Simultaneous scheduling of machines and tools in multimachine flexible manufacturing system with alternate machines using metaheuristic algorithms

M Maruthi Prasad[1], K Prahlada Rao[2]

1 Research Scholar, JNTUA, Ananthapuramu, AP, India

2 Professor, Dept. of Mechanical, JNTUCEA, Ananthapuramu. AP, India

Scheduling jobs and tools is a significant problem for manufacturing systems. Inefficient job scheduling on the tool loading may results in under utilization of capital-intensive machines and the high-level machine idle time. Therefore, efficient scheduling of jobs and tools enables a manufacturing system to increase their machine's utilization and decrease their idle time.This article deals with simultaneous scheduling of machines and tools in a multimachine flexible manufacturing system togenerate best optimal sequences that minimizes makespan with alternate machines and a copy of every tool type. Tools are stored and supplied from central tool magazine to the required machine to perform the operations. Scheduling of flexible manufacturing systems is a well-known NP-hard problem which is very complex, due to additional considerations like material handling, alternative routing, and alternate machines. This paper proposes nonlinear mixed integer programming (MIP) formulation to model this simultaneous scheduling problem and Black widow optimization & Jaya algorithms are used to solve this problem.

The outcomes show that BWO algorithm produces better results than Jaya algorithm for simultaneous scheduling of machines and tools with alternate machines.

Keywords Flexible manufacturing system, Simultaneous scheduling, Makespan, Alternate machines, BWO algorithm, Jaya algorithm

1.Introduction

The flexible manufacturing system (FMS) is indeed an integrated manufacturing system that consists of numerous facilities such as computer numerically controlled (CNC) machines, automated guided vehicles (AGVs), automated storage/retrieval systems (AS/RSS), central tool magazines (CTM), tool transporters (TT), robots, and automated inspectional controlled by a central computer (Agnetis et al., 1997; Baker, 1974). Numerous subsystems' flexibilities are combined to provide an overall flexibility in the FMS. FMS is a relatively new technology in industrial automation, and various academics have been drawn to it during the previous three decades. FMS has a number of advantages, including increased productivity, lower work-in-process inventory, increased machine utilization, supervision-free production, increased product variety, and excellent quality to fulfil customer expectations. Jobsetting time was almost decreased by using fixtures, pallets, TT, and CTM (Jerald and Asokan, 2006). The major advantage of a flexible manufacturing system is its capacity for great flexibility in the management of production facilities and resources. These systems are most often used in smaller lot manufacturing, wherein their efficiency is comparable with that of mass manufacturing. The disadvantage is the heavy installation cost (Peter Kostal, 2011). Flexible manufacturing systems (FMS) may be roughly divided into four groups: single flexible machines (SFM), flexible manufacturing cells (FMCs), multi-machine FMS (MMFMS), and multi-cell FMS (MCFMS).

Scheduling is the practice of allocating resources over time in order to accomplish a set of tasks. It is a vital decision-making process for the majority of industrial and service industries to succeed.A well-designed schedule helps the industry to make the most use of its resources and achieve the strategic goals outlined through its production plan. FMS manages a variety of scheduling options, including such machine assignment for jobs and tool selection, to improve efficiency and flexibility. Appropriate scheduling is crucial in FMS.(Gamila andMotavalli 2003).

2.Literature Review

Scheduling in flexible manufacturing systems is a very well NP-hard issue that is very complicated owing to the inclusion of material handling, alternate routes, and different equipment. A flexible manufacturing system's performance may be improved by effective resource usage, effective integration, and synchronization of resource scheduling. While significant research has been conducted separately on machine scheduling and tool scheduling, the two challenges are in reality inextricably linked. Recently, more emphasis has been placed on the combined impact of machine and tool scheduling.

Giffleret al [1] devised an enumerative approach for generating all active schedules for "n" jobs and "m" machines issues.

P.Chandra et al.[2] described a procedure for measuring the optimal job and tool sequences which significantly reduce overall fixture and tool changing time in order to make sure jobs are completed ahead of schedule in an environment which includes a machine, tools, and a collection of jobs, which each needs a machine operation.

Jain et al [3] proposed an algorithm for dealing with unforeseen machine failures, increased order priority, rush order arrivals, and order cancellations that modifies just those operations which must be rescheduled and used in conjunction with existing scheduling methods to enhance the productivity of flexible manufacturing systems.

Prabhaharan et al [4] tried an article on the topic of unified operation and tool scheduling in a "FMC" that consists of "M-indistinguishable work cells" and a CTM. They suggested the "Simulated Annealing Algorithm and Priority Dispatching Rule Algorithms" to reduce the time required to create the "Unified Job and Tool Scheduling".

