelmayr et al. [2].

BPPs have many applications, such as filling up containers, loading trucks with weight capacity, creating file backup in removable media and technology mapping in field-programmable gate array semiconductor chip design [13,14]. The problem treated is inspired by a real case study concerning Irankhodro Co., a well-known Iranian car manufacturer. Irankhodro Co. is essentially interested in improving logistics activities of its assembly lines.

Assembly lines are designed for flow-oriented production systems. They are mainly intended for continuous production of high-volume and low-variety commodities [15]. An assembly line consists of workstations arranged along a conveyor belt or a similar mechanical material handling equipment. The commodities are sequentially passed through the line and are moved from a workstation to the next. At each workstation, several tasks, typically installing parts to commodities, are repeatedly performed and their operation sequence or precedence relationship is predetermined [16–18].

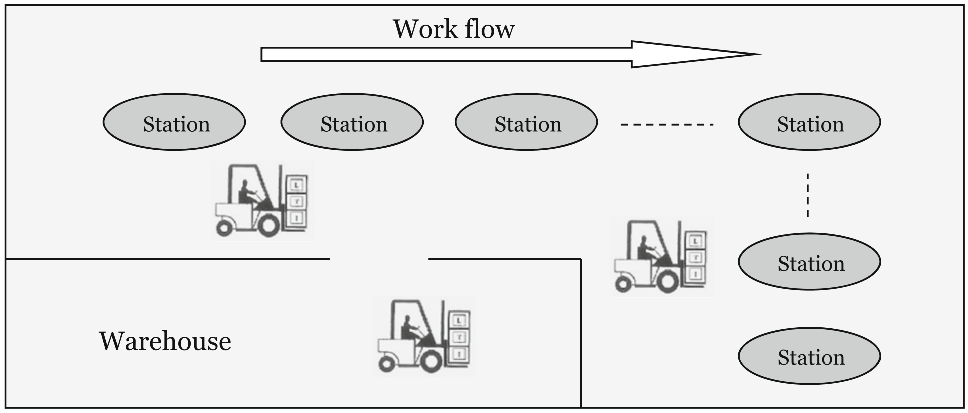

Fig. 1 The schematic view of the problem

Typically, a simple assembly line balancing problem can be stated as follows. Given a set of tasks, time requirement of each task, and precedence relationship of tasks, the problem is to assign the tasks to an ordered sequence of workstations while meeting the production requirement at minimum number of workstations or minimum cycle time [19].

The problem under consideration is inspired from assembly line of Irankhodro Co.; yet it is not actually the classical assembly line balancing itself. It is necessary to indicate that apart from transporters, like conveyors, needed for moving commodities themselves to next workstations, some other transporters are also required. These transporters carry the parts, assembled on commodities at each workstation, from the warehouse to each workstation. These transporters could be either manual trucks or motorized vehicles such as lift trucks.

In Irankhodro Co., the assembly line is already balanced and optimal assignment of tasks to each workstation is determined. Now, Irankhodro Co. aims at determining the minimum number of lift trucks required to transfer parts from the local warehouse of the assembly line to their corresponding workstations. After finding the minimum possible lift trucks to conduct all transportations, there is a second objective which is to smooth the workload of each used lift truck to maximum extent possible. In other words, the purpose is to make the work of each lift truck equal or nearly equal. Therefore, the case of Irankhodro Co. is a multi-objective problem. Figure 1 shows the schematic of the problem.

In its principles, this problem is more similar to BPP than assembly line balancing. To describe the problem in bin packing terms, we can state that each part transportation is an object and the transportation time can be considered as the volume of object. Each lift truck is like a bin and the total available time of a lift truck is the capacity of bin. There are some other technical constraints regarding other aspects of the problem discussed later. This paper first designs a mathematical model in the form of mixed integer linear programming model to deal with the problem. It then proposes an effective greedy heuristic algorithm to solve large-sized problems.

The rest of the paper is organized as follows. Section 2 formally defines and formulates the problem. Section 3 presents the proposed local search greedy heuristic. Section 4 solves the real case studied. Section 5 finally concludes the paper.

