

Workflow balancing in a speed frame assembly shop floor operations through genetic algorithm

S.Rajakumar

*Department of Mechanical Engineering
Sree Sakthi Engineering College, Karamadai, Coimbatore, Tamilnadu
rajakumar6@yahoo.com*

Date of Submission: 30-07-2020

Date of Acceptance: 09-08-2020

ABSTRACT: The assembly planning of speed frame, a textile machine involves allocation of operations to cross trained operators. The assignment of assembly operations to the operators is modeled into a parallel machine scheduling problem with precedence constraints using the objective of minimizing workflow among operators. Workflow is defined as the workloads assigned to the operators. A genetic algorithm (GA) is used to analyze this problem and its performance is compared with traditional heuristics. The Relative Percentage of Imbalance (RPI) is adopted for evaluating the performance of these heuristics. The GA produced well balanced workflow schedules with lesser RPI values for all operators than other heuristics. A computer program has been coded on IBM / PC compatible system in the C++ language for studying the performance of real data from the shop floor.

Keywords: workflow, workload, balancing, scheduling, assembly, genetic algorithm

I. INTRODUCTION

The machine (shop floor) scheduling involves assignment of jobs to machines supported some objective function. Different algorithms are developed by researchers to estimate the performance of scheduling. The assembly planning of a speed frame considered for the study may be a repetitive type manufacturing system, which involves assignment of assembly operations, to be executed in parallel by operators. The operators are cross trained to do all types of operations required for an entire assembly. There are n operations to be executed by m operators in parallel to finish the assembly. There are some precedence operations to be allocated first among all operations. This is often modeled as a classical parallel machine scheduling problem with precedence constraints. In parallel machine scheduling with precedence constraints, there are n jobs to be scheduled on m machines satisfying the given precedence constraints. An

operation can't be started until all its precedence operations are completed. During this research work, the workflow balancing approach is employed to assign operations to operators within the assembly workplace. Workflow refers to workloads of the operators, which are represented by sum of all assembly-processing times assigned to them. The idle time isn't considered for calculating the workloads. The assembly operations are scheduled to the operators in such how that their workloads got to be balanced. The choice of operations for the assignment is predicated on four different heuristics: Random (RANDOM), Shortest Processing Time (SPT), Longest Processing Time (LPT) and Genetic Algorithm (GA). Rajakumar et.al[1] compared GA with RANDOM heuristic for the above problem. In this work, GA is compared with other two heuristics namely SPT and LPT and also with RANDOM. Previously, they [2] also conducted simulated experiments on the above problem using three traditional heuristics. All the preceding operations are scheduled first and succeeding operations are scheduled subsequently. Both the preceding and succeeding operations are selected individually consistent with the above four assembly scheduling heuristics. The Relative Percentage of Imbalance (RPI) in workloads of operators is adopted to estimate the performance of heuristics [3].

II. WORKFLOW BALANCING IN ASSEMBLY OPERATIONS SCHEDULING

Shop Floor Control (SFC) provides clear visibility over the manufacturing activities administered in a workplace. The manufacturing activities must be properly scheduled for efficient uses of resources. The workflow must be distributed uniformly in workplace to avoid bottlenecks. The presence of bottlenecks results in under utilization of resources. The case study involves allocation of assembly operations to the

operators in order that they're well utilized. The bottleneck refers to maximum time which will be consumed by an operator to end all the operations assigned to him. This bottleneck affects the work of other operators in order that productivity is lost within the workplace. Hence, workflow balancing is adopted to avoid such bottlenecks. The operators must be assigned with suitable workflow so as each operator will have balanced workloads. The workflow balancing is to minimize imbalances in workloads present among the operators.

