MultiCraft

International Journal of Engineering, Science and Technology
Vol. 12, No. 3, 2020, pp. 90-107

INTERNATIONAL JOURNAL OF ENGINEERING, SCIENCE AND TECHNOLOGY

www.ijest1-ng.com
www.ajol.info/index.php/ijest
© 2020 MultiCraft Limited. All rights reserved

Multi-objective optimization of mixed model assembly line balancing in an assemble-to-order industry with stochastic environment

Ayele Legesse[1]*, Ermias Tesfaye[2] and Eshetie Berhan[3]

[1]Department of Mechanical Engineering, Haramaya Institute of Technology, Haramaya University, Haramaya, ETHIOPIA
[2]School of Mechanical and Industrial Engineering, Addis Ababa Institute of Technology, Addis Ababa University, Addis Ababa, ETHIOPIA
[3]School of Mechanical and Industrial Engineering, Addis Ababa Institute of Technology, Addis Ababa University, Addis Ababa, ETHIOPIA
*E-mails: Ayele Legesse[1] (kiyakontes@yahoo.com, *Corresponding Author), Ermias Tesfaye[2] (ermiastes@gmail.com), Eshetie Berhan[3] (berhan.eshetie@gmail.com)*
ORCID iD: https://orcid.org/0000-0002-7125-4994 (Berhan)

Abstract

The objective of this research is to propose a methodology for multi-objective optimization of a mixed-model assembly line balancing problem with the stochastic environment. To do this a mathematical model representing the problems at hand is developed with objectives of minimizing cycle time and minimization of the number of workstations (which is of Type-E ALB problem). And two optimization meta-heuristics are considered to solve it, namely, Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) and Multi-Objective Genetic Algorithm (MOGA). To test the performance of the algorithms three different size standard problems in Assemble-to-order types of industry are taken and five demand arrival scenarios are considered to incorporate the stochastic nature of the demand arrival for each model in all problems. Both the algorithms are coded and run using MATLAB® 2013a and are compared based on different performance measures. The results indicated that MOGA outperformed NSGA-II in most of the test problems. Nevertheless, both algorithms have resulted in significant improvements in the performance measures in Assemble-to-order types of industry dataset compared to the existing line configuration.

Keywords: Assembly Line, Multi-objective optimization, Single model, mixed-model, stochastic environment, Genetic Algorithm

DOI: http://dx.doi.org/10.4314/ijest.v12i3.9

Cite this article as:
Ayele Legesse, Ermias Tesfaye and Eshetie Berhan 2020. Multi-objective optimization of mixed model assembly line balancing in an assemble-to-order industry with stochastic environment. *International Journal of Engineering, Science and Technology*, Vol. 12, No. 3, pp. 90-107. doi: 10.4314/ijest.v12i3.9

Received: February 5, 2019; Accepted: August 31, 2019; Final acceptance in revised form: September 3, 2020

1. Introduction

Nowadays, due to the level of globalization, the ever-increasing competition, dynamic and uncertain global market with greater need for flexibility and responsiveness, the ability of a company to compete effectively is very crucial, which in return is influenced, to a large extent, by its capacity to produce an increased number of customer-based products in a timely manner (Samy and El-Maraghy, 2010). This requires the manufacturing companies to consider decisive strategic activity to design and optimize their own production systems as well as possible with a goal to control (innovate, improve, etc.) almost every process which can be managed to obtain an efficient production system ((Amouzgar, 2012; Albert, 2012; Abeya and Mulugeta, 2014; Pavel and Ulrych, 2012; Zacharia and Nearchou, 2012; Delice et al., 2017). In this context, the design of real-world manufacturing systems becomes more and more important. Particularly, the design of an efficient assembly line has a considerable industrial importance system (Baudin, 2002; Zacharia and Nearchou, 2012) as it can make the overall operations as effective as possible.

Most of the work related to the assembly lines (ALs) concentrate on the assembly line balancing (ALB) which deals with the allocation of the tasks among stations so that the precedence relations among them are not violated and a given objective function is optimized (Wenqiang *et al*., 2014; Li-Man, 2014). Optimization of assembly lines encompasses one or more predetermined objective(s). However, in real environment of assembly lines, two or more optimization objectives are significantly desired to be achieved simultaneously. A large part of real-world optimization problems are of multi-objective in nature (Naveen and Dalgobind, 2013). Most of the time, these objectives can be conflicting and compromised with each other (Saif *et al*., 2014, Manavizadeh *et al.*, 2012, Naveen and Dalgobind, 2013, Pavel and Ulrych, 2012, Lapierre *et al.*, 2006).

Previous research shows there is a tremendous increase in the researches in assembly line optimization in the past few years, especially in the years 2008 and onwards. Nonetheless, relative to the importance of including the aforementioned real-world assembly line problem occurrences, only a few researchers have considered incorporating them in their problem set-up (Zein-Eldin, 2014). Even among those studies considering multi-objective nature and randomness of task completion time of real-world mixed-model assembly line problems none of them has applied them in large scale problem instances. This paper deals with multi-objective optimization of mixed model assembly line balancing in an assemble-to-order industry with stochastic environment, in which assemble-to-order types of jobs with large scale assembly line balancing problem is considered.

2. Review of Literature

Assembly line balancing with a single objective with consideration of a line with a single model product in which the task times are deterministic was studied by several researchers (Neda *et al.*, 2012; Chica *et al.*, 2010). They introduced a technique of balancing assembly lines of Type-I problem with Tabu Search (TS) algorithm in which they have applied the method on a real industrial dataset after comparing it with the ones in the literature. Similar work in the same category was done in which Multi-Started Neighborhood Search Heuristic (MSNSH) method was applied in a real industrial dataset of a motorcycle manufacturer (Uddin and Lastra, 2011), which then was tested against the initial solution of the case company using simulation-based performance analysis using ARENA. Several other works of literature also dealt with such problems with different solution approaches like genetic algorithm (Aravelli, 2014), tabu search (Beyer and Deb, 2001), hybrid of genetic algorithm and tabu search (Kaveh and Laknejadi, 2011), ant colony algorithm (Bukchin and Rubinovitz, 2003), hybrid ant colony algorithm (Randy and Sue, 2004; Liu and Chen, 2002).

Liu *et al.* (2008) introduced a multi-objective ant colony optimization algorithm for the 1/3 Variant of the Time and Space Assembly Line Balancing Problem. Similar work in the same category was done by Bautista and Pereira (2002), in which they have worked on a solution procedure for type E simple assembly line balancing problem in which optimization objectives are minimization of the cycle time and the number of station. Blum *et al*. (2008) also presented a solution approach for assembly line balancing with Type-E problem in which they used a modified genetic algorithm (GA) for fuzzy assembly line balancing. Kara *et al*. (2014) presented a paper on assembly line balancing under uncertainty in which interval for operation times was assumed to be probabilistic and have proposed robust optimization models and exact solution methods to solve such problems. A similar effort in this category was made with other researchers to solve such problems using multiple single-pass heuristic algorithm (Sheu and Chen, 2008), bidirectional heuristic (Koltai *et al*., 2014). Chiang *et al.* (2012) introduced a Pareto based artificial bee colony algorithm for multi-objective single model assembly line balancing with uncertain task times. Several techniques have also been used by other researchers to come up with solutions for problems of such category, for instance, genetic algorithm (Wu *et al*, 2008) , and simulated annealing (Bukchin and Rubinovitz, 2003) are some of which hybridized the well-known methods to mitigate the shortcomings of the traditional versions of the methods resulting with such methods as hybrid simulated annealing algorithm Burcin (Liu and Chen, 2002), and hybrid multi-objective evolutionary algorithm (Nourmohammadi and Zandieh, 2011).

Previous researchers have also formulated a mathematical model and used a heuristic that minimizes the number of stations for predetermined cycle time. Similarly, other researchers also made effort to solve ALB problems in this category using different solution approaches as ant colony algorithm (Simaria and Vilarinho, 2004), genetic algorithm (Ponnambalam *et al*, 2000, Akpinar and Bayhan, 2014; Vilarinho and Simaria, 2006), simulated annealing (Fattahi *et al.*, 2011), hybrid genetic algorithm (Chica *et al*., 2010; Neda *et al*, 2012). Mixed-model assembly line balancing problems with single optimization objective, stochastic task time consideration, and straight type line configuration was presented by Sivasankaran and Shahabudeen (2013). In this paper, two kinds of robust criteria are provided; min-max related and α-worst scenario-based. They have designed a genetic algorithm-based robust optimization framework for each scenario. Similarly, (Bock, 2008) presented a paper on the same category in which overlapped and stopped operation was considered both under certainty (deterministic) and uncertainty (stochastic) environment in which they have developed mathematical model for solving the problem.

Assembly line balancing with a multi-objective and consideration of a line with mixed-model in which the task times are deterministic was studied by various researchers. Among the many others, Carlo-Colon and Nambiar (2008) presented a multi-objective optimization method for the mixed-model-line assembly line design problem. It was adopted based on the strength of Pareto Evolutionary Algorithm II (SPEA2) for the Mixed-Model Assembly Line balancing and equipment selection problem. Similar efforts were made with other researchers in this same area where solution methods such as ant colony optimization algorithm (Van Hop, 2004), genetic algorithm (Akpınar and Bayhan, 2011, Kucukkoc and Zhang, 2014; Su *et al.*, 2014, Kara and Tekin, 2009; Wenqiang and Mitsuo, 2011).