2 Problem Definition and Formulation

The problem under consideration is inspired from a real case study in Irankhodro industrial Co., an Iranian car manufacturer. In this problem, there are a set of n tasks each of which corresponds to transportation of one part from a local warehouse to a workstation. The time execution of task j (i.e., the transportation time of carrying a batch of part j to its corresponding workstation) is denoted by p_j where $j = 1, 2, \ldots, n$. There are a maximum number of m lift trucks available to operate the tasks. The daily production rate is D. The batch size and quantities of part j in the end product are b_j and r_j, respectively. Since there is no holding place in workstations, only one batch can be held close to the workstation. In this case, for each part j, one lift truck needs to operate task j a number of $D \cdot \frac{r_j}{b_j}$ times. Therefore, it daily takes

$$p_j \left(D \cdot \frac{r_j}{b_j} \right)$$

time units from one lift truck to carry part j. For example, assume production rate $(D) = 1{,}000$, batch size $(b_j) = 100$, quantities of part in the end product $(r_j) = 2$ and the transportation time of each batch from the warehouse to the workstation $(p_j) = 15$. For this part, $1{,}000 \cdot \frac{2}{100} = 20$ batches are daily needed. Since $p_j = 15$, the total transportation time of this part becomes 300 time units.

The total working time of a day is denoted by C time units while the allowance is A time units. This results in a total of $C - A$ daily available time units for each lift truck. In our case, we interpret this problem as a bi-objective case in lexicographic manner (more details could be found in Chankong and Haimes [20]). That is, the objectives are optimized sequentially, not simultaneously. The first objective is to find the minimum number of lift trucks. The second objective is to assign parts to lift trucks to carry parts from the local warehouse to workstations in such a way that the used lift trucks have the equal (or near equal) workload. To smooth the workload assigned to lift trucks, we decided to minimize the maximum workload difference among lift trucks.

The assignment has to be done in a way that one part is allocated to exactly one lift truck. Moreover, for ease of transportation and to avoid any interruption, disordering of the assignments is prevented. In other words, tasks are sequentially assigned to lift trucks. For example, consider three parts $j-1$, j and $j+1$. It is not feasible to assign parts $j-1$ and $j+1$ to one lift truck while assigning part j to another lift truck.

Before presenting the mathematical model, the established notations and parameters are presented below.

n Number of tasks (parts)
m Maximum number of available lift trucks
j Index for tasks
i, l Indices for lift trucks
D Production rate
p_j Execution time of activity j in minutes
b_j Batch size of activity
r_j Quantities of part j in end product
C Total available time in minutes
A Allowance time in minutes

The problem under consideration can be mathematically formulated as a mixed integer linear programming model. To formulate the problem, two binary variables are defined. The first one is used to show the lift truck assignment of tasks and the second one is to count the number of lift trucks used. These two binary variables are:

$X_{j,i}$ Binary variable taking value 1 if task j is carried out by lift truck i, and 0 otherwise
Y_i Binary variable taking value 1 if lift truck i is used, and 0 otherwise

Apart from the binary variables, the following continuous variable is also established.

W Continuous variables for measuring the maximum workload deviation

The mathematical formulation is as follows:

$$\text{Minimize } Z_1 = \sum_{i=1}^{m} Y_i \tag{1}$$

$$\text{Minimize } Z_2 = W \tag{2}$$

Subject to:

$$\sum_{i=1}^{m} X_{j,i} = 1 \quad \forall_j \tag{3}$$

$$X_{j,i} \leq X_{j+1,i} + X_{j+1,i+1} \quad \forall_{j<n,\, i<m} \tag{4}$$

$$X_{j,m} \leq X_{j+1,m} \quad \forall_{j<n} \tag{5}$$

$$\sum_{j=1}^{n} X_{j,i}\left(P_j \cdot \frac{D \cdot r_j}{b_j}\right) \leq C - A \quad \forall_i \tag{6}$$