III. LITERATURE REVIEW

In parallel machine scheduling problem, there are m machines on which n jobs are to be scheduled. Each job has got to be scheduled on one the machines during the fixed time interval. The aim is to seek out the schedule that optimizes a particular performance measure [4]. The jobs are divided into groups and that they are individually scheduled with list scheduling algorithms and joined together for the ultimate schedule [5]. An imbalance measure is a function of workload differences among machines or machine groups [6]. Genetic algorithms can handle any objective functions and constrains defined on discrete, continuous or mixed search space [7]. Heuristic serial schedule (SS) algorithm is proposed for unrelated parallel machine scheduling problem of minimizing makespan subject to precedence constraints [8]. The precedence constraints greatly increase the complexity of scheduling problem to get feasible solutions, especially during a parallel machine environment [9]. Hassan et al.[10] considered the matter of unrelated parallel machines with precedence constraints (UPMPC), with the aim of minimizing the makespan. Prot et al.[11] discussed the structure of precedence constraints plays an incredible role on the complexity of scheduling problems[12]. Thiago et al.[13] solved the cross-docking center,(CDC) scheduling problem as a parallel machine problem with precedence relationships among trucks. Assembly scheduling aims to synchronise sub-assembly and final assembly processes to minimise the entire costs satisfying assembly precedence constraints [14]. Workflow management is important for process monitoring and bottleneck analyses, especially within the assembly systems

where many material, information, value, etc. integrate together [15]. Ma et al.[16] considered a case study of workforce allocation during a gear production line at a motorbike gearing factory. Work content imbalance between stations degrades system performance. The effects of this imbalance are often mitigated by proper job sequencing [17]. Now it's evident that case study could also be modeled into a parallel machine scheduling problem with precedence constraints and be solved with heuristic algorithms.

IV. PROBLEM DEFINITION

The manufacturing unit considered for this study produces textile machines namely carding, draw and speed frames. These machines are utilized to produce yarn from raw cotton. The assembly planning of a speed frame is taken into account for case study. The length of speed frame is 25 meters and required components of assembly are delivered to the assembly floor. The mainframe of speed frame is erected at one place in the shop floor and other components and sub assemblies are fixed on it along entire length at various locations. Some components are fixed first because of precedence relationship to accommodate other fitting components. The fitting of sub assemblies and other components can be executed parallel at different places along the entire length of mainframe. The aim of this study is to find out suitable scheduling heuristic in allocating assembly operations to operators so that their workloads are well balanced. The balanced workloads reduce idle time present and ensure the earliest completion of assembly. This problem of workflow balancing for the assembly of speed frame is modelled as a parallel machine scheduling problem using the analogy of operations for jobs and operators for machines with precedence constraints. This assembly scheduling problem deals with assigning of n operations on m operators with precedence constraints. The operations may be assigned on any one of the operators based on an objective function. Each operation is indicated with the processing time p_j where $j = 1, 2, 3, \dots, n$. The processing time is deterministic and integer.