There are also other works which considered the hybrid solution methods such as ant colony optimization with genetic algorithm (Kara, 2008), hybrid genetic algorithm (Dong *et al.*, 2014), and those literatures in which two-sided version of mixed-model multi-objective optimization of ALB with deterministic task time and straight-line configuration was considered (Xu and Xiao, 2009; Vilarinho and Simaria, 2002; Vilarinho and Simaria, 2006; Uğur and Bilal, 2009). However, very few researchers considered the stochastic version of such ALB problems. Mixed-model assembly line balancing problems with multiple optimization objectives with consideration of stochastic task time and straight type line configuration was presented in Saif *et al.* (2014), Neda *et al.* (2012) and Yabo *et al.* (2014). Neda *et al.* (2012) presented a mixed-model assembly line balancing in the make-to-order and stochastic environment. The number of stations and cycle time being the two major conflicting objectives, the study introduced two additional balancing measures (horizontal and vertical balancing) which are minimized simultaneously. They have used multi-objective evolutionary algorithms and using the problem in the Make-to-Order environment as a testbed, they have compared five Multi-Objective Evolutionary Algorithms (MOEAs). Other solution methods for problems of this category such as ant techniques (Yabo *et al.*, 2014), modified ant colony optimization were also introduced.

Tables 1a and 1b elaborate the different categories of ALB problems presented by different researchers along with the experimental datasets considered. Moreover, one can see that very few of the publications incorporated real-world datasets and the problem category of MM_MO_St_S. This paper tries to present Multi-Objective Optimization of Mixed Model Assembly Line Balancing in an Assemble-to-Order Industry with Stochastic Environment which makes its category MM_MO_St_S, with both standard benchmark problems as well as real-world ALB problem datasets are utilized.

Table 1a. Categories of the single model ALB literatures and the respective problem types & the experimental dataset used

Category	Literature	Problem Type							Experimental Dataset					Category	Literature	Problem Type							Experimental Dataset				
		I	II	E	F	3	4	5	RW	HD	BM	RG	RB			I	II	E	F	3	4	5	RW	HD	BM	RG	RB
SM_SO_D_S	(Lapierre *et al*, 2006, Sheu and Chen, 2008, and Tamás *et al*, 2011)	x							x					SM_MO_D_S	(Chiang *et al*, 2012, Wu *et al*, 2008, Ponnambalam *et al*, 2000, Chica *et al*, 2010)	x									x		
					x				x						(Zein-Eldin, 2014)				x						x		
	(Blum, 2008, Liu *et al*, 2008)	x									x				(Bukchin and Rubinovitz, 2003)	x								x			
	(Kara *et al*., 2014)				x					x					(Nourmohammadi and Zandieh, 2011; Liu and Chen, 2002, Zacharia and Nearchou, 2012		x								x		
	(Bautista and Pereira, 2002)	x											x	SM_MO_D_U	(Rea *et al*., 2008, Armin *et al*, 2013)	x									x		
SM_SO_D_U	(Sachin and Prashant, 2014)				x						x			SM_MO_St_S	(Ullah et al, 2014 and Wenqiang *et al*., 2014)		x								x		
SM_SO_St_S	(Gamberini *et al* ,2009)				x								x		(Fattahi *et al*, 2011),	x									x		
SM_SO_St_U	(Adil and Lale, 2007)	x									x			SM_MO_St_U	(Bagher *et al*., 2011)	x									x		

Experimental Dataset: RW: Real World dataset, HD: Hypothetical Data, BM: Benchmark, RG: Randomly generated, RB: Real world dataset and Benchmark
Categorization: SM: Single Model, MM: Mixed Model, SO: Single Objective, MO: Multi-Objective, D: Deterministic Task Time, St: Stochastic Task Time, S: Straight Line Configuration, U: U-Type Line Configuration

Table 1b. Categories of the mixed model ALB literatures and the respective problem types & the experimental dataset used

Category	Literature	Problem Type							Experimental Dataset				
		I	II	E	F	3	4	5	RW	HD	BM	RG	HB
MM_SO_D_S	(Simaria and Vilarinho, 2004)		x								x		
	(Noorul Haq *et al*, 2006)	x								x			
	(Neda *et al.*, 2012)		x						x				
	(AlGeddawy and ElMaraghy, 2010)	x							x				
	(Akpinar and Bayhan, 2014)		x							x			
	(Bock, 2008)					x				x			
	(Sivasankaran and Shahabudeen,, 2013)	x								x			
	(Carlo-Colon and Nambiar, 2008)		x									x	
MM_SO_D_U	(Dong *et al*, 2014)		x								x		
	(Kara and Tekin, 2009)	x										x	
	(Kara, 2008)					x					x		
MM_SO_St_S	(Xu and Xiao, 2009)					x					x		
MM_SO_St_U	(Uğur *et al.*, 2011)					x					x		

Category	Literature	Problem Type							Experimental Dataset				
		I	II	E	F	3	4	5	RW	HD	BM	RG	RB
MM_MO_D_S	(Van Hop , 2004, Akpınar *et al.*, 2013; Vilarinho and Simaria, 2002, Vilarinho and Simaria, 2006,)	x									x		
	(Akpınar and Bayhan, 2011)	x								x			
	(Parisa *et al.*, 2018)	x					x		x				
	(Kucukkoc and Zhang, 2014)					x				x			
	(Su *et al.*, 2014)			x								x	
	(Li-Man, 2014)		x								x		
	(Kadir *et al.*, 2016)	x					x				x		
MM_MO_D_U	(Parames and Suchada, 2013)	x									x		
MM_MO_St_U	(Ullah *et al*, 2014)		x								x		

RW: Real World dataset, HD: Hypothetical Data, BM: Benchmark, RG: Randomly generated, RB: Real world dataset and Benchmark

Categorization: MM: Mixed Model, SO: Single Objective, MO: Multi-Objective, D: Deterministic Task Time, St: Stochastic Task Time, S: Straight Line Configuration, U: U-Type Line Configuration

3. Methodology

The required data in this study are collected from the case company and are of the secondary type, which is going to be used in the implementation and line performance analysis. Moreover, three different size standard problems are considered as a benchmark. A mathematical model for the considered ALB problem type is developed and it is coded using MATLAB® using two farmhouse optimization techniques; Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) and Multi-Objective Genetic Algorithm (MOGA). The two algorithms are run for the desired stopping criteria and are compared by using the line efficiency, line smoothness index, and the run time of the algorithm.

Line Efficiency (*E*): The line efficiency is the ratio between total station time to the product of cycle time and the number of workstations, represented as a percentage. It shows the percentage use of the line and is expressed as:

$$E = \frac{\sum_{i=1}^{k} ST_i}{k \times CT} \times 100 \quad (8)$$

where, k = total number of workstations, and CT = cycle time

Smoothness Index (SI): The smoothness index is an index for the relative smoothness of a given assembly line. A smoothness index of 0 indicates a perfect balance. A smaller SI results in a smoother line, thereby reducing the in-process inventory.

$$SI = \sqrt{\left(\sum_{i=1}^{k} \left(ST_{\max} - ST_i \right)^2 \right)} \quad (9)$$

where ST_{max} = maximum station time, and ST_i = Station time of station i.

Execution (CPU) Time: Execution time is the time it takes the algorithm to reach the stopping generation. It is considered by many researchers since it is directly tied to the efficiency of the algorithm selected.

Experimental Dataset

To test the performance of the algorithms three different size standard problems are adapted from www.assembly-line-balancing.de, a site presenting standard ALB problem setup for various types, size, and configurations, with their combined precedence diagrams as shown in Figure 1 through 3, and 4 for the data collected from the case company. The problems range from small to large scale ALB instances based on the categorization used in Akpınar *et al.* (2013). Table 1 shows the experimental dataset used and the problem-specific assumptions for each ALB problem.

Case Study (Large size): The required data for the case study problem are collected from Lifan Motors (Ethiopia) (Table 2). Lifan Motors (Ethiopia) is a company engaged in assembly and distribution of LIFAN vehicles in Ethiopia, owned by a subsidiary branch of Lifan Motors (China) called Yangfan Motors Plc. In this study production of two models of cars; L-530 and Mini Van are taken for the mixed-model assembly line optimization.