Udhaykumaret al [5] suggested an Ant Colony Optimization technique for determining the optimal work and tool arrangement that would allow schedules to be coordinated for the shortest possible make-span in a Flexible Manufacturing System.

P.Udhaya Kumar et al. [6] suggested Non-Traditional Optimization Algorithms like PSO, SA, ACO and Genetic Algorithm to provide an optimal configuration of "Job and Tool" that coordinates the least lateness for FMS, and it has been mentioned that the PSO Technique gives extra workable alternatives with a much more perceptive computing effort.

J.Aldrin Raj et al.[7] implemented concurrent scheduling of machines and tools to seek the optimal optimal sequences by utilizing four heuristics: primary concern dispatching rules, a customized non-delay schedule generation algorithm with six different rules, a customized Giffer and Thompson algorithm, as well as an AIS algorithm, with the aim of reaching the shortest possible make span in such a multi-machine flexible manufacturing system.

A.Costa et al.[8] created a hybrid genetic algorithmem that integrates a local search improvement scheme for scheduling "n" tasks on "m" parallel machines that are susceptible to tool change procedures due to tool wear, with the objective of reducing overall completion time.

N. Sivarami Reddy et al. [9] presented concurrent scheduling of machines and tools to select the highest optimal sequences through the use of three heuristics, such as the Crow search algorithm, the Symbiotic organisms search algorithm, and the Flower pollination algorithm, with the aim of reaching the shortest possible make span inside a multi-machine flexible manufacturing system. The FPA was found to offer the greatest outcomes.

Mehrabadet al [10] proposed a tabu search technique for resolving the flexible job shop scheduling issue with the objective of minimizing the makespan time.

In this study, a novel metaheuristic search method called BWO is employed to decrease makespan by scheduling operations and tools concurrently.

3 Problem Formulation

CTM is often incorporated in FMS to aid in the storage of tools. The required tool is pooled with other machines or transported from the CTM to another machine via a tool transporter during the job's machining, which reduces the number of tools in the CTM and hence tooling costs.The next parts define the issue, its assumptions, and restrictions.

3.1. Problem definition

Considering the processing times for 'n' tasks {J1,J2,........Jn} that must be processed by'm' machines {M1,M2,.....,Mm} and need tools as from CTM with 't' tools {T1, T2,.....,Tt}. The optimal sequence that minimizes the time required to complete a task by combining the selection of jobs, machines, and tools is to be found using heuristic processes. BWO is employed in this study to build optimum schedules with minimum makespan as the target. The same set of issues that were previously analyzed using the techniques described(xx) in is evaluated, and the findings are compared to those previously obtained.

The approach used in this work is shown via the use of an example problem. A work set 5 has the tasks, machinery, and tools indicated in Table 1. Suppose the task set 5 has five tasks, the very first three among which need three operations and last two of which need just two. The system is said to be composed of four machines & four tools in total. M1-T4 (6), for example, indicates that job 1 'J1' requires machine M1, tool T4, and a six-unit processing time.Other jobs are assigned in accordance with the notation described above. It is essential to provide a task sequence that minimizes the makespan while taking machine and tool restrictions into account. During the scheduling process, a choice must be made on the machine and tool to be used for each project. Both the machine and the CTM will have a backlog of requests from unfinished tasks. A task that is compatible with the request must be chosen to minimize the makespan. As a result, a series of operations is created that reduces the overall amount of time spent. This is referred to as a single jobset. Similarly, in our investigation, 22 job sets were produced at random with same processing time.

Table 1. job set 5

Jobs	Operation I	Operation II	Operation III
Job 1 (J1)	M1-T4 (6)	M2-T1 (12)	M4-T4 (9)
Job 2 (J2)	M1-T2 (18)	M3-T1 (6)	M2-T3 (15)
Job 3 (J3)	M3-T3 (9)	M4-T4 (3)	M1-T2 (12)
Job 4 (J4)	M4-T2 (6)	M2-T3 (15)	-
Job 5 (J5)	M3-T2 (3)	M1-T1 (9)	-

3.2. FMS environment

Four machines, a CTM having four tools, and automated tool changer (ATC), 1 automated pallet changer (APC), & 2 automated guided vehicles (AGVs) compose the FMS environment under examination. A loading and unloading facility is located on one side.Each machine center contains buffer storage for completed jobs. An automated storage and retrieval system (AS/RS) is used to store and retrieve raw materials. The whole system, composed of several components, is depicted in Figure 1.