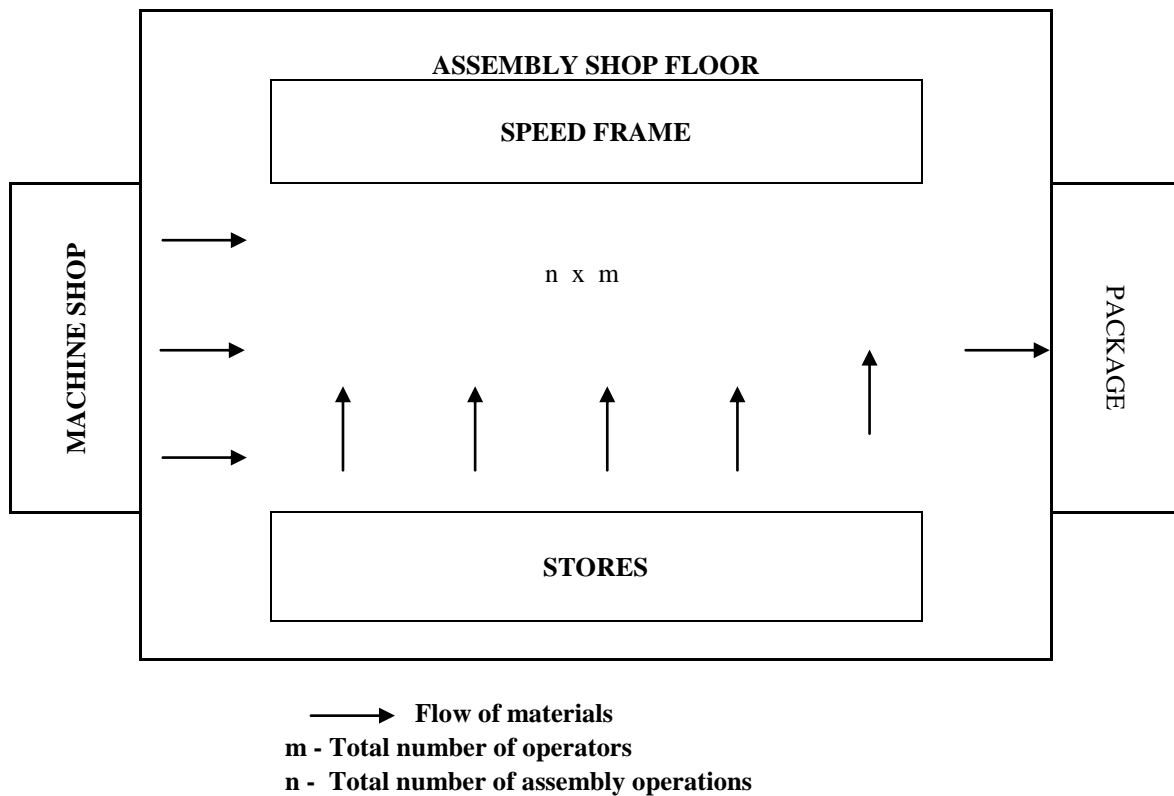


Fig. 1. Layout of the assembly floor

V. METHODOLOGY

The assembly problem consists of n operations J_i , $i = 1, 2, 3, \dots, n$ and m operators M_j , $j = 1, 2, 3, \dots, m$. The operation J_i requires operation time p_j , which may be processed by anybody of the operators. There are some precedence operations, which must be completed before starting new operations. The operations are allocated such that workloads of operators are evenly distributed. An operation could also be allocated only after ensuring that each one precedence operations are allocated. Let S represents the list of assembly operations to be assigned to the operators. This list is made with operations numbers and their operation times.

$S = \{p_i\}$ $i = 1, 2, 3, \dots, n$, where p_i indicates the operation time of i th operation.

Let S_1 be the list of preceding operations that has got to be assigned before succeeding operations and S_2 is another list containing succeeding operations. Both S_1 and S_2 are extracted from the list S . The assembly operations must be assigned such workloads of operators are well balanced to realize maximum efficiency within the workplace. The scheduling heuristics play an important role in producing good schedules with

balanced workloads. This problem belongs to NP complete and hence it's solved using heuristic algorithms. The case study consists of sixty two assembly operations to be scheduled on six cross trained operators. Both the preceding and succeeding assembly operations are executed in parallel.

The four scheduling heuristics RANDOM, SPT, LPT and GA are used for the choice of an operation from the list of unscheduled operations. The operator with the smallest amount workload is chosen for assigning new operation. The scheduling heuristics are applied to both the sub lists separately. These heuristics are combined to supply ten different heuristics. The name of the heuristics is represented with two characters. The primary character within the heuristic name represents the heuristics applied to the list of precedence operations and therefore the second character represents the heuristics applied to the list of succeeding operations. The name of the heuristics applied to both the preceding and succeeding operations are shown within the following Table 1.

Workload Allocation

The workload allocation algorithm is described below [2].

Step 1 Select an operation from the lists of precedence operations S_1 or succeeding operations S_2 if all the precedence operations are allocated.

