Applied Mechanics and Materials *Online: 2013-08-30*
ISSN: 1662-7482, Vol. 376, pp 197-206
doi:10.4028/www.scientific.net/AMM.376.197

Solving Job Shop Scheduling Problem Based on Employee availability Constraint

G. Ramya[1, a] and M.Chandrasekaran[2,b]

[1]Research Scholar, Department of Mechanical Engineering, Sathyabama University, Chennai, Tamilnadu, India.

[2]Professor, Department of Mechanical Engineering, Vel's University, Chennai, Tamilnadu, India.

[a]ramyacadcam@gmail.com, [b]ch_sekaran@yahoo.com

Keywords: Job Shop Scheduling, Employee Workload, Shuffled Frog Leaping Algorithm and Sheep Flock Heredity Model Algorithm.

Abstract. Manufacturing System is enabled with an excellent knowledge on production plan, proper scheduling of machinery process, employee timetabling and labor costs. Heuristic algorithms are developed to bring optimized results in stipulated time with respect to optimum schedule. This article deals with minimizing the maximum completion time (makespan) based on job scheduling and minimization of labor costs based on employee workload with Shuffled Frog Leaping Algorithm and Sheep Flock Heredity Model Algorithm. The labor costs minimization and makespan which is to find a schedule that satisfies the organization's rules, employee's preferences, due date and customers. The formulation of assigning workload for employees is concerned with assigning number of employees into a given set of shifts over a fixed period of time and week task. The main problem attempts to minimize labor costs based on performance criteria and assigning the loads equally among all employees. Several local search methods and heuristics algorithms has been proposed in many research on Job shop scheduling. The Results are compared with other heuristics in terms of makespan, idle time and Labor costs the Shuffled Frog Leaping algorithm performs result oriented than other Heuristics Algorithm.

Introduction

A schedule is an allocation of tasks to the time intervals on the machines to find a schedule that minimizes the overall completion time, which is called the makespan. In the job shop scheduling problem n jobs have to be processed on m different machines. Each job consists of a several sequence of tasks that have to be processed during an uninterrupted time period of a fixed length on a given machine and has its own processing order. Due dates are treated as deadlines and require the job-shop scheduling to meet specific due dates in order to avoid delay penalties including customer's bad impression, cost of lost future sales and rush shipping cost.

In the last few decades many researchers have been focusing on solving various scheduling problems with different approaches by considering various objectives [1]. Many valid approaches and its advances are compared and shared between competitors in rapid form [2]. Mattfeld et a proposed randomly generated solutions with precedence relations which are not uniformly distributed [3]. Shmoys et al.proposed several poly-logarithmic approximations for evaluating an optimal schedule with makespan minimization criteria [4]. French predicted that no efficient algorithms will ever be developed for the majority of scheduling problems [5]. As a result, the focus of optimization research has turned to be enumerative approaches.

It has been recognized by many researchers that scheduling problems can be solved optimally using mathematical programming techniques and one of the most common forms of mathematical formulation for job shop scheduling problem was the Mixed Integer linear Programming (MIP) format of Manne [6]. Blazewicz et al. Emphasized the difficulties of JSP and indicated that mathematical programming models have not been achieved enough breakthroughs for scheduling problems [7]. Eusuff et al. proposed a new meta-heuristic algorithm called Shuffled Frog Leaping Algorithm for solving scheduling problems with discrete decision variables [8]. SFLA is a population-based cooperative search metaphor combining the benefits of the genetic-based memetic algorithm and the social behavior based particle swarm optimization Inspired by natural memetics [9]. Muzaffar Eusuff and Lansey described the algorithm is imitating the total sequence of modeling process and searching for best food with behavior of frogs placed on separate positioned stones in a pond [10] and also SFLA has been tested with a large number of combinatorial problems and found to be efficient in finding global solutions [11]. The SFLA comprises a set of interacting virtual population of frogs partitioned into different group's population memeplexes which are referred to searching for food [12]. The algorithm functions are simultaneously independent in local search of each memeplex [13]. SFHM algorithm was used for minimizing mean tardiness and mean flow time multi objective criteria [14-15]. An effective SFLA was used for minimizing maximum completion time (i.e., makespan) [16]. In terms of processing time and makespan the SFLA compares the results rapid favorably with the Sheep Flock Heredity Model Algorithm, Artificial Intelligence System, Genetic Algorithm, and Particle Swarm Optimization [17].