Table 2. Experimental datasets and problem-specific assumptions

	Problem Dataset			
	Mertens	**Lutz**	**Kilbridge**	**Case Study**
Number of Tasks	7	32	45	55
Size Category	Small	Medium	Large	Large
Problem specific assumptions				
Positive Zoning Restriction Between tasks	Task 1 and task 2	Task 30 and task 32	Task 40 and task 41	Task 37 and task 41
Negative Zoning Restriction Between tasks	Task 2 and task 7	Task 4 and task 16	Task 4 and task 9	Task 41 and task 55
The maximum and the minimum number of tasks per station respectively	2 and 4	4 and 11	5 and 11	7 and 14

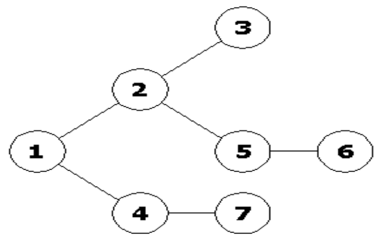


Figure 1.The combined precedence diagram of Mertens

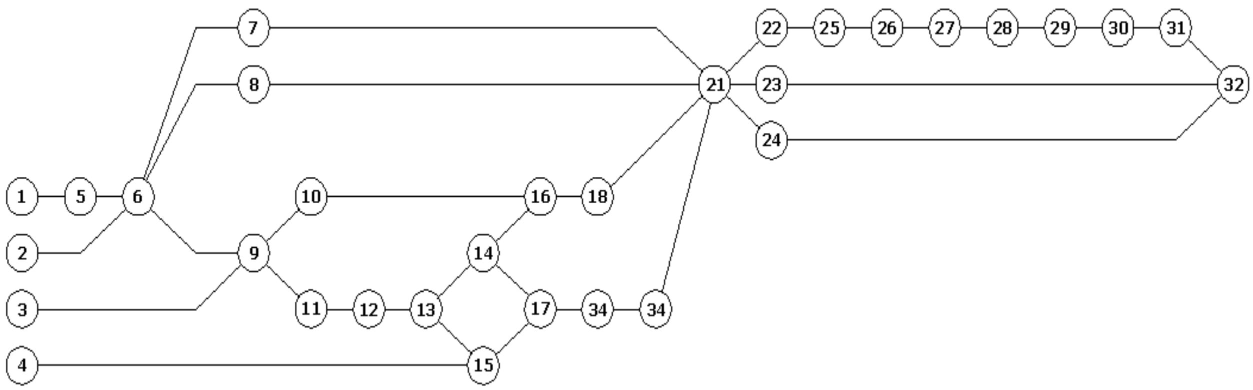


Figure 2. The Combined precedence diagram of Lutz

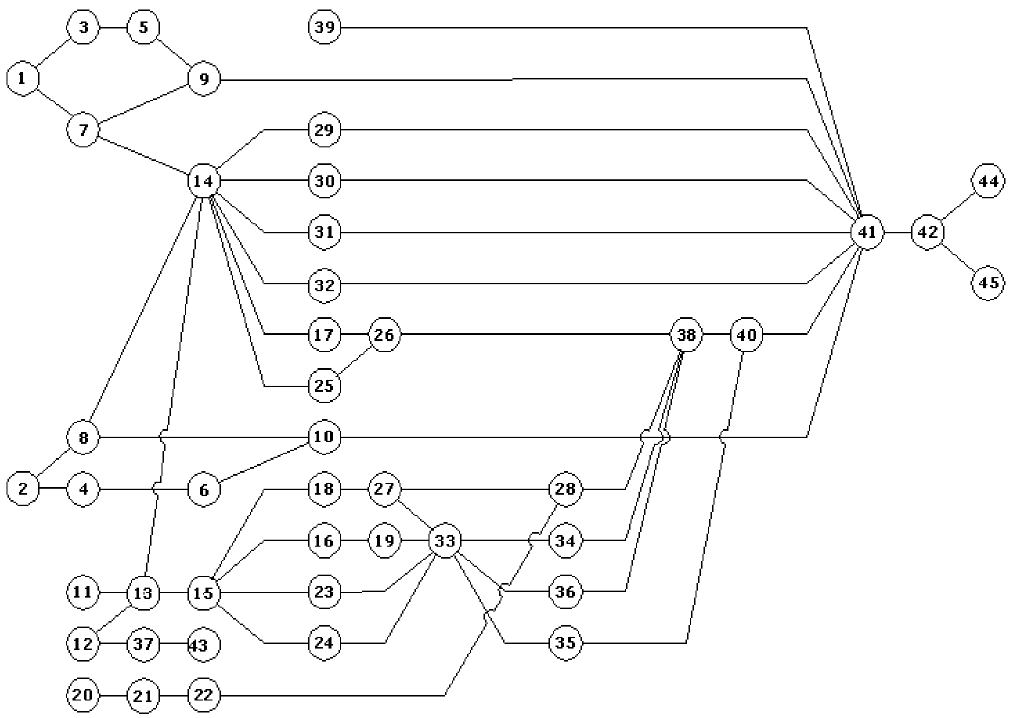


Figure 3. The combined precedence diagram of Kilbridge

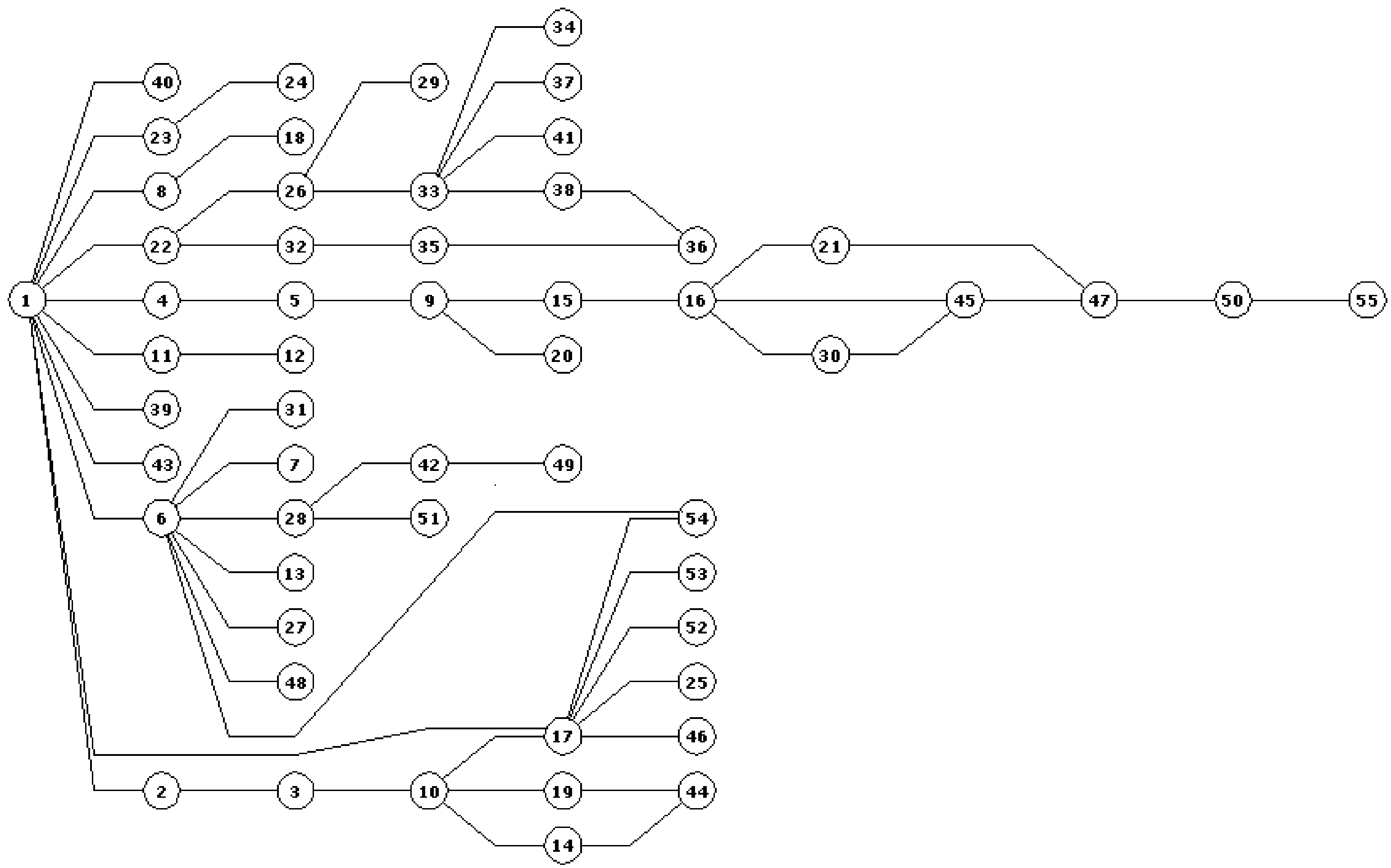


Figure 4. The combined precedence diagram of Case Company

4. Mathematical Model Formulation

4.1. Assumptions

- The precedence relationships among tasks for each model are known and the precedence diagrams for all the models can be combined such that the resulting diagram contains the N tasks.
- The assignment of tasks to a specific workstation can be forced or forbidden through the definition of positive or negative zoning restrictions.
- The setup time of tasks is included in their task times.
- The same tasks of different models could have different task completion times due to different assembly procedure requirements as a result of model variety.
- The time required to perform task *j* is stochastic, and it has a distribution with mean t_j and standard deviation σ_j.
- Each task *j* is assigned to only one workstation *v*, processed once, and only one task is allowed to process on a single station at a time. Common tasks could be assigned to different stations whereas some tasks, according to some considerations (e.g. specific equipment requirements), should be assigned to one station.
- The assembly line is a straight line and has a serial layout.
- Task times are independent of the sequences.