$$\sum_{j=1}^{n} X_{j,i} \leq n \cdot Y_i \quad \forall_i \tag{7}$$

$$\sum_{j=1}^{n} X_{j,l}\left(P_j \cdot \frac{D \cdot r_j}{b_j}\right) - \sum_{j=1}^{n} X_{j,i}\left(P_j \cdot \frac{D \cdot r_j}{b_j}\right) - (C-A)(1-Y_i) \leq W \quad \forall_{i,l \neq i} \tag{8}$$

$$W \geq 0 \tag{9}$$

$$X_{j,i}, Y_i \in \{0,\ 1\} \tag{10}$$

Equations 1 and 2 present objective functions including the number of lift trucks used and the maximum workload difference among the lift trucks used, respectively. Constraint set 3 assures that each task j is assigned to exactly one lift truck. Constraint sets 4 and 5 together specify that the disordering of assignment has to be avoided. Constraint set 6 ensures that lift truck overloading does not occur; that is, the total time required by a lift truck to carry out the assigned task does not exceed the total available time. Constraint set 7 is used to indicate if each lift truck i is used or not. Constraint set 8 calculates the maximum workload difference of lift trucks. Constraint sets 9 and 10 define the decision variables.

Table 1 presents the number of binary variables and constraints needed by the model to formulate a problem with n tasks and m available lift trucks.

The proposed mixed integer linear programming model is bi-objective; therefore, multi-objective tools can be used. It is necessary to remind that the real case is going to be solved in lexicographic manner. After careful review of the model and the real case, it is found out that we can convert the multi-objective lexicographic model into a single-objective model by considering the following objective.

$$\text{Minimize} \sum_{i=1}^{m} Y_i + \left(\frac{1}{(C-A)}\right) W$$

where the first term corresponds to minimizing the number of lift trucks used and the second term refers to minimizing smoothing rate.

In fact, the model optimizes the objectives sequentially since the second term never exceeds one. Therefore, the pri-

Table 1 The number of binary variables and constraints

	Binary variables	Continuous variables	Constraints
Volume	$mn + m$	1	$2n + m^2 + nm$

ority of minimizing the number of lift trucks is always more than smoothing the workload.

3 The Proposed Heuristic Algorithm

Although the BPP is NP-hard [4], we found that the mathematical model of relatively large instances could be solved to optimality by common specialized operations research software such as LINGO 10. To tackle even larger-sized problems, an approximation is necessary. We therefore propose a simple yet effective heuristic algorithm. A quality solution could be generated through rules used in this algorithm.

The proposed heuristic algorithm first attempts to find the minimum number of lift trucks, then attempts to maximize the smoothing rate. To this end, we use the simple first fit algorithm [21] to determine the minimum number of lift trucks. The algorithm assigns tasks (starting from task 1, task 2 and so on) to lift truck 1 to extent possible. Then, the maximum possible tasks are assigned to lift truck 2. The procedure proceeds until all tasks are assigned to lift trucks. The number of lift trucks used is the approximation number of the minimum possible lift trucks needed to conduct tasks.

1	1	1	1	2	2	2	2	3	3

Fig. 2 The example for encoded solution

Next, the algorithm looks for smoothing the workload of lift trucks. The initial solution of this phase is the one found in previous phase. We use the following encoding scheme to represent a solution to the algorithm. The encoded solution is a string of lift truck numbers. For example, consider a problem with ten tasks where at least three lift trucks are needed to carry them out. Figure 2 presents one possible solution.

To improve the solution obtained by the first phase, we propose a local search-based heuristic with following steps:

Step 1: Take solution s and make $counter = 0$, and $best = W$

Step 2: Calculate the average workload of liftrucks ($\bar{I}$):

$$\bar{I} = \frac{\sum_{j=1}^{n}\left(P_j \cdot \frac{D \cdot r_j}{b_j}\right)}{Q}$$

where Q is the minimum number of liftrucks found in previous phase.

Step 3: Calculate the workload of each liftruck (I_i):

Step 4: Determine the liftruck k with maximum workload deviation ($\max_i\{|I_i - \bar{I}|\}$).