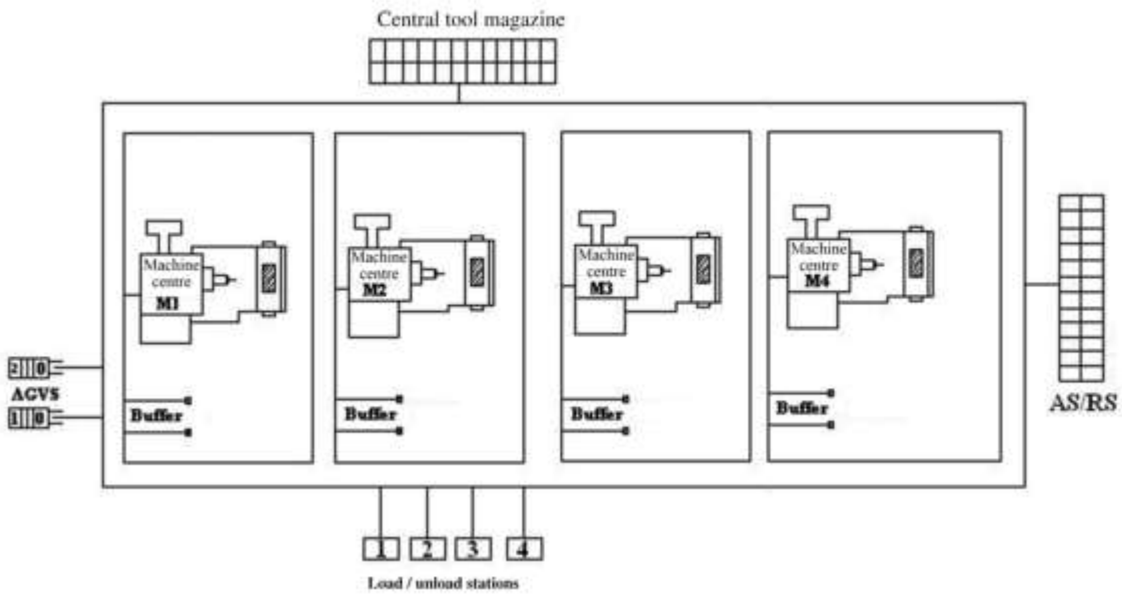

Figure 1. FMS environment.

3.3. Assumptions and constraints

The following assumptions are made for the present work being carried out.

1. Jobs are self-contained and comprise strictly ordered sequences of operations; no task or activity is prioritized.
2. Job anticipatory behavior is not permitted.
3. A given operation may be carried out by analternate machine.
4. At time zero, all jobs are accessible concurrently.
5. Each operation has a defined scope of work and duration (cost).
6. The system is aware of the future availability of machines, i.e., the system does not assume 100% machine availability in its initial form.
7. After an operation is completed on one machine, it is quickly transferred to the next machine, with little traveling time.
8. The setup times for operations are irrespective of their sequence of execution and are included in processing timings.
9. CTM stores tools.
10. The tool transporter facilitates the movement of tools across the system.
11. The system's machines share tools.

The constraints on the problem are listed below.

• There are precedence restrictions, which mean that for each job, a set of pre-specified operations sequences will be in place that cannot be modified. Consider the 3132 operation.

3—Job No. 1—J3's first operation

3—The first operation of J3 is carried out on machine 3.

2—The first functioning of J3 needs the use of tool 2.

J3's second operation can be executed only after the first operation has been completed, and so operation 32XX cannot be executed prior to 31XX. This is referred to as a precedence restriction.

The proposed metaheuristic search algorithm that minimizes makespan by simultaneously scheduling jobs and resources with alternate machines while ignoring tool transfer times, are described in the next part.

4. Model formulation

In this section, a nonlinear MIP model is introduced to clearly specify the crucial parameters and their effect on the FMS scheduling problem.

4.1 Notations

Subscripts

j index for a job

i, h,r,s indices for operations

k index for a tool

g alternate machine index

Parameters and sets:

J	:	job set on hand for processing
n_j	:	operations in job j
N	:	$\sum_{j \in J} n_j$, total operations in job set J.
I	:	{1,2,-------N}, index set for operations
I_j	:	$\{J_j+1, J_j+2, \text{------} J_j+n_j\}$, the indices' set in I linked with job j, where J_j is jobs' operations listed before job j and $J_1=0$
IF_i	:	$I-\{h; h \geq i, i, h \in I_j\}$operations' index set without operation i and same job's following operations t to operation i.
IP_h	:	$I-\{i; i \leq h, i, h \in I_j\}$operations' index set without operation h and same job's preceding operations t to operation i.
TL	:	{ }, the set of tool types to carry out the jobs' operations
R_k	:	{ }, indices set in I linked with tool type k in TL $k \in TL$
u	:	first operation that uses tool k, $u \in R_k, k \in TL$
v	:	preceding operation of i, $i, v \in R_k, k \in TL$
RT_{kj}	:	I_j I R_k is the index set of operations in I common for tool k and job j for all k,j, $k \in TL,\ j \in J$
a_j	:	job j ready time
b_k	:	tool k ready time
$R_{(g)}$	:	ready time of machine indicated by $g, \forall g = 1,2.....$
R_i	:	ready time of machine for operation i