4.2. Notations

j	Index of task, $j = 1,2, \ldots, n$
l	index of product, $l = 1, 2, \ldots, m$
v	index of workstation, $v = 1, 2, \ldots, k$
t_{jlv}	Performance time of task *j* of product *l, on workstation v*
x_{jlv}	if task *j* of model *l* is assigned to workstation *v*, $x_{jlv} = 1$; otherwise, $x_{jlv} = 0$
IPj	Immediate Predecessor of task *j*
ZP	Tasks with Positive zoning restriction
ZN	Tasks with Negative zoning restriction
C_T	Total cycle time of mixed-model
α_l	expected demand ratio of model *l* during the planning period

4.3 Mathematical Model

$$OF1: \min Z1 = \sum_{v=1}^{k} \sum_{l=1}^{m} \sum_{j=1}^{n} v x_{jlv} \quad (1)$$

$$OF2: \min Z2 = \sum_{l=1}^{m} \left\{ \alpha_1 \max \left\{ \sum_{j=1}^{n} t_{jlv} x_{jlv} \right\} \right\} \quad (2)$$

Subject to:

$$\sum_{v=1}^{k} x_{jlv} = 1 \text{ where } j = 1, \ldots, \text{n, and } l = 1, \ldots, \text{m;} \quad (3)$$

$$x_{jlv} \le \sum_{v=1}^{k} v x_{olv} \quad \text{where } l = 1, \ldots, \text{m, and } j = 1, \ldots, \text{n } \forall o \in IP_j \quad (4)$$

$$\sum_{v=1}^{k} x_{jlv} - \sum_{v=1}^{k} x_{qpv} = 0, \forall (qp, jl) \in ZP \quad (5)$$

$$x_{jlv} - x_{qpv} \le 1, \forall (qp, jl) \in ZN, \text{ in which v = 1, …,k;} \quad (6)$$

$$x_{jlv} = 0 \text{ or } 1, \text{ in which j = 1, …, n, l = 1,…,m and v = 1, …, k} \quad (7)$$

The objective function (1) minimizes the total number of stations and objective function (2) minimizes the total cycle time. Constraint (3) assures that each task will be assigned to a single station. Constraint (4) addresses precedence relations. Constraint (5) is aimed at tasks with positive zoning that should be assigned to the same stations and constraint (6) targets tasks with negative zoning that should not be assigned to the same stations.

4.4 Stochastic Environment Considerations

The stochastic environment incorporates the random time of completion of tasks and random arrival of demand for each product model.

4.4.1.Stochastic Time Considerations

The time in this problem is considered to be a random time which varies between the minimum and the maximum times in the standard problem and the case study dataset. However, in this study, the random values of time are generated after categorizing the tasks under number of time intervals to maintain the time variation among tasks of larger time and those with smaller time. Due to the task completion time variability of same tasks of different models assumption (section 4.1), different times are generated for model-1 and model-2. The random times are generated for 50 data points for each task on MATLAB® 2013a and rounded to two decimal places.

4.4.2. Demand Arrival

The stochastic nature of the demand arrival for the mixed models for which the balancing problem is carried out is considered through the following scenarios, in which the two models are subjected to different demand proportions from the total demand for the models based on the same time horizon.

Table 2. The Product Demand Arrival Scenario

Demand Scenario	Demand Proportion	
	Model-I	Model-II
1	0.75	0.25
2	0.60	0.40
3	0.50	0.50
4	0.40	0.60
5	0.25	0.75

To solve the developed mathematical model based on the experimental datasets, two optimization meta-heuristics are considered, namely, Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) and Multi-Objective Genetic Algorithm (MOGA). The basic objective being the selection of an efficient method to solve the assembly line problems of the current type based on problem size variations of small, medium and large scale.

5. Representation of Chromosome (Solution)

In the genetic algorithm, an individual is an encoding of a potential solution. The encoding of solutions used in this study is of type 'one-to-one', which means that each solution is represented exactly by one chromosome and the decoding of each chromosome results in exactly one solution for the problem. The encoding scheme is workstation oriented and it is similar to the one used by Simaria and Vilarinho (2004). The chromosome is a string of length N where each element represents a task and the value of each element represents the workstation to which the corresponding task is assigned. Fig.5 shows a chromosome and the decoded balancing solution.

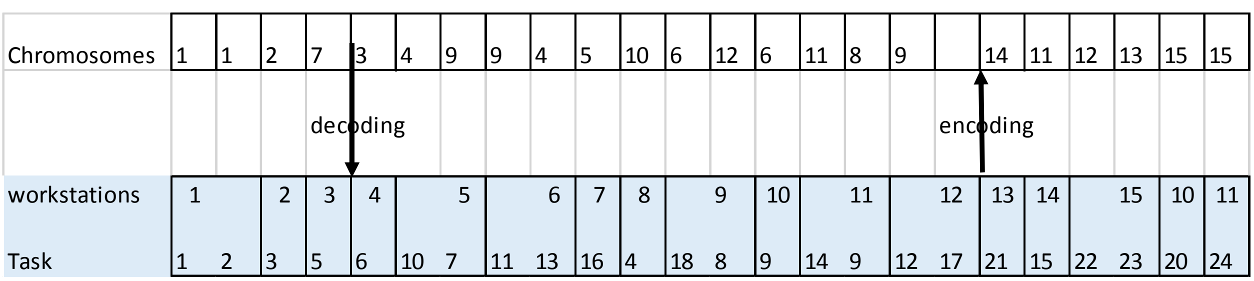


Fig.5.An example of a chromosome and the corresponding balancing solution

5.1 Population Initialization

The population is initialized by either using random generation (Tang and Liang, 2012), or by the creation of feasible population which satisfy the constraints if any by using different heuristic rules, or in some studies, solutions obtained from another optimization algorithm are used to seed the initial population (Goldberg, 1989). In this study, five heuristic rules are used to generate feasible initial solutions for the three standard problems and for the case study considered. All the heuristics rules are applied to generate sub-populations which later form a complete feasible population with random permutation and concatenation of the matrices of the subpopulations. The five priority rules used in generating the subpopulation are:

i. Maximum total number of follower tasks (Arcus, 1965)
ii. Maximum number of immediate follower tasks (Hailemariam, 2009)
iii. Minimum total number of predecessor tasks (Elsayed and Thomas, 1994)
iv. Minimum task number (Arcus, 1965)

v. Random task assignment (Arcus, 1965)

The algorithms parameter setups are discussed in Table 3.

Table 3. Parameter setup

	NSGA-II	MOGA
Population		
Mertens	48	48
Lutz	96	96
Kilbridge	96	96
Case Study	112	112
Maximum generation	50,000	50,000
Other Algorithm Specific Parameters	Distribution index for crossover: 20	Crossover fraction: 0.8
	Distribution index for mutation: 100	Mutation rate: 0.01

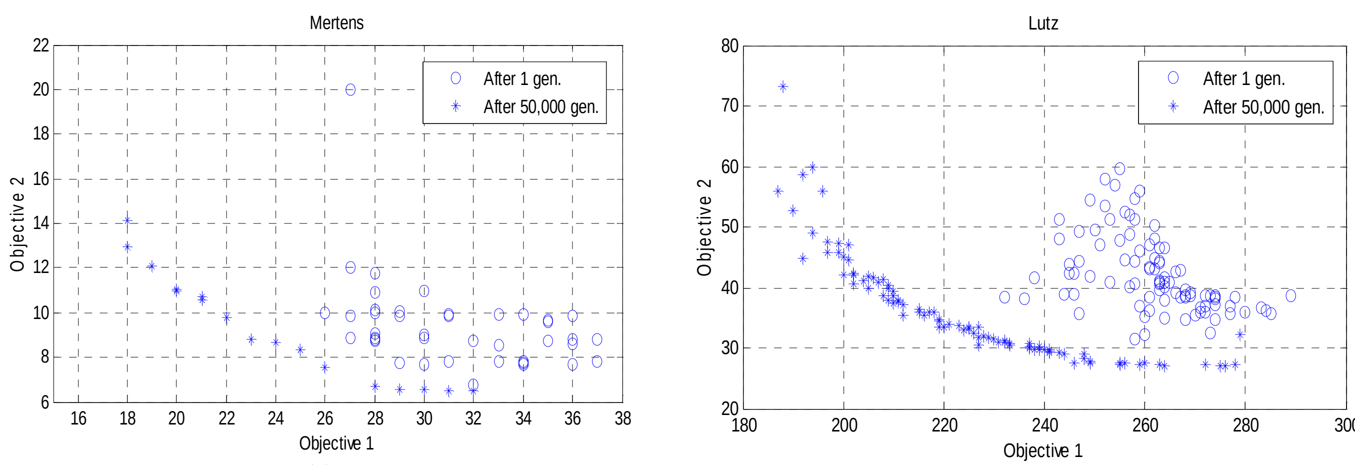
6. Results and Discussion

A perfect balance of the line means the combination of the elements of the work to be done in such a manner that at each station the sum of the elemental times just equals the cycle time. When a perfect balance cannot be achieved, we measure the effectiveness of the balance by the different methods. In this research the following three performance measures are used to compare the algorithms used based on the experimental datasets considered in the demand arrival scenarios formed.