Step 5: Reassign the tasks assigned to liftruck k and its adjacent liftrucks.

If we have $I_i > \bar{I}$, two new solutions (s' and S'') are generated by assigning the first task of liftruck k to liftruck $k-1$ and the last task of liftruck k to liftruck $k+1$.

If we have $I_i < \bar{I}$, two new solutions (s' and S'') are generated by assigning the last task of liftruck $k-1$ to liftruck k and the first task of liftruck $k+1$ to liftruck k.

Step 5: Accept the solution with lower maximum workload deviation (W) among two solutions s' and S''

$$W = \max_{i<l}\{|I_i - I_l|\}$$

Step 6: If $W < best$, set $counter = 0$, and $best = W$; otherwise, increase $counter$ by one unit.

Step 7: If $W = 0$ or $counter > 20$, terminate the procedure; otherwise, go to Step 2.

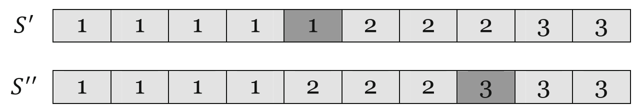

Fig. 3 Two new solutions generated by the operator

To better clarify the operator used in Step 4, consider the solution presented in Fig. 2, and suppose the lift truck with maximum workload deviation is lift truck 2 and its workload is greater than the average. Therefore, two new solutions are generated as shown in Fig. 3. The W of these two solutions is calculated and the solution with lower W is accepted.

4 Numerical Experiments

This section evaluates the performance of the mixed integer linear programming model and the proposed local search-based algorithm. First, we use the data taken from the real case in Irankhodro Company. Later, we generate a set of experimental instances to further evaluate the performance.

4.1 The Results on the Case Study: Irankhordo Co.

The case studied in this paper is taken from engine assembly line of Peugeot 206. In this assembly line, there are 72 tasks and a maximum of four lift trucks available. The working time per shift is 480 min while the allowance time is 30 min. The production rate is 240 per working shift. After obtaining values of b_j, r_j and p_j, the model is used to solve the real case to optimality. Tables 2, 3 and 4 show that frequency of b_j and r_j for 264 parts, respectively. Cumulative frequency is the total frequency below the given part batch. The relative frequency is the proportion of parts belonging to the given part batch.

We also ran the local search heuristic algorithm to evaluate its performance. The mathematical model and the heuristic algorithm are coded into LINGO 10 and Borland C++, respectively. The MILP model optimally solves the problem with computational time of 1,047 s. The optimal solution shows that three lift trucks can carry out the tasks. Regarding the second objective, the optimal solution has a very small maximum workload deviation (2.8 %). The heuristic algorithm finds also the minimum of three lift trucks needed to do tasks. It also assigns tasks with a very close to optimal maximum workload deviation (3.4 %), less than 1 % difference.

Table 2 The frequency of b_j

Part batch	Frequency	Cumulative frequency	Relative frequency (%)
1–100	2	2	3
101–200	5	7	7
201–300	10	17	14
301–400	8	25	11
400–500	11	36	15
500–600	29	65	40
>600	7	72	10
Total	72		100

Table 3 The frequency of r_j

r_j	Frequency	Cumulative frequency	Relative frequency (%)
1	53	53	74
2	10	63	14
3	2	65	3
4	7	72	10
Total	264		100

Table 4 The average RPD obtained by the solution methods

Instance		Model			Algorithm	
D	n	Average RPD	Time (s)	Optimality gap (%)	Objective	Time (s)
200	40	0.0	497	0	6.2	5
	70	3.6	5,000	10.2	2.4	4
	100	15.3	5,000	22.1	0.0	8
	150	–	5,000	–	0.0	15
300	40	0.0	5,000	0	4.6	4
	70	10.6	5,000	13.7	1.2	6
	100	18.3	5,000	25.4	0.0	9
	150	–	5,000	–	0.0	18
400	40	0.0	506	0	3.6	4
	70	8.7	2,106	9.6	1.5	9
	100	21.9	5,000	29.5	0.0	12
	150	–	5,000	–	0.0	17