The application developed in this paper aims to realize an integrated system which has rapid response to changing customers requirements and capability to integrate heterogeneous manufacturing facilities [18]. The Employee time tabling philosophy is still employed by the majority of manufacturing enterprises for Job Shop Scheduling, process shift planning and production planning [19] and also to improve the production costs, minimizing labor costs, maximizing completion time and increase the employee satisfaction [20].

In this work SFLA and SFHM algorithm are used for solving the scheduling problem to meet due dates in a simple job shop. It is developed to approximately minimize the maximum completion time and in-process labor cost. Several benchmark problems are solved by the proposed algorithms and the results are compared with literature results.

This work focuses on two stages. First stage an objective function formulation was developed for minimizing the Employee workload and labor cost based on employee availability. The Second stage is refining the results of makespan with Shuffled frog leaping algorithm and Sheep Flock Heredity Model Algorithm. The paper describes how to integrate Employee workload and job shop scheduling with employee availability constraint and also this algorithm uses to refine makespan.

Overall System Architecture

In manufacturing systems, the decisions related to employee timetabling and scheduling jobs on the machines are often made in a sequential process. The objective of job scheduling is to find the optimum schedule to minimize the costs whereas the objective is to maximize employee satisfaction and to minimize labor costs. In many manufacturing industries employee workload assignment is first prepared and then the scheduling of jobs must take based on the resources and employee availability or first the scheduling of jobs is done and the employees workload allotment established based on the machine loads. However, the resulting problem has generally been considered as too complex to be used in practical situations. To develop a window based application which helps the organization to attain best procurement practices and supports the operation of procurement activity at the optimum total cost in the correct quality at the correct time and location for express gain by signing a contract. We propose to integrate the two problems by associating each job on machine and a set of activities performed by the employees. The system has been designed to store the data needed for the above mentioned scheme and meets all the required computations. Specifically this covers that the required job profile is not known in advance but is determined by the job schedule and the employee profile is determined by the selected employee schedules.

Job Shop Scheduling Problem. The Typical scheduling problems involve minimizing the maximum gj(t) value (the maximum cost problem) or minimizing the sum of gj(t) values (the total cost problem). Scheduling is defined as the art of assigning resources to tasks in order to insure the termination of these tasks in a reasonable amount of time. To meet an optimal objective solution or set of objectives these approaches are used for determination of the starting time and finishing time of processing of each part. Some other cases scheduling problem is addressed after the orders are released into the shop floor, along with their process plans and machine routings [21]. Scheduling plays a crucial role to increase the efficiency and productivity of the manufacturing system. The problem of scheduling is one of the operational issues to be addressed in the system on a daily or weekly basis and also static, dynamic penalty functions are rarely considered [22]. Job shop scheduling problems are Non-Polynomial (NP) hard] so it is difficult to find optimal solutions [23]

Employee Workload Problem. The workload of the employee has been found to be an effective way explicitly to consider relationships between the end items and the various processes and labors. The workload module systems determine the quantity of each labor that will be used in the production of a prescribed volume of final work, and the times at which each of them must be utilized to meet prescribed due dates for the final products. As a means for production scheduling, Employee workload systems leave a good deal to be desired and provides the means to make broad scheduling decisions. It does not encompass short term scheduling decisions like machine loading and operations sequencing. Once work load has set due dates for each stage, it becomes the responsibility of the shop floor scheduling system to meet such deadlines. This is a critical activity because the load on work centers changes over time. There can be such unexpected events as machine breakdowns, raw material shortage, scrap and rework, all causing the actual lead time to differ from the planned one. Moreover, computation does not take into consideration capacity constraints at the shop level, the choice between which job to process, and which one to delay, becomes crucial. Actual installed or available production capacity is ignored with the result that the schedules can prescribe machine loading in excess of 100% utilization. Production volumes and due dates must be adjusted manually to achieve feasible schedules. However, the main difference between tabling and finite scheduling is simply tries to schedule all activities required to meet a given master schedule while holding down work-in-progress inventory. If infeasibility occurs, production management must produce a new master timetable and production schedule to generate another plan or find alternative sources of production capacity. Finite scheduling is an optimization technique that tries to generate a sequence of operations over a given set of machines with the sole purpose of minimizing some type of shop performance measure like makespan, mean flow time, etc.

Employee Workload. Workload of each employee consists two modules. First, the input modules are activity list, activity attributes, milestone list, work load scope statement, preference list, consolidated list with all detailed profiles and organizational process assets. Second, the output modules are production schedule network and project document updates.