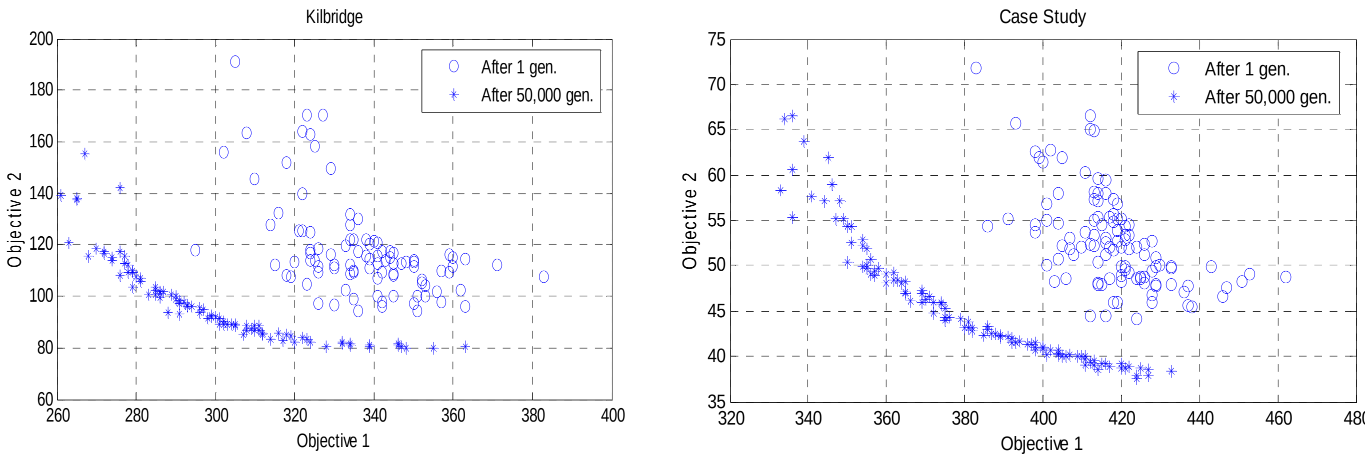
6.1 Results of the Algorithms

Both the NSGA-II and MOGA are coded using MATLAB® 2013a and with the basic parameters set, they were run on a PC with Intel® Core™ i3-2310M CPU with 2.10GHz processor speed and 4.0 GB (3.16GB usable) RAM for 50,000 generations based on the test problem selected and the demand arrival scenario considered.

Evolutionary Algorithms (EAs), under which the two algorithms used, MOGA and NSGA-II, fall do statistically converge to the true Pareto front (Van Veldhuizen and Lamont, 1998). Consequently, the convergence of the algorithm from the initial generation to the 50,000th generation is shown in Fig.6 (a) through (d) for each test problem based on equal likelihood of the product demand arrival scenario (i.e., Scenario-3). From this we can see how the solution advanced towards the Pareto optimal solutions, and in which the solutions (Chromosomes) advance to the axes of the two antagonistic objectives. The other thing we can see from these convergence diagrams is that all the individuals' chromosomes at the 50,000th generation dominate the individuals at the 1st generation. Moreover, it is worthy to note that NSGA-II maintains the number of solution constant up to the end of the maximum generation due to the elite preservation technique it uses. This is done by adding all individual in the front until the population reaches N (number of population used). This is why some chromosomes have survived even though they are dominated by another chromosome. If in some cases the population exceeds N then individuals in the particular front are selected based on their crowding distance in the descending order until the population size is N. And hence the process repeats to generate the subsequent generations.

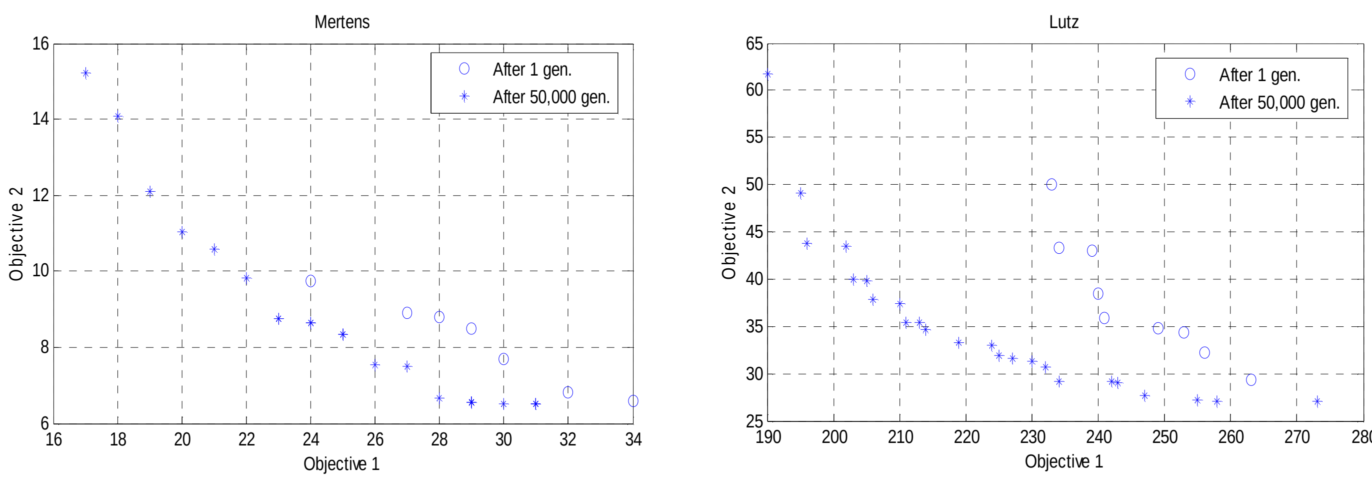


(a) (b)

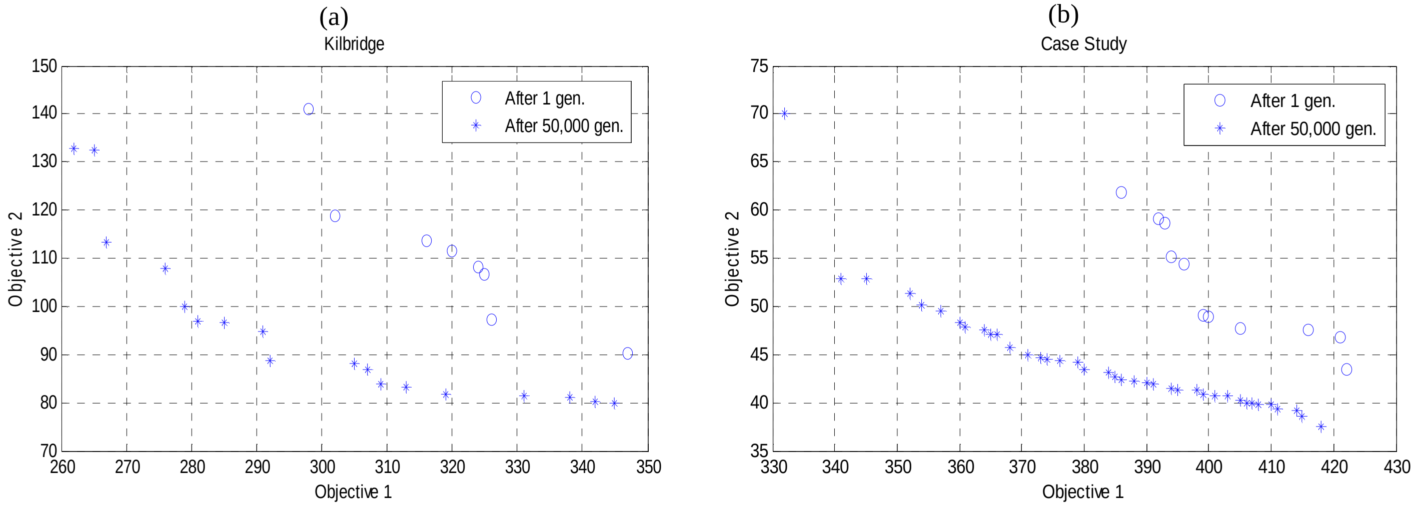


(c) (d)

Figure 6. The convergence of the NSGA-Solutions after 50,000 generations: Mertense, Lutz, Kilbridge, and Case Company problem



(a) (b)



(c) (d)

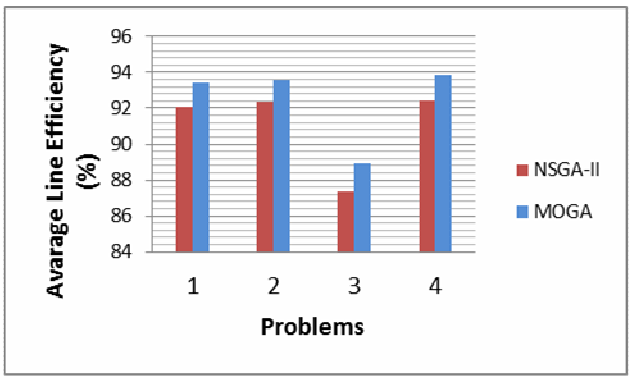
Figure 7. The convergence of the NSGA-II Solutions after 50,000 generations: Mertens (a), Lutz (b), Kilbridge (c), and Case Company problem (d)

The convergence of the MOGA from the initial generation to the 50,000th generation is shown in Figure 7 (a) through (d) for each test problem based on equal likelihoods of the product demand arrival scenario (i.e., Scenario-3). From this one can observe that the number of points on the Pareto front increases with generation resulting in relatively smoother front. And the convergence towards the minima is also significantly visible.