4.2 Experimental Instances

The required data for a problem consist of n, m, p_j, b_j, r_j, D, C and A. We choose to use

$$n = \{40,\ 70,\ 100,\ 200\} \quad \text{and} \quad \mathrm{D} = \{200,\ 300,\ 400\}.$$

Parameter m is set to $\frac{5Dn}{10{,}000}$ lift trucks as the upper bound of lift trucks needed. Parameters p_j and r_j are generated from uniform distributions over (5, 20) and (1, 3), respectively. Parameter b_j of a task is randomly set to these numbers {50, 100, 150, 200}. For parameters C and A, we have $C - A = \{400\}$. Therefore, there are 12 combinations of

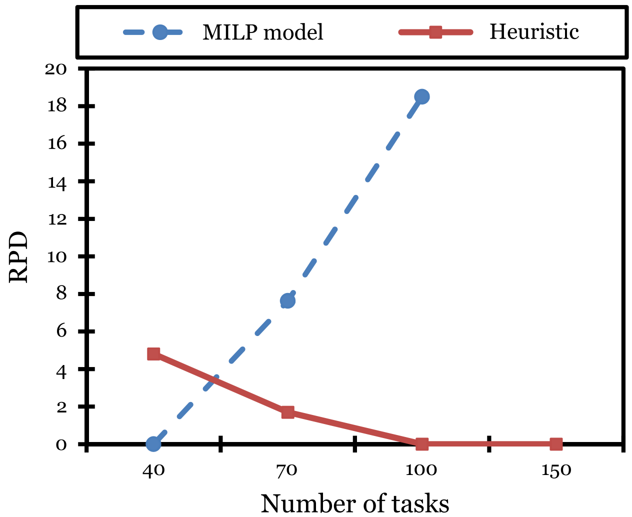

Fig. 4 Average RPD obtained by the algorithms in different sizes of the number of tasks

n and D. We generate five instances for each combination, summing up to 60 instances.

To compare the performance of the proposed heuristic and MILP model, we use relative percentage deviation (RPD) which is as follows.

$$\text{RPD} = \frac{\text{Alg}_x - \text{Min}}{\text{Min}}$$

where Min is the best solution obtained by algorithms in any of instances, and Alg_x is the solution obtained by algorithm x. The MILP model is capable of optimally solving 21 out of 60 instances. In the rest, the heuristics obtain better solutions. On average, the heuristic gains lower RPD than the MILP model.

Figure 4 shows the average RPD obtained by the heuristic and MILP model in different sizes of n. Regarding the computational time, the heuristic is much faster than MILP model. The average computational time of the heuristic is 9 s while the MILP model is given a computational time limit of 5,000 s. To statistically compare the methods, we use one-way analysis of variance (ANOVA) test where the type of method is the single factor. The results show that there is a statistically difference between the performance of methods with p value very close to zero (p value = 0.007). The MILP model is statistically more effective than the heuristic in small-sized instances. While in larger sizes, the heuristic statistically outperforms the MILP model.

5 Conclusion

A real case of BPPs inspired from an Iranian car manufacturer was studied. The problem was to investigate logistic activities of its assembly line. There were some workstations in which parts were assembled to semi-finished products. These parts first had to be transported to workstations by lift trucks. The objective was to find the minimum possible lift trucks to conduct all transportations as well as smoothing the workload of lift trucks. The problem was formulated through a mixed integer linear programming model. To solve the problem in large-sized cases, a local-based heuristic was proposed. In this algorithm, first the minimum number of lift trucks is determined, then, the workloads smoothed. The model and algorithm were evaluated by solving the real data taken from the company.

As a future research direction, it is interesting to adapt and develop other novel heuristics for the problem under consideration. It can be also interesting to consider the case of bi-objective BPP to simultaneously minimize the minimum number of bins and workload variance among the bins.

Acknowledgments We greatly thank two anonymous referees for their constructive comments on this manuscript and all the individuals who sacrificed their time to be interviewed as part of this study.