Timetable Life cycle. Timetable life cycle enhance the collection of logically related work activities usually culminating in the completion of a major deliverable. Collectively the time table's phases are known as Timetable life cycle. Framing employee timetabling is based on the Interpersonal skills, Preferences, Understanding the Product environment, General Knowledge skills, Negotiation, Leadership, Mentoring and Knowledge on application areas.

Minimization of Maximum Completion Time and Labor Costs

Makespan Minimization. The normally, manufacturing system consists active period starts from the first day of production on the machine with certain set of actions and operations. In general Meeting the due dates is the most important goal of scheduling to avoid the delay penalties including customer's bad impression, lost future sales. Due-date oriented functions, whereas the main aim of optimizing the makespan is to minimizing the labor costs and maximizing the output.

Labor Costs Minimization. During the production and process time the labor cost is considering with stipulated time based on the number of employees. Total accountability on every unit should me readily available in every set of job production. The total labor cost in every production should me equal to + or – 1 deviate from actual. When the production function starts the process management can be completed or to be completed with regular break even analysis by applying optimization methods to meet the regular bench mark of production cost management. Proper scheduling of machinery processes and operations are enabled in master production schedule in which production to be reached on climax within the stipulated time with an excellent knowledge on production engineer. The employee to avoid absenteeism is to bring out per capita per month analysis. One labors produces per day products worth of + or – k thousands. 16 x 3 Labors produces per day products worth 16 x 3 x 7 is equal to 336 k thousands (3.36 + or – L per day).

Problem Description

Common Time Representation for Employee Timetabling and Job Shop Scheduling [24]. We consider the following employee timetabling and job shop scheduling problem with single level jobs. Let T denote a time horizon with a set of elementary time periods t = 0, T = 1. E denote employees in organization comprising a set of employees E = {1, . . .,E} and M denotes set of machines M = {1, . . .,m}. Consider a non-pre-emptive job shop with m machines (M_i = i, …. m) & n jobs (N_i= i, …n). When j_i is the set of job to be processed on machine M_i . The operation sequence of the job j is denoted by O_{ij} (Where i^{th} operation on j^{th} machines M_j). Objective functions depend on due date which are associated with the jobs. A job consists of number of operations (O_{i1}, O_{i2} ,….O_{in}). There is set of activities A = {1, . . .,A} where each activity may be required by a job j and has to be performed by one or several employees. The organization has to process a set of n jobs J = {1, n} during the time horizon (T). Each job j has a release date rj and a due date dj .We assume that there is a production cost Wjt if job j starts at time t and an employee satisfaction cost Ceat if employee e is assigned to activity a at time t and A contains non working activities representing employee inactivity (break, lunch, etc.) gathered in set P.

Objective Function Formulation. The Eq (1) shows the objective of the problem is to minimize the labor cost subject to the following constraints [24]. Eq (2 to 5) represents exactly once the each job has to be started, All the started jobs finished within its time zone, each job can be processed by a machine at each time period with satisfaction of precedence constraint, each employees has to assign with each activity (At least one) at each time period and each employee has a specific constraints taking into account of minimum or maximum consecutive periods of work, and other complex regulation constraints. For instance, if no employee can work more than two consecutive shifts, the constraints of the form can be defined for each time period t = 1,T = 2 for each employee.

$$\min(f) = \sum_{j=1}^{n}\sum_{t=0}^{T-1} Wji\,Xji + \sum_{e=1}^{E}\sum_{a=1}^{A}\sum_{t=0}^{T-1} Ceat\,Yeat \tag{1}$$

Subject to

$$\sum_{t=0}^{T=1} Xji;\ \forall, \mathrm{Xjt} = \{1,0\} \tag{2}$$

$$\sum_{j=1}^{T=n1} Xji;\ \forall, \mathrm{Xjt} = 0;\ \mathrm{t} = \{rj, \ldots\ldots \mathrm{d}j - \mathrm{pj}\} \tag{3}$$

$$\sum_{j-1}^{n}\sum_{t=0}^{t-1} \mathrm{p}jt\,\mathrm{d}jtkjt;\ \forall, \mathrm{djt} = \{1 < k, k < p\} \tag{4}$$

$$\sum_{t=0}^{t=T} \mathrm{Y}\,eat = 1;\ \forall, \mathrm{t} = \{0, \ldots\ldots, T-1\} \tag{5}$$

$$\sum_{e=1}^{E} \geq \sum_{j=1}^{n}\sum_{a=1}^{A}\sum_{t=0}^{T-1} Weatj\,(Ceatj - Peatj - Reatj\,):\ \forall, \mathrm{t} = \{0, \ldots\ldots, T-2\}, \mathrm{Wj} > Pj = \{0,1\}\ \&\ Rj < n, t = 0, T-2 \tag{6}$$