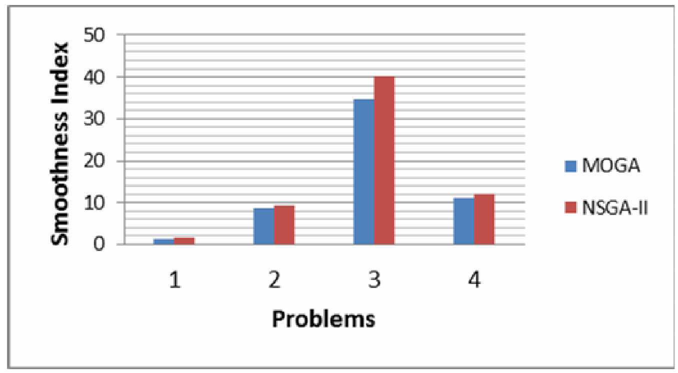
6.2 Comparison of Methods

In this section, the comparison of the performances of lines resulted from the two solution approaches (NSGA-II vs. MOGA) and the solution methods with the current setup of the case company dataset are made. The two algorithms used in this study are compared on the basis of how efficient is the line we get through the optimization, how smooth is the resulting line i.e, smoothness index, and the run time needed by the algorithm to reach the desired generations (stopping criteria).

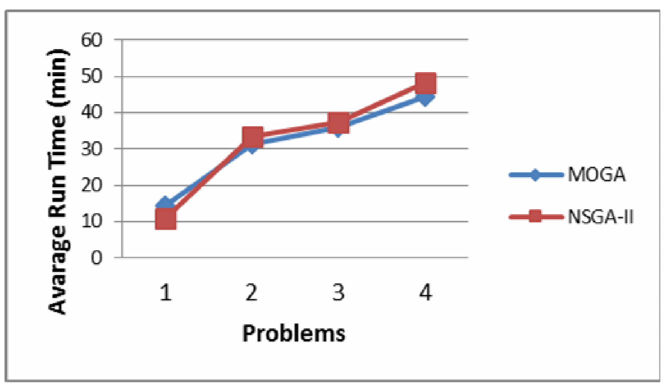
From the average efficiencies of the two algorithms Fig.8 (b), one can see that MOGA performs well in all the test problems averaged from the five demand scenarios considered. And from the average line smoothness indices of results of MOGA and NSGA-II as a result of reassignment of tasks to the workstations Fig.8 (b) we can be observed that MOGA outperformed NSGA-II in all test problems. However, one can see that in most of the test problems (Mertens, Lutz and Case company dataset) they have resulted in approximate values, except Kilbridge in which MOGA surpasses NSGA significantly.



(a)



(b)



(c)

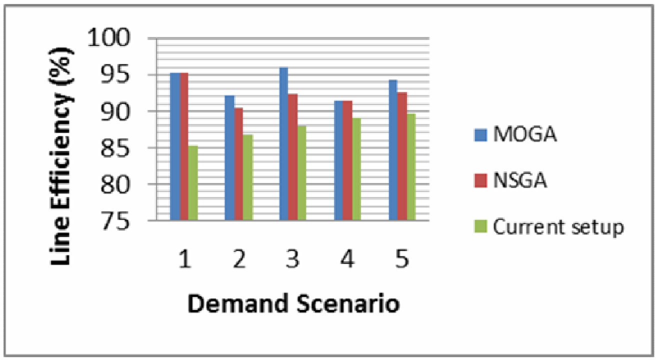
Figure 8. Average Line efficiencies (a), average line smoothness indices (b) and the Run (execution) times (c) of the two algorithms

From the average efficiencies of the two algorithms Figure 8 (b), one can see that MOGA performs well in all the test problems averaged from the five demand scenarios considered. And from the average line smoothness indices of results of MOGA and NSGA-II as a result of reassignment of tasks to the workstations Figure 8 (b) we can be observed that MOGA outperformed NSGA-II in all test problems. However, one can see that in most of the test problems (Mertens, Lutz and Case company dataset) they have resulted in approximate values, except Kilbridge in which MOGA surpasses NSGA significantly.

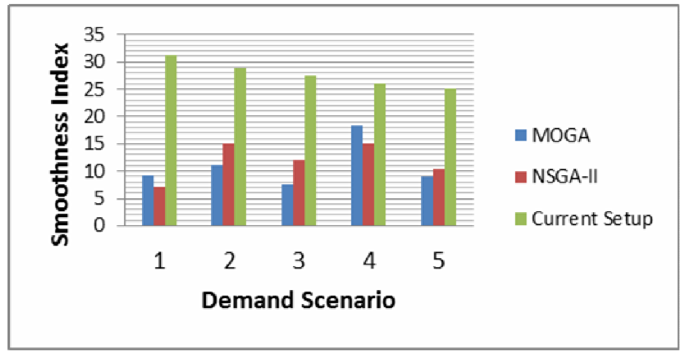
When we see the average run times of the two algorithms from Figure 8 (c), we can see that NSGA-II performs faster on the small size problem (Mertens) as compared to its rival- MOGA. This agrees with (Neda *et al*, 2012), in which it is found to be performing well in small ALB problems. However, MOGA surpasses NSGA-II in all the other three test problems (Lutz, Kilbridge and Case company dataset), which range medium to large problem instances.

6.3 Current state vs. Algorithms Results

The comparison of the current setup of the case company with the results of the two algorithms (Fig.9), shows both NSGA-II and MOGA resulted in lines which outperform the existing line set up in all the product demand arrival scenarios with average efficiency improvements (change) of 4.60% and 6.10%, respectively. This indicates that both the algorithms are capable of solving ALB problems with similar problem setups.



(a)



(b)

Figure 9. Line efficiency (a) and Smoothness Index (b) of the NSGA-II and MOGA Result, and The current case company setup for all demand arrival scenarios

As can be seen from Figure 9(b), both the algorithms resulted in much smoother lines compared to the current setup of the case company with average smoothness index reduction (improvement) of 15.85 by NSGA-II and 16.71 by MOGA Table 4 summarizes the results of the two algorithms on the case company dataset versus the performance of the existing line configuration.

Table 4. Summary of average results of MOGA and NSGA-II on the case company dataset

Problems	Average Efficiency (%)		Average Smoothness Index		Average Run Time	
	NSGA-II	MOGA	NSGA-II	MOGA	NSGA-II	MOGA
Mertens	92.078	93.406	1.708	1.31	11	14.5
Lutz	92.366	93.536	9.352	8.68	33.4	31.2
Kilbridge	87.352	88.96	40.074	34.74	37.3	36.1
Case St.	92.392	93.85	11.91	11.05	48.3	44.3
Current State	87.754	87.754	27.76	27.76	-	-
Average Improvement	4.638	6.096	-15.85	-16.71	-	-

7. Conclusions

This paper deals with the multi-objective optimization of mixed model assembly line balancing in an assemble-to-order industry with stochastic environment. A mathematical model representing the problems at hand is developed with some real-life considerations. The objectives considered in the model are the minimization of cycle time and minimization of the number of stations, which are to be optimized simultaneously. The stochastic environment incorporates the random time of completion of tasks and random arrival of demand for each product model. To solve these mathematical model two optimization meta-heuristics are considered, namely, Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) and Multi-Objective Genetic Algorithm (MOGA). The basic objective being the selection of an efficient method to solve the assembly line problems of the current setup type based on problem size variations of small, medium and large scale instances.

To test the performance of the algorithms three different size standard problems; Mertens (7), Lutz (32), and Kilbridge (45), and a case study problem (55) were considered. And five heuristic rules are used to generate feasible initial solutions for the three standard problems and for the case study considered. All the heuristics rules are applied to generate subpopulations which later form a complete feasible population with random permutation and concatenation of the matrices of the subpopulations. The two algorithms are compared based on the average line efficiency from the demand scenario considered, average smoothness index, and average run time required by the algorithm to reach the specified stopping criteria. The result has shown that NSGA-II performs faster on the small size problem as compared to its rival- MOGA. However, MOGA surpasses NSGA-II, in all the other test problems ranging from medium to large problem instances. Moreover, MOGA is found to result in efficient lines in all the considered test problems. Nevertheless, both algorithms have resulted in significant improvements in both performance measures i.e., 4.60% efficiency improvement by NSGA-II and 6.10% improvement by MOGA, and with respect to smoothness indices; NSGA-II resulted in 15.85 decreases (improvement) and MOGA resulted decrease in SI value by 16.71. These show that both algorithms could be considered as proficient candidates for solving ALB problems with the problem setups considered in this study.

In this study, the initial solutions are generated using different heuristics methods and then encoded to be used for population initialization. However, better solution diversity could be acquired by embedding this heuristics in the genetic algorithm in which a heuristic is selected randomly at a time for each chromosome (individual) generated, which in combination with other similarly generated chromosomes form the initial population. This could be incorporated in the future research endeavors with similar theme to the current study. The other areas of interest for future research include the application of the current problem setups and solving techniques to assembly lines with different line configurations (for instance U-Type) and line variation i.e., lines with station paralleling, two-sided straight type assembly lines, etc. Furthermore, future researches could include hybridization of solutions techniques to solve problems of the current type for they are capable of resulting good solution quality as compared to their traditional versions by the utilization of the strong sides of the individual techniques by mitigating their shortcomings.