References

1. Chen, B.; Srivastava, B.: An improved lower bound for the bin packing problem. Discret. Appl. Math. **66**(1), 81–94 (1996)
2. Hemmelmayr, V.; Schmid, V.; Blum, C.: Variable neighbourhood search for the variable sized bin packing problem, Comput. Oper. Res. **39**(5), 1097–1108 (2012)
3. Haouari, M.; Serairi, M.: Heuristics for the variable sized bin-packing problem. Comput. Oper. Res. **36**(10), 2877–2884 (2009)
4. Khanafer, A.; Clautiaux, F.; Talbi, E.G.: New lower bounds for bin packing problems with conflicts. Eur. J. Oper. Res. **206**(2), 281–288 (2010)
5. Serairi, M.; Haouari, M.: A computational study of lower bounds for the two dimensional bin packing problem. Electron. Notes Discret. Math. **36**, 891–897 (2010)
6. Castro, P.M.; Oliveira, J.F.: Scheduling inspired models for two-dimensional packing problems. Eur. J. Oper. Res. **215**(1), 45–56 (2011)
7. Lewis, J.E.; Ragade, R.K.; Kumar, A.; Biles, W.E.: A distributed chromosome genetic algorithm for bin-packing. Robot. Comput. Integr. Manuf. **21**(4–5), 486–495 (2005)
8. Falkenauer, E.: Genetic Algorithms and Grouping Problems. John Wiley, New York (1998)
9. Chan, F.T.S.; Au, K.C.; Chan, L.Y.; Lau, T.L.: Using genetic algorithms to solve quality-related bin packing problem. Robot. Comput. Integr. Manuf. **23**(1), 71–81 (2007)
10. Kang, K.; Moon, I.; Wang, H.: A hybrid genetic algorithm with a new packing strategy for the three-dimensional bin packing problem. Appl. Math. Comput. **219**(3), 1287–1299 (2012)
11. Dowsland, K.A.: Some experiments with simulated annealing techniques for packing problems. Eur. J. Oper. Res. **68**(3), 389–399 (1993)
12. Liu, D.S.; Tan, K.C.; Huang, S.Y.; Goh, C.K.; Ho, W.K.: On solving multiobjective bin packing problems using evolutionary particle swarm optimization. Eur. J. Oper. Res. **190**(2), 357–382 (2008)

13. Mandal, C.A.; Chakrabarti, P.P.; Ghose, S.: Complexity of fragmentable object bin packing and an application. Comput. Math. Appl. **35**(11), 91–97 (1998)
14. Xavier, E.C.; Miyazawa, F.K.: The class constrained bin packing problem with applications to video-on-demand. Theor. Comput. Sci. **393** (1–3), 240–259 (2008)
15. Boysen, N.; Fliedner, M.; Scholl, A.:. A classification of assembly line balancing problems. Eur. J. Oper. Res. **183**(2), 674–693 (2007)
16. Becker, C.; Scholl, A.: A survey on problems and methods in generalized assembly line balancing. Eur. J. Oper. Res. **168**(3), 694–715 (2006)
17. Chen, J.C.; Chen, C.C.; Su, L.H.; Wu, H.B.; Sun, C.J.: Assembly line balancing in garment industry. Expert Syst. Appl. **39**(11), 10073–10081 (2012)
18. Toksarı, M.D.; İşleyen, S.K.; Güner, E.; Baykoç, Ö.F.: Assembly line balancing problem with deterioration tasks and learning effect. Expert Syst. Appl. **37**(2), 1223–1228 (2010)
19. Yagmahan, B.: Mixed-model assembly line balancing using a multi-objective ant colony optimization approach. Expert Syst. Appl. **38**(10), 12453–12461 (2011)
20. Chankong, V.; Haimes, Y.Y.: Multiobjective Decision Making: Theory and Methodology. Dover Publications, New York (2008)
21. Xia, B.; Tan, Z.: Tighter bounds of the first fit algorithm for the bin-packing problem. Discret. Appl. Math. **158**(15), 1668–1675 (2010)