For a given schedule (S), C*eatj* is the cost at which job j finished processing on machine i and Wj is the weighted time of job j spends in the queue before the first machine i. All ready times, processing times and due dates are assumed to be integer. In the above function n^{th} job is performed in i^{th} machine with j^{th} operation with unit time consideration for time P*eatj* and cost $j_{cost.}$ If the i^{th} machine is assigned with j^{th} operation for the first job is Xjt is 1,0. If the i^{th} machine is assigned with j^{th} operation for the k^{th} job is $P_{ij}^{(k)}$ is 1,0. Further, For Solving the above objective functions to find an optimum solution Heuristics method named SFLA and SFHMA has to be implemented and validated.

Proposed Methodologies

Shuffled Frog Leaping Algorithm. In this section, an SFLA for solving the JSS problem with minimizing total holding cost and makespan criterion are proposed by population initialization, partitioning scheme, memetic evolution process, shuffling process, and a local search. SFLA was combination of memetic Algorithm and Particle Swarm Optimization. It has been performed from memetic evolution of a group of frogs when seeking for food. The initial population of frogs was partitioned into groups or subsets called “memeplexes” and the number of frogs in each subset was equal.

At First, the SFLA is initially applied to different functions and to identify the fundamental weaknesses of this method as per the elimination of the effective frogs from memeplexes by solving procedure in consequence order. This method is similar to the SFLA, partitions particles into different groups called memeplexes and identified the best particle in each memeplex thereafter determines its movement through the search space in each iteration of the algorithm toward the global best particle and the worst particle in each memeplex keeps track of its coordinates in the solution space by moving toward the local best particle in the same memeplex.

The SFLA was follows two search techniques a) local search and b) global information exchange. Based on local search to reach the makespan, the frogs in each subset improve their positions to have more foods. After local search, obtained information based on Global information exchange between each subset was compared to other to produce best sequence way of schedule. Each operation is decided by meeting pre-specified due dates and minimizing objective function. Initial population of sequence generated randomly by increasing order and selected sequence divided into number of memeplexes.

Local Search Procedure. The division is done with the high level frog (column sequence) arranged in first memeplex, second one arranged in second memeplex, the last frog to the last memeplex and repeated frog back to the next order memeplex. Fitness function evaluated within the limits that the memeplex are infeasible.

Global information Exchange. The best frog memeplex values were identified each subset was compared to each other to produce best sequence way of schedule. For each iteration the frogs with the best fitness and worst fitness were identified and also the frog with the makespan schedule was identified. Finally, if the convergence criterion is not satisfied the position of the worst frog for the memeplex is adjusted and new subsets of memeplex will be created for the next iteration.

Sheep Flock Heredity Model Algorithm (SFHMA). Let us consider the several separated flocks of sheep in a field [25] as shown in Fig 1.

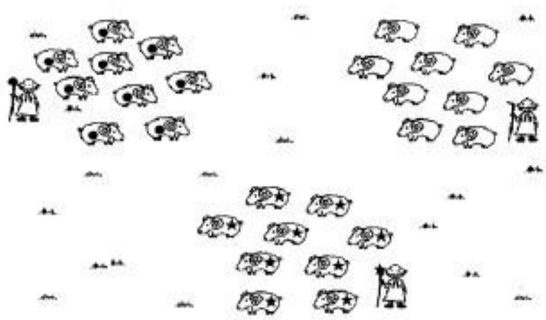

Fig 1. Flocks of sheep in the field

Fig 2. Mix of two flocks of sheep

Normally, sheep in an each flock are living within their own flock under the control of shepherds. The genetic inheritance only occurs within the flock group and the each sheep with high fitness characteristics to their environment breed in the flock. Let us assume that two sheep flocks were occasionally mixed in a moment when shepherds looked aside as shown in Fig 2. Then the certain moment, the shepherd of corresponding flock group runs into the mixed flock, and separates the sheep as before. However, shepherds cannot distinguish their sheep originally they owned because their appearance of all flock group of sheep are same and unique. Therefore, one flock from each sheep group is inevitably mixed with the other flocks in different group. The characteristics of the sheep in the neighboring flocks can be inherent to the sheep in other flocks in this occasion. The flock of the sheep, which has better fitness characteristics to the field environment, breeds most. In sheep flocks heredity model algorithm special string structure called hierarchical genetic operations like crossover level operations and mutation level operations are introduced. They are (1) sub-chromosome level genetic operation and (2) chromosome (global) level genetic operation.