The lists of preceding and succeeding operations are prepared using heuristic algorithms.

Step 2 Find out the cumulative workload W_i of each operator. The workload W_k of an operator is calculated by adding up of all the processing time of operations assigned to him. Workload of any operator is

$$W_i = \sum_{j=1}^t p_j \text{ if } j \in R_i \quad (i = 1, \dots, m) \tag{1}$$

R_i – list of allocated jobs
 t – total number of operations assigned
 m – number of operators

Step 3 Select the operator M_k with the least workload W_k among m operators for assigning the operations from list of unscheduled operations. The lists of unscheduled operations are prepared according to the heuristics.

Step 4 Assign the operation J_i from the lists of operations S_1 or S_2 to the operator M_k .

Step 5 Repeat the steps 2 to 4 until all the operations are assigned.

Table 1
List of heuristics

Serial Number	Name of the heuristics	Name of the strategy applied to	
		Preceding operations	Succeeding operations
1	RR	RANDOM	RANDOM
2	RS	RANDOM	SPT
3	RL	RANDOM	LPT
4	SR	SPT	RANDOM
5	SS	SPT	SPT
6	SL	SPT	LPT
7	LR	LPT	RANDOM
8	LS	LPT	SPT
9	LL	LPT	LPT
10	GA	GA	GA

VI. PERFORMANCE MEASURE

The workloads of all operators are calculated and compared with maximum workload of an operator. The operator with the utmost workload is that the bottleneck within the workplace. The workload W_{max} of an operator is

taken into account as the index for comparing the workloads of other operators. The difference between maximum and available workloads is named an imbalance in workloads. The imbalances in workloads are expressed as

$$I_i = W_{max} - W_i \quad i = 1, 2, \dots, m \tag{2}$$

The allocation of operations could also be distributed to others to eliminate the bottleneck workload. The RPI in workloads of all operators are calculated for evaluating performance of varied

scheduling heuristics shown in Table 1. It denotes percentage of deviation of workloads on operators from boundary of maximum workload. It's expressed as

$$RPI = \frac{(W_{max} - W_i)}{W_{max}} \times 100 \quad (3)$$

The ten scheduling heuristics produce ten different RPI values.

VII. PROPOSED GENETIC ALGORITHM IN PARALLEL MACHINE SCHEDULING WITH PRECEDENCE CONSTRAINTS

The GA has become popular due to its efficiency in searching optimal solutions during a larger complex space. The operation times are ordered consistent with GA and assigned to the smallest amount loaded operator one by one. The population size (pop_size) is adequate to the

amount of operations **n** to make sure direction of search around optimal or near optimal solutions. Same size is maintained altogether in generations. A gene represents an operation. The length of chromosomes or number of genes is adequate to number of operations **n** considered for the experiment. A chromosome consists of two parts i.e. first part represents the preceding and second part denotes the succeeding operations as shown in Fig.2 [1].

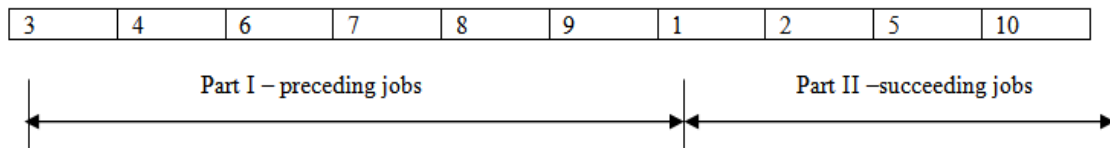


Fig. 2. Representation of a chromosome

Each chromosome represents one potential solution. Fitness function represents the average RPI among the operators. The fitness value $f(c_i)$ of every chromosome is calculated. Generally, the fitness function is taken into account as same because the objective functions for the

$$new_f(c_i) = 1 - \frac{f(c_i)}{F} \quad (4)$$

$$\text{where } F = \sum_{i=1}^{pop_size} f(c_i) \quad (5)$$

Probability of selection $p(c_i)$ of a chromosome is found by dividing new fitness value with the sum of latest fitness values of all chromosomes therein population.