References

Adil B., Lale Ö., 2007. Stochastic U-line balancing using genetic algorithms, *International Journal of Advanced Manufacturing Technology*, Vol. 32, pp. 139–147. DOI:10.1007/s00170-005-0322-4

Akpınar S. and Bayhan G.M. 2011. A hybrid genetic algorithm for mixed model assembly line balancing problem with parallel workstations and zoning constraints, *Engineering Applications of Artificial Intelligence*, Vol. 24, No. 3, pp. 449–457 https://doi.org/10.1016/j.engappai.2010.08.006.

Akpınar S., Bayhan G.M., Baykasoglu A. 2013. Hybridizing ant colony optimization via genetic algorithm for mixed-model assembly line balancing problem with sequence dependent setup times between tasks, *Applied Soft Computing*, Vol. 13, No. 1, pp. 574–589 https://doi.org/10.1016/j.asoc.2012.07.024.

Akpinar S. and Bayhan G.M. 2014. Performance evaluation of ant colony optimization-based solution strategies on the mixed-model assembly line balancing problem, *Engineering Optimization*, vol. 46, No. 6, pp. 842-862. https://doi.org/10.1080/0305215X.2013.806915.

Akpinar S. and Baykasoğlu A. 2014. Modeling and solving mixed-model assembly line balancing problem with setups. Part I: A mixed integer linear programming model, *Journal of Manufacturing Systems*, Vol. 33, No. 1, pp. 177-187 https://doi.org/10.1016/j.jmsy.2013.11.004.

Arcus A.L., 1965, A computer method of sequencing operations for assembly lines, International Journal of Production Research, Vol. 4, No. 4, pp. 259-277

AlGeddawy T. and ElMaraghy H., 2010. Design of single assembly line for the delayed differentiation of product variants, *Flexible Services and Manufacturing Journal*, Vol. 22, pp. 163–182, DOI:10.1007/s10696-011-9074-7.

Aravelli A., 2014. Multi-objective design optimization of engineering systems: uncertainty approach and practical applications, Ph.D. Thesis, Department of Mechanical and Aerospace Engineering, University of Miami, Miami, USA.

Armin S., Nils B., Malte F. 2013, The assembly line balancing and scheduling problem with sequence-dependent setup times: problem extension, model formulation and efficient heuristics, *OR Spectrum*, Vol. 35, pp. 291–320. DOI: https://doi.org/10.1007/s00291-011-0265-0.

Bagher M., M. Zandieh, H. Farsijani., 2011. Balancing of stochastic U-type assembly lines: an imperialist competitive algorithm, *The International Journal of Advanced Manufacturing Technology*, Vol. 54, pp. 271–285. DOI: 10.1007/s00170-010-2937-3.

Baudin M., 2002. Lean assembly: The nuts and bolts of making assembly operations flow, productivity, ed. New York: Productivity Press.

Bautista J., Pereira J. 2002. Ant Algorithms for Assembly Line Balancing. In: Dorigo M., Di Caro G., Sampels M. (eds) Ant Algorithms. ANTS 2002. *Lecture Notes in Computer Science*, Vol 2463. Springer, Berlin, Heidelberg

Beyer H.-G. and Deb K., 2001. On self-adaptive features in real-parameter evolutionary algorithm, *IEEE Transactions on Evolutionary Computation*, Vol. 5, No. 3, pp. 250–270, DOI: 10.1109/4235.930314.

Blum C., Bautista J., Pereira J. 2008. An extended beam-ACO approach to the time and space constrained simple assembly line balancing problem. In: van Hemert J., Cotta C. (eds) Evolutionary Computation in Combinatorial Optimization. *EvoCOP 2008*. Lecture Notes in Computer Science, Vol 4972. Springer, Berlin, Heidelberg, pp. 85–96, https://doi.org/10.1007/978-3-540-78604-78.

Bock S., 2008. Using distributed search methods for balancing mixed-model assembly lines in the automotive industry, *OR Spectrum*, Vol. 30, pp. 551–578. DOI:10.1007/s00291-006-0069-9.

Bukchin J. and Rubinovitz J., 2003. A weighted approach for assembly line design with station paralleling and equipment selection, *IIE Transactions*, Vol. 35, No. pp. 73-85, https://doi.org/10.1080/07408170304429.

Chiang W.-C., Urban T.L. & Xu X. 2012. A bi-objective metaheuristic approach to unpaced synchronous production line-balancing problems, *International Journal of Production Research*, Vol. 50, No. 1, pp. 293-306. https://doi.org/10.1080/00207543.2011.571460.

Carlo-Colon H.J., Nambiar A.N. 2008. A max-plus algebra based algorithm for assembly line balancing problem, *Proceedings of the 2008 Industrial Engineering Research Conference (IERC)*.

Chica M., Cordón Ó., Damas S., Bautista J. 2010. Adding Diversity to Two Multiobjective Constructive Metaheuristics for Time and Space Assembly Line Balancing. In: Lee S., Suárez R., Choi BW. (eds) Frontiers of Assembly and Manufacturing. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-14116-4_17

Dong J., Zhang L., Xiao T. & Mao H. 2014. Balancing and sequencing of stochastic mixed-model assembly U-lines to minimise the expectation of work overload time, *International Journal of Production Research*, Vol. 52, pp. 7529-7548. https://doi.org/10.1080/00207543.2014.944280.

Elsayed A. E. and Thomas O. B., 1994. Analysis and control of production systems, 2nd ed. New Jersey: Prentice Hall International Series in Industrial and Systems Engineering.

Fattahi P., Roshani A. and Roshani A. 2011. A mathematical model and ant colony algorithm for multi-manned assembly line balancing problem, *International Journal of Advanced Manufacturing Technology*, Vol. 53, pp. 363-378. https://doi.org/10.1007/s00170-010-2832-y.

Gamberini R., E. Gebennini , A. Grassi, A. Regattier, 2009, A multiple single-pass heuristic algorithm solving the stochastic assembly line rebalancing problem, *International Journal of Production Research*, Vol. 47, No. 8, pp. 2141-2164. https://doi.org/10.1080/00207540802176046.

Goldberg D.E., 1989. Genetic algorithm in search, optimization and machine learning," ed. Reading, Massachusetts: Addison Wesley.

Grzechca W., IntechOpen, DOI: 10.5772/19953.

Hailemariam D., 2009. Mixed Model Assembly Line Balancing Using Simulation Techniques :A Case Study in Ambassador Garment and Trade PLC, School of Graduate Studies, Addis Ababa University.

Kadir B., Ibrahim K., Sule Itir S., David Z. Z., 2016. Lexicographic bottleneck mixed-model assembly line balancing problem: Artificial bee colony and tabu search approaches with optimised parameters, *Expert Systems with Applications*, Vol. 50, No. 15, 151-166, http://dx.doi.org/10.1016/j.eswa.2015.12.018.

Kara Y. and Tekin M. 2009. A mixed integer linear programming formulation for optimal balancing of mixed-model U-lines, *International Journal of Production Research*, Vol. 47, No. 15, pp. 4201-4233. https://doi.org/10.1080/00207540801905486.

Kara Y. 2008. Line balancing and model sequencing to reduce work overload in mixedmodel U-line production environments, *Engineering Optimization*, Vol. 40, No. 7, pp. 669-684 https://doi.org/10.1080/03052150801982509.

Kara Y., Atasagun Y., Gökçen H., Hezer S. and Demirel N., 2014. An integrated model to incorporate ergonomics and resource restrictions into assembly line balancing, *International Journal of Computer Integrated Manufacturing*, Vol. 27, No. 11, pp. 997-1007. https://doi.org/10.1080/0951192X.2013.874575

Kaveh A., and Laknejadi K., 2011. A novel hybrid charge system search and particle swarm optimization method for multi-objective optimization, *Expert Systems with Applications*, Vol. 38, No. 12, pp. 15475–15488, Doi:10.1016/j.eswa.2011.06.012.

Kucukkoc I. and Zhang D.Z. 2014. Simultaneous balancing and sequencing of mixed-model parallel two-sided assembly lines, *International Journal of Production Research*, Vol. 52, pp. 3665-3687. https://doi.org/10.1080/00207543.2013.879618.

Lapierre S.D., Ruiz A., Soriano P., 2006. Balancing assembly lines with tabu search, *European Journal of Operational Research*, Vol. 168, No. 3, pp. 826–837 https://doi.org/10.1016/j.ejor.2004.07.031.

Li-Man L., 2014. Construction and comparison of multi-model and mixed-model assembly lines balancing problems with bi-objective, *Journal of Industrial and Production Engineering*, Vol. 31, No. 8, pp. 483-490. https://doi.org/10.1080/21681015.2014.992984.

Liu S. B., Ng K. M., and Ong H. L., 2008. Branch-and-bound algorithms for simple assembly line balancing problem, *International Journal of Advanced Manufacturing Technology*, Vol. 36, pp. 169–177 https://doi.org/10.1007/s00170-006-0821-y.