Numerical Illustration

Shuffled Frog Leaping Algorithm.

Initiation, Population Creation, Mutation and Shuffling. Initial population of sequence generated randomly by increasing order and selected sequence divided into number of memeplexes for assignment tasks are shown in Fig.3. For each individual population i P calculate the fitness function f(i). Based on the fitness function calculate the size of each memeplex subsets and also randomly generate the population of the job sequence. The next step of operation sequences are grouped randomly. The processing hours & labor cost details are shown in Fig.4.In the mutation operation a memeplex subsets are generated using the mutation strategy to find the population P in descending order based on their fitness. Their corresponding machine sequences are shown in Fig.5.The trial sequence obtained by the crossover operation generation is compared with the target sequence to determine the jobs and machine schedule that participates in the next generation and the fittest is passed on to the next generation

Iterations. For each iteration process, the frogs with the best fitness and worst fitness were identified and also the frog with the makespan schedule was identified. Finally, if the convergence criteria are not satisfied the position of the worst frog for the memeplex is adjusted and new subsets of memeplex will be created for the next iteration. This procedure is repeated for desired number of iterations to reach optimal result. The first stage and last stage iteration results are shown in Fig. 6 and Fig.7.

Sheep Flock Heredity Model Algorithm.

Initiation, Population Creation, Mutation and Sub String selection. The initial sequence generated randomly. With a crossover probability a second and a third sub chromosomes are chosen randomly and crossover is performed. Probability for this chromosome is less than process mutation probability. Second and Third sub strings are selected to perform this process. Each sub string are chosen randomly to perform inverse mutation. Crossover probability this sting is less than the process crossover probability. Probability for this string is less than process mutation probability. The mutual sub string positions are randomly selected to perform inverse mutation.

Final result obtained using SFL and SFHM algorithm. The best solutions found in 100 iterations of the local search process, Global information exchange with SFL algorithm and Global level crossover process, Inverse mutation process with SFHM algorithm for minimizing labor costs and makespan are listed in Fig.8. These results are compared with genetic algorithm solutions which are obtained from genetic evolver with same processing time and processing hours. Based on employee availability constraint the labor cost (Lc) for each job was given below. $Lc^1_1 = Lc^1_2 = 7$, $Lc^1_i = 3$ $(i = 3, 4, \ldots, 8)$, $Lc^1_9 = Lc^1_{10} = 1$,

Initial Stage Iteration : 0	Task List	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
	Job Operation ID	11	12	13	14	15	21	22	23	24	25	31	32	33	34	35	41	42	43	44	45	51	52	53	54	55
	Job Sequence	1	1	1	1	1	2	2	2	2	2	3	3	3	3	3	4	4	4	4	4	5	5	5	5	5
	Machine Sequence	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
	Processing Time	64	66	31	85	44	7	69	68	14	18	74	70	60	1	90	54	45	98	76	13	80	45	10	15	91

Fig.3. Initial Operation Sequence and Processing Time of jobs

Example . 1 Processing Hours & Labor Cost Details													
PLANT - I													
Working Hours - Weak I									Total Hrs	Rate / Hr	Gross Amount	Others (Less)	Net Amount
Sl. No	NAME	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7					
1	Gokul NP	HD	12	12	12	12	Sun	12	48	16.67	800	180	620
2	Karthi P	HD	12	12	12	12	Sun	12	48	16.67	800	180	620
3	Muruga K	HD	12	12	12	12	Sun	12	48	16.67	800	180	620
4	Kumar T	HD	12	12	8	12	Sun	12	44	16.67	733	180	553
5	Prakash G	HD	12	12	12	12	Sun	12	48	16.67	800	180	620
6	Kavin P	HD	12	12	8	12	Sun	12	44	16.67	733	180	553
7	Prasanth A	HD	12	12	0	12	Sun	12	36	16.67	600	180	420
8	Prasanth B	HD	12	12	8	12	Sun	12	44	16.67	733	180	553
9	Murugan S	HD	12	12	8	12	Sun	12	44	16.67	733	180	553
10	Pradeep.A	HD	12	12	12	12	Sun	12	48	16.67	800	180	620
11	Ganapathy M	HD	14	12	8	12	Sun	12	46	16.67	767	180	587
12	Tamilselvan K	HD	12	12	12	12	Sun	12	48	16.67	800	180	620
13	Vaikka D	HD	12	12	12	12	Sun	12	48	16.67	800	180	620
14	Tamileniyan K	HD	12	12	12	12	Sun	12	48	16.67	800	180	620
15	Santhosh	HD	12	12	8	12	Sun	12	44	16.67	733	180	553
Daily Total Hrs		0	182	180	144	180	0	0	686	-	11436		8736