$$p(c_i) = \frac{new_f(c_i)}{\sum_{i=1}^{pop_size} new_f(c_i)} \quad (6)$$

The cumulative probability $q(c_i)$ is calculated for every chromosome..

$$q(c_i) = \sum_{k=1}^i p(c_k) \quad (7)$$

A proportionate selection procedure is adopted. A random number k between 0 and 1 is generated and a chromosome is chosen on $q(c_{i-1}) < k < q(c_i)$. This process is repeated for all the chromosomes to stay population size constant. It enables the choice of the fittest chromosomes for

subsequent generation and therefore the worst are eliminated.

The probability of crossover is 0.9 in order that 90% of chromosomes of a population are selected for the genetic operations to supply off springs. Simple representation isn't fitted to combinatorial problems and hence permutation

representation is employed. Gen et. al [7] proposed Partial Mapped Crossover (PMX) for permutation representation [4]. So, PMX is employed for crossover operations and it is often an extension of two point crossovers with a special repairing procedure. It solves the conflict by legalizing the off spring with the mapping relationship. Random numbers are generated between 0 and 1 for all chromosomes. The chromosomes with random numbers less than 0.9 are selected for crossover operations. Two crossover sites are selected for every a part of a chromosome randomly by using random numbers. The crossover operations are executed by exchanging genes among paired

chromosomes separately in each part. These chromosomes are copied to subsequent population.

VIII. EXPERIMENTAL RESULTS

The assembly scheduling of a speed frame has sixty-two operations, which are to be assigned on six cross-trained operators. The lists of operation times and therefore the precedence operations are fed into computer model through an input data file. The assembly operation times are distributed between 1 and 450 minutes. The output shows the workflow in terms of operation times to the operators. The RPI among operators is calculated. The RPI values produced are between 0.6829 to 12.2508%.

Table 2
 Relative percentage of imbalance in workloads for the case study

Name of the heuristics	Operators						Average RPI
	1	2	3	4	5	6	
RR	1755	1637	1691	1636	1661	1655	4.7009
RS	1603	1704	1890	1485	1808	1545	11.5079
RL	1660	1655	1660	1695	1699	1666	1.5597
SR	1659	1638	1750	1635	1670	1683	4.4286
SS	1475	1560	1709	1887	1596	1808	11.3672
SL	1663	1655	1656	1695	1680	1686	1.3274
LR	1643	1659	1630	1685	1756	1662	4.7551
LS	1681	1545	1601	1810	1906	1492	12.2508
LL	1648	1685	1685	1675	1660	1682	0.7418
GA	1675	1675	1669	1670	1662	1684	0.6829

The workflow distribution among the operators for heuristics is shown within the Table 2. The RPI value of 0.6829 for GA is lesser than other heuristics. The proposed methodology using GA produces balanced workflow schedule. The lesser RPI values indicate the balanced workflow among the operators. Thus, GA performs better than the opposite heuristic methods of allocation of assembly operations among the operators.

IX. CONCLUSION

Workflow balancing of assembly operations of a repetitive type manufacturing system using heuristic algorithms has been studied. The workflow balancing heuristics RANDOM, SPT, LPT and GA are applied to 2 lists of assembly operations separately namely preceding and succeeding generated from the list of operations. The operator with less workload has been selected for assigning subsequent new operation. First, the precedence list is taken into account for the

allotment. After exhausting that list, the operations from the succeeding list are assigned to the operators. The computational experiments are conducted on computer model. The performance measure of RPI in workloads is employed to estimate the heuristics. The GA has produced lesser RPI values among other heuristics considered for the study. Thus, GA produces balanced workflow to the assembly of operations among the operators. Further studies may be made to match this result with the results of other Meta heuristics solutions.