Liu C-M. and Chen C.-H., 2002. Multi-section electronic assembly line balancing problems: A case study, *Production Planning & Control: The Management of Operations*, Vol. 13, No. 5, pp. 451-461, DOI: 10.1080/09537280210142781.

Manavizadeh N., Rabbani M., Moshtaghi D., Jolai F., 2012. "Mixed-model assembly line balancing in the make-to-order and stochastic environment using multi-objective evolutionary algorithms," *Expert Systems with Applications*, Vol. 39, No. 15, pp. 12026–12031, https://doi.org/10.1016/j.eswa.2012.03.044.

Naveen K., Dalgobind M., 2013. Assembly line balancing: A review of developments and trends in approach to industrial application, *Global Journal of Researches in Engineering Industrial Engineering*, Vol. 13, No. 2, pp. 29-50.

Noorul Haq A., Jayaprakash J., and Rengarajan K., 2006. A hybrid genetic algorithm approach to mixed-model assembly line balancing, *International Journal of Advanced Manufacturing Technology*, vol. 28, pp. 337–341, DOI:https://doi.org/10.1007/s00170-004-2373-3.

Nourmohammadi A. and Zandieh M., 2011. Assembly line balancing by a new multi-objective differential evolution algorithm based on TOPSIS, *International Journal of Production Research*, Vol. 49, No. 10, pp. 2833-2855. https://doi.org/10.1080/00207540903473367.

Parames C., Suchada K., 2013. PSONK: particle swarm optimization with negative knowledge for multi-objective u-shaped assembly lines balancing with parallel workstations", *Journal of Advanced Manufacturing Systems*, Vol. 12, No. 1, pp. 15–41. https://doi.org/10.1142/S0219686713500029

Parisa S., Rui D., José S., 2018. Balancing mixed-model assembly systems in the footwear industry with a variable neighbourhood descent method, *Computers & Industrial Engineering*, Vol. 121, pp. 161-176, https://doi.org/10.1016/j.cie.2018.05.020

Pavel R. and Z. Ulrych, 2012. Simulation optimization in manufacturing systems, *Annals & Proceedings of DAAAM International*, Vol. 23, pp. 221-224.

Ponnambalam S.G., Aravindan P. & Naidu G.M., 2000. A multi-objective genetic algorithm for solving assembly line balancing problem, *International Journal of Advanced Manufacturing Technology*, Vol. 16, pp. 341–352 https://doi.org/10.1007/s001700050166.

Randy L. H. and Sue E.H., 2004. Practical Genetic Algorithms, Hoboken, New Jersey: John Wiley & Sons, Inc.

Rea K., Hiroshi K., Mitsuo G., 2008, U-shaped assembly line balancing problem with genetic algorithm, *International Journal of Production Research*, Vol. 46, No. 16, pp. 4637-4649. https://doi.org/10.1080/00207540701247906.

Sachin J., Prashant A., 2014. Balancing U-shaped assembly lines with resource dependent task times: A Simulated Annealing approach, *Journal of Manufacturing Systems*, Vol. 33, pp. 522–534. https://doi.org/10.1016/j.jmsy.2014.05.002.

Saif U., Guan Z., Wang B. & Mirza J. 2014. Pareto lexicographic α-robust approach and its application in robust multi objective assembly line balancing problem", Frontiers of Mechanical Engineering, Vol. 9, pp. 257–264. https://doi.org/10.1007/s11465-014-0294-x.

Samy S. N. and El-Maraghy H., 2010. A model for measuring products assembly complexity, *International Journal of Computer Integrated Manufacturing*, Vol. 23, pp. 1015–1027. https://doi.org/10.1080/0951192X.2010.511652

Simaria A.S. and Vilarinho P.M. 2004. "A genetic algorithm based approach to the mixed-model assembly line balancing problem of type II," *Computers & Industrial Engineering*, Vol. 47, No. 4, pp. 391–407, DOI: 10.1016/j.cie.2004.09.001.

Sheu D.D. & Chen J.-Y. 2008. Line balance analyses for system assembly lines in an electronic plant, *Production Planning & Control: The Management of Operations*, Vol. 19, pp. 256-264, https://doi.org/10.1080/09537280801966616.

Sivasankaran P. and Shahabudeen P., 2013. Genetic algorithm for concurrent balancing of mixed-model assembly lines with original task times of models, *Intelligent Information Management*, Vol. 5, pp. 84-92. DOI: 10.4236/iim.2013.53009.

Su P., Wu N.Q. and Yu Z.Q. 2014. A petri net-based heuristic for mixed-model assembly line balancing problem of Type-E, *International Journal of Production Research*, Vol. 52, No. 5, pp. 1542-1556 https://doi.org/10.1080/00207543.2013.849010.

Tamás Koltai T., Tatay V. & Kalló N. 2014. Application of the results of simple assembly line balancing models in practice: the case of a bicycle manufacturer, *International Journal of Computer Integrated Manufacturing*, vol. 27, pp. 887-898 https://doi.org/10.1080/0951192X.2013.834482.

Tang Q. H. and Y. L. Liang, 2012. The improved genetic algorithm for balancing mixed-model assembly line, *Applied Mechanics and Materials*, Vol. 127, pp. 603-608. https://doi.org/10.4028/www.scientific.net/AMM.127.603.

Uddin M.K. and Lastra J.L.M, 2011. Assembly Line Balancing and Sequencing, Assembly Line - Theory and Practice,

Uğur Ö. and Bilal T., 2009. Balancing of mixed-model two-sided assembly lines, *Computers & Industrial Engineering*, Vol. 57, pp. 217–227 https://doi.org/10.1016/j.cie.2008.11.012.

Uğur Ö., Talip K., Bilal T., 2011. A genetic algorithm for the stochastic mixed-model U-line balancing and sequencing problem, *International Journal of Production Research*, Vol. 49, No. 6, pp. 1605-1626 https://doi.org/10.1080/00207541003690090.

Van Hop N. 2004. A heuristic solution for fuzzy mixed-model line balancing problem, *European Journal of Operational Research*, Vol. 168, pp. 798–810. https://doi.org/10.1016/j.ejor.2004.07.029.

Van Veldhuizen D.A. and Lamont G.B. 1998. Evolutionary computation and convergence to a pareto front, In Proceedings, Genetic Programming 1998 Conference, pp. 221--228, University of Wisconsin, Madison, Wisconsin, USA.

Vargas-Vera M., Hu B. (eds) Pervasive Computing and the Networked World. ICPCA/SWS 2013. Lecture Notes in Computer Science, vol 8351. Springer, Cham.https://doi.org/10.1007/978-3-319-09265-2_43

Vilarinho P.M. and Simaria A.S. 2002. A two-stage heuristic method for balancing mixed-model assembly lines with parallel workstations, *International Journal of Production Research*, Vol. 40, No. 6, pp. 1405-1420 https://doi.org/10.1080/00207540110116273.

Vilarinho P.M. and Simaria A.S., 2006. ANTBAL: an ant colony optimization algorithm for balancing mixed-model assembly lines with parallel workstations, *International Journal of Production Research*, Vol. 44, No. 2, pp. 291-303. https://doi.org/10.1080/00207540500227612.

Wenqiang Z. and Mitsuo G., 2011. An efficient multiobjective genetic algorithm for mixed-model assembly line balancing problem considering demand ratio-based cycle time, *Journal of Intellegent Manufacturing*, Vol. 22, pp. 367–378 https://doi.org/10.1007/s10845-009-0295-5.

Wenqiang Z., Weitao X., and Mitsuo G., 2014. Hybrid multiobjective evolutionary lgorithm for assembly line balancing problem with stochastic processing time, *Procedia Computer Science*, Vol. 36, pp. 587-592 https://doi.org/10.1016/j.procs.2014.09.058.

Wu E.-F., Jin Y., Bao J.-S. & Hu X.-F. 2008. A branch-and-bound algorithm for two-sided assembly line balancing, *The International Journal of Advanced Manufacturing*, Vol. 39, pp. 1009–1015. https://doi.org/10.1007/s00170-007-1286-3

Xu W. and Xiao T. 2009. Robust balancing of mixed model assembly line, COMPEL - The *International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, Vol. 28, No. 6, pp. 1489-1502 https://doi.org/10.1108/03321640910992038.

Yabo L., Yang W., Feng Z. 2014. Optimization algorithm for balancing and sequencing mixed assembly line. In: Zu Q.,

Zacharia P. Th., Nearchou A.C., 2012. Multi-objective fuzzy assembly line balancing using genetic algorithms, *Journal of Intelligent Manufacturing*, Vol. 23, pp. 615–627. https://doi.org/10.1007/s10845-010-0400-9.

Zein-Eldin R.A. 2014. A hybrid SS-SA approach for solving multi-objective optimization problems, *European Journal of Scientific Research*, Vol. 121, No. 3, pp. 310-320.

Biographical notes

Ayele Legesse is with the Department of Mechanical Engineering, Haramaya Institute of Technology, Haramaya University, Haramaya, Ethiopia

Ermias Tesfaye and Eshetie Berhan are with the School of Mechanical and Industrial Engineering, Addis Ababa Institute of Technology, Addis Ababa University, Addis Ababa, Ethiopia