Fig.4 Processing Hours & Labour Cost Details

Machine 1																									
Start Time	0	74	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	81	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	524	588	n/a	n/a	n/a
Comp Time	74	81	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	161	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	588	642	n/a	n/a	n/a
Idle Time	0	0	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	363	0	n/a	n/a	n/a
Machine 1																									
Start Time	0	n/a	81	n/a	150	n/a	n/a	n/a	245	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	588	n/a	654
Comp Time	n/a	n/a	150	n/a	220	n/a	n/a	n/a	290	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	654	n/a	699
Idle Time	0	n/a	81	n/a	0	n/a	n/a	n/a	25	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	298	n/a	0
Machine 1																									
Start Time	0	n/a	n/a	74	n/a	n/a	134	n/a	n/a	n/a	n/a	n/a	232	n/a	n/a	380	n/a	448	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Comp Time	n/a	n/a	n/a	134	n/a	n/a	232	n/a	n/a	n/a	n/a	n/a	263	n/a	n/a	448	n/a	458	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Idle Time	0	n/a	n/a	74	n/a	n/a	0	n/a	n/a	n/a	n/a	n/a	0	n/a	n/a	117	n/a	0	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Machine 1																									
Start Time	0	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	290	n/a	n/a	366	380	n/a	395	n/a	480	n/a	n/a	n/a	n/a	n/a	n/a
Comp Time	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	366	n/a	n/a	380	395	n/a	480	n/a	481	n/a	n/a	n/a	n/a	n/a	n/a
Idle Time	0	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	290	n/a	n/a	0	0	n/a	0	n/a	0	n/a	n/a	n/a	n/a	n/a	n/a
Machine 1																									
Start Time	0	n/a	n/a	n/a	n/a	150	n/a	232	n/a	245	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	480	n/a	n/a	n/a	524	n/a
Comp Time	n/a	n/a	n/a	n/a	n/a	168	n/a	245	n/a	335	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	524	n/a	n/a	n/a	615	n/a
Idle Time	0	n/a	n/a	n/a	n/a	150	n/a	64	n/a	0	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	145	n/a	n/a	n/a	0	n/a

Fig.5 Machine Sequence after mutation

Task List	11	6	7	13	12	10	18	20	17	15	19	21	3	9	24	8	4	23	14	5	1	16	2	25	22
Job Operation ID	31	21	22	33	32	25	43	45	42	35	44	51	13	24	54	23	14	53	34	15	11	41	12	55	52
Job Sequence	3	2	2	3	3	2	4	4	4	3	4	5	1	2	5	2	1	5	3	1	1	4	1	5	5
Machine Sequence	1	1	2	3	2	5	3	5	2	5	4	1	3	4	4	3	4	3	4	5	1	1	2	5	2
Processing Time	74	7	69	60	70	18	98	13	45	90	76	80	31	14	15	68	85	10	1	44	64	54	66	91	45