REFERENCES

- [1]. Rajakumar S, Arunachalam V P, Selladuri V (2006) Workflow balancing in parallel machine scheduling with precedence constraints using genetic algorithm. Journal of Manufacturing Technology Management, 17(2):239- 254.
- [2]. Rajakumar S, Arunachalam V P, Selladuri V (2005) Simulation of workflow balancing

- in assembly shopfloor operations.. *Journal of Manufacturing Technology Management*, 16(3):265- 281.
- [3]. Rajakumar S, Arunachalam V P, Selladuri V (2004) Workflow balancing in parallel machine scheduling. *International Journal of Advanced Manufacturing Technology*, 23:366-374.
- [4]. Mokotoff E (2001) Parallel machine scheduling problems: A survey. *Asia – Pacific Journal of operation Research*, 18(2): 193-242.
- [5]. Hall LA, Schulz AS, Shmoys DB, Wein J (1997) Scheduling to minimize average completion time: off and on line algorithms, *Mathematics of Operation Research*, 22:513-544.
- [6]. Kumar N, Shankar K (2001) Comparing the effectiveness of workload balancing objectives in FMS loading. *International Journal of Production Research*, 39(5): 843 - 871.
- [7]. Gen M, Cheng R (1997) *Genetic Algorithms and Engineering Design*, New York: John Wiley & Sons.
- [8]. Liu C, Yang S (2011) A heuristic serial schedule algorithm for unrelated parallel machine scheduling with precedence constraints, *Journal of Software*, 6(6): 1146-1153.
- [9]. Mojtaba Afzalirad, Javad Rezaeian (2016) Design of high-performing hybrid meta-heuristics for unrelated parallel machine scheduling with machine eligibility and precedence constraints, *Engineering Optimization*, 48 (4): 706-726.
- [10]. Hassan MA, Kacem I, Martin S, Osman IM (2016) Unrelated Parallel Machine Scheduling Problem with Precedence Constraints: Polyhedral Analysis and Branch-and-Cut. In: Cerulli R, Fujishige S, Mahjoub A (eds) *Combinatorial Optimization, ISCO 2016, Lecture Notes in Computer Science*, Vol 9849, Springer.
- [11]. Prot D, Bellenguez-Morineau O (2018) A survey on how the structure of precedence constraints may change the complexity class of scheduling problems, *Journal of Scheduling*, 21:3–16.
- [12]. Zhang A, Qi X, Li G (2020) Machine scheduling with soft precedence constraints, *European Journal of Operational Research*, Elsevier.
- [13]. Thiago Henrique Nogueira, Felipe Provezano Coutinho, Rafael Pereira Ribeiro, Martín Gómez Ravetti (2020) Parallel machine scheduling methodology for a multi-dock truck sequencing problem in a cross-docking center, *Computers & Industrial Engineering*, Vol 143.
- [14]. Saisumpan Sooncharoen, Pupong Pongcharoen, Christian Hicks (2020) Grey Wolf production scheduling for the capital goods industry, *Applied Soft Computing*, Vol 94.
- [15]. Cheng Qian, Yingfeng Zhang, Chen Jiang, Shenle Pan, Yiming Rong (2020) A real-time data-driven collaborative mechanism in fixed-position assembly systems for smart manufacturing, *Robotics and Computer-Integrated Manufacturing*, Vol 61.
- [16]. Ma H, Lee H K, Shi Z, Li J (2020) Workforce Allocation in Motorcycle Transmission Assembly Lines: A Case Study on Modeling, Analysis, and Improvement, in *IEEE Robotics and Automation Letters*, Vol 5(3): 4164-4171.
- [17]. Theodor Freiheit, Wei Li (2017) The effect of work content imbalance and its interaction with scheduling method on sequential flowline performance, *International Journal of Production Research*, 55:10, 2791-2805.