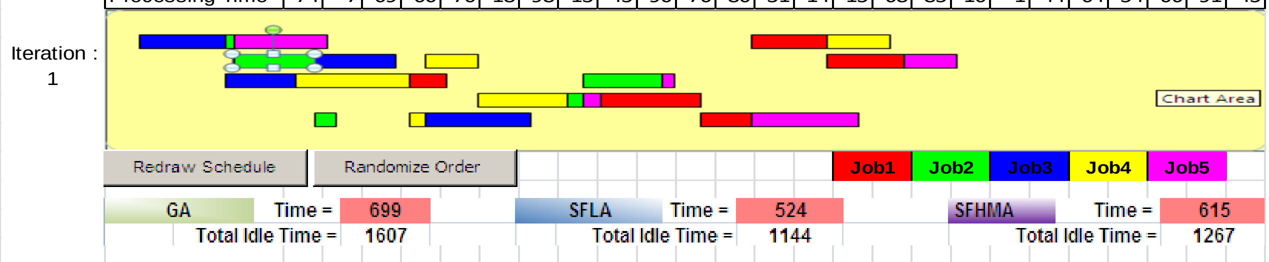

Fig. 6 First Iteration

Task List	4	19	2	14	5	22	15	10	7	18	24	1	3	8	6	12	16	17	25	23	11	20	21	13	9
Job Operation ID	14	44	12	34	15	52	35	25	22	43	54	11	13	23	21	32	41	42	55	53	31	45	51	33	24
Job Sequence	1	4	1	3	1	5	3	2	2	4	5	1	1	2	2	3	4	4	5	5	3	4	5	3	2
Machine Sequence	4	4	2	4	5	2	5	5	2	3	4	1	3	3	1	2	1	2	5	3	1	5	1	3	4
Processing Time	85	76	66	1	44	45	90	18	69	98	15	64	31	68	7	70	54	45	91	10	74	13	80	60	14

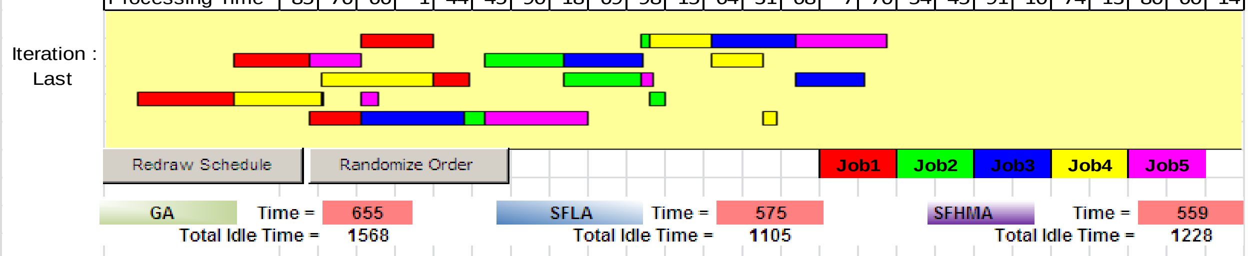

Fig. 7 Last Iteration

Iteration	GA			SFLA			SFHMA		
	Makespan	Idle Time	Labor cost	Makespan	Idle Time	Labor cost	Makespan	Idle Time	Labor cost
Iteration:1	699	1607	11427	524	1144	11425	615	1267	11433
Iteration:2	693	1604	11411	536	1139	11409	607	1254	11412
Iteration:3	687	1601	11395	548	1134	11393	599	1241	11391
Iteration:4	681	1598	11379	560	1129	11377	591	1228	11370
Iteration:5	675	1595	11363	572	1124	11361	583	1215	11349
Iteration:6	669	1592	11347	584	1119	11345	575	1202	11328
Last Iteration	655	1568	11267	575	1105	11254	559	1228	11281

Fig.8 Final results obtained after 100 iterations

Conclusion

To avoid customer's bad impression and to improve the customers Satisfaction by delivering the jobs within the due date is a very important criteria in manufacturing system. In order to avoid delay penalties including customer's bad impression, cost of lost future sales and rush shipping cost, due date constraints are considered. The objective considered in this paper is minimizing makespan and labor cost with Shuffled frog leaping algorithm and Sheep Flock Heredity Model Algorithm. Strict due date parameter, assigning employee load based on processing time, machine availability and loose due date parameter are used for analyzing the labor cost. The proposed heuristics are used for testing evolver based genetic solver problems. Results shows that the proposed algorithm produces good quality results compared with other Heuristics approach procedures.

References

[1] Yarong Chen, Zailin Guan & Xinyu shao, " A Comparative Analysis of job Shop Scheduling Algorithms" MSIE (2011) – IEEE transactions 978-1-4244-8385, Pg. No. 1091-1095

[2] Ahmed S. Ghoniem, "Static and dynamic job-shop scheduling using rollinghorizon approaches and the Shifting Bottleneck Procedure "Thesis, Master of Science In Industrial and Systems Engineering (2002)

[3] Mattfeld, D.C., Bierwirth, C., and Kopfer, H., (1998), "A Search Space Analysis of the Job Shop Scheduling Problem", to appear in Annals of Operations Research.

[4] Shmoys, D. B., Stein, C., and Wein, J., (1994), "Improved Approximation Algorithms for Job shop Scheduling Problems", SIAM Journal of Computer science, June, 23(3), pp. 617-632.

[5] French, S., (1982), "Sequencing and Scheduling - An Introduction to the Mathematics of the Job- Shop", Ellis Horwood, John-Wiley & Sons, New York.

[6] Manne, A. S., (1960), "On the Job-Shop Scheduling Problem", Operations Research, vol 8, pp. 219-223.

[7] Blazewicz, J., Dror, M., and Weglarz, J., (1991), "Mathematical Programming Formulations for Machine Scheduling: A Survey", European Journal of Operational Research, 51(3), pp. 283-300

[8] Eusuff M, Lansey K, Pasha F (2006) Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization. Eng Optim 38:129–154

[9] Mohammadreza Farahani, Saber Bayat Movahhed, Seyyed Farid Ghaderi1 , A hybrid meta-heuristic optimization algorithm based on SFLA. 2nd International Conference on Engineering Optimization September 6 - 9, 2010, Lisbon, Portugal

[10] M. Eusuff, K. E. Lansey, Optimization of water distribution network design using the shuffled frog leaping algorithm, Journal of Water Resource Planning and Management, vol. 129, issue 3, 210-225,2003

[11] Emad Elbeltagi , Tarek Hegazyz and Donald Griersonmad (2007), “A modified shuffled frog- leaping optimization algorithm applications to project management” Structure and Infrastructure Engineering, Vol. 3, No. 1, March 2007, 53 – 60

[12] Quan-Ke Pan & Ling Wang & Liang Gao & Junqing Li, An effective shuffled frog-leaping algorithm for lot-streaming flow shop scheduling problem.Int J Adv Manufacturing Technology, DOI 10.1007/s00170-010-2775-3

[13] M. Alinia Ahandani, N. Pourqorban Shirjoposht & R. Banimahd , Job-Shop Scheduling Using Hybrid Shuffled Frog Leaping, online

[14] M.Chandrasekaran, P.Asokan, S.Kumanan, ” Multi objective optimization for job shop scheduling problems using Sheep Flocks Heredity Modal Algorithm”, Int J Mfg Sci and Tech, Vol 9, 2:47-54, 2007.

[15] Chandramouli, Anandaramam “ An Improved Sheep Flock Heredity Algorithm for Job Shop Scheduling”, International Journal of Industrial Engineering Computation 2 (2011), Pg. Np. 749-764

[16] M.Eusuff, K.Lansey, F.Pasha, ”Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization”. Eng Optima 38:129–154, 2006.

[17] Emad Elbeltagia,, Tarek Hegazy and Donald Grierson, Comparison among five evolutionary-based optimization algorithms, Advanced Engineering Infomatics 19 (2005),43-53

[18] Amnon Meisels and Andrea Schaerf , “ Modelling and solving employee timetabling problems”, Annals of Mathematics and Artificial Intelligence 39: 41–59, Kluwer Academic Publishers. Printed in the Netherlands, 2003.

[19] Christian Artigues et. Al. “ Solving an Integrated employee timetabling and Job shop scheduling problem via hybrid branch and bound “ Computers & Operations Research (2009), Pg. No. 2330 –2340.

[20] A.T. Ernst et al. ” Staff scheduling and roistering: A review of applications, methods and models”, European Journal of Operational Research, 153 3–27, 2004.

[21] Elanz Baghal, Azardoost, “ A hybrid algorithm for multi objective flexible Job Shop Scheduling Problem” Proceedings of the 2011 International conference on Industrial Engg. & operational Management.

[22] Liang Sun et. al,” Solving Job Shop Scheduling problem using genetic algorithm with penalty function”, International Journal of Intelligent information processing, Vol. no. 2, Dec- 2010

[23] Ren Qing et. al, “ An Improved genetic algorithm for Job Shop Scheduling Problem” International Conference publications on Computational Intelligence Security (2010), Pg. No. 113 – 116

[24] Christian Artigues et. Al., “A Flexible Model and a Hybrid Exact Method for Integrated Employee Timetabling and Production Scheduling”, Author manuscript, published in "Practice and Theory of Automated Timetabling VI, Brno Czech Republic (2006)" DOI : 10.1007/978-3-540-77345-0_5

[25] Koichi Nara, Tomomi Takeyama & Hyurchul kim,” A New Evolutionary Algorithm based on sheep flock heredity model & its application to scheduling problem , IEEE Transactions, Volume No. 1, Pg. No. 503-508 (1999)