AMi : {(x,y)/x: alternate machine, y: processing time for operation i on that machine.}, the set of combination of a alternate machines and processing times on those machines for operation i, $\forall i, \quad i \in I \cap I_j \quad , \quad \forall j, \quad j \in J$

AMi(g): operation i processing time on the machine indicated by g $\forall g = 1,2...... \forall i = 1,2.....n$

S : (max($R_{(g)}$, b_k) + $AM_i(g)$), a set of selected machines for operations associated in set I

m_i : selected machine for operation i $\forall i = 1,2,3,.....n$

MS : {m3,m1,m4,m2,------} a set of selected machines for operations associated in set I

pt_i : operation i processing time on selected machine from available alternate machines.

ct_i : operation i completion time on selected machine from available alternate machines.

Decision variables

$$q_{rs} = \begin{cases} 1 & \text{if } ct_r \text{ is less than } ct_s, \text{where } r \text{ and } s \text{ are operations of different jobs} \\ 0 & \text{otherwise} \end{cases}$$

$$\beta_{rs} = \begin{cases} 0 & \text{if } ct_r \text{ is less than } ct_s, \text{where } r \text{ and } s \text{ are operations of different jobs}, r, s \in R_k \\ 1 & \text{otherwise} \end{cases}$$

4.2 Mathematical model

In the formulation, machine and tool indices are not employed specificallybecause routing for each job is determined from the given multiple routes as per equation (5), selected machines in the route are assigned to a set MS, and the machine and tool indices are known for each operation index in *I*. For the operations that belong to the same tool and different jobs, tool need not follow precedence constraints of jobs and for operations that belong to the same tool and same job, tool needs to follow precedence constraints of jobs. The optimization problem is cast as minimization of objective function. The objective function, Z, may be conveniently chosen as the maximum completion time among all jobs' last operations i,e MSN. The objective function for minimization of MSN is

$$Z = \min(\max(ct_i)) \quad for \ \forall \ i, \ i \in I$$

Subject to the following constraints

$Z \ge ct_{N_j+n_j} \ \forall j, j \in J$ The constraint 1 ensures that MSN is greater than or equal to the completion time of last operation of all the jobs	(1)
$ct_i - ct_{i-1} \ge pt_i \quad for \ \forall \ i-1, i \in I_j, \ j \in J \ , i \in R_k, \ k \in TL$ The constraint set 2a is the operations' precedence constraints.	(2a)
$ct_{N_j+1} \ge pt_{N_j+1} \ for \ \forall \ j \ and \ j \in J \ , N_j + 1 \in R_k, k \in TL$ The constraint set 2b is the constraint for the completion time of first operations of jobs.	(2b)
$\left.\begin{array}{l} ct_r \ge ct_s + pt_r - H\, q_{rs} \\ ct_s \ge ct_r + pt_s - H(1 - q_{rs}) \end{array}\right\}$ $\forall r, \ r \in I_j, s \in I_l \ where \ j, l \in J, \ j \langle l$	(3)

$(1+H\ \beta_{rs})ct_r \geq ct_s + pt_r - H\,q_{rs}$ $(1+H\ \beta_{rs})ct_s \geq ct_r + pt_s - H(1-q_{rs})$ $\forall r,\ r \in I_j, s \in I_l\ \ where\ j,l \in J,\ j \langle l$ H is a large positive integer in the constraint sets 3 and 4 which ensures that no two operations allocated to the same machine or the tool can be concurrently performed.	(4)
max(R_i, b_k) $\leq$ ct_i - pt_i, $\forall i,\ and\ i \in I$ The 5th constraint specifies that operation i can begin only after the maximum the machine ready time and the tool ready time.	(5)
The machine for an operation i from alternate machines is selected such that operation i is finished at the earliest as given in (5). $MS = \{m_i / S\ is\ \min.\ \forall g = 1,2..\}\quad \forall i = 1,2,3,.....N$	(6)

$ct_i \geq 0,\ \forall i,\ \ i \in I$

,

$q_{rs} = 0,1\ \ \forall r,\ \ \ r \in I_j, s \in I_l\ where\ j,l \in J,\ j \langle l$

$\beta_{rs} = 0,1\ \ \forall r,\ \ \ r \in I_j, s \in I_l\ where\ j,l \in J,\ j \langle l, r, s \in R_k$

The process times are nonnegative and the decision variables take values of 0 and 1.However, this formulation is intractable due to its size and nonlinearity, thus meta-heuristic algorithm namely CSA is used to obtain near optimal or optimal solutions.

Since the MSN needs to be minimized, calculation of MSN for a given schedule needs to be developed. Flow-chart for such a computation is shown in figure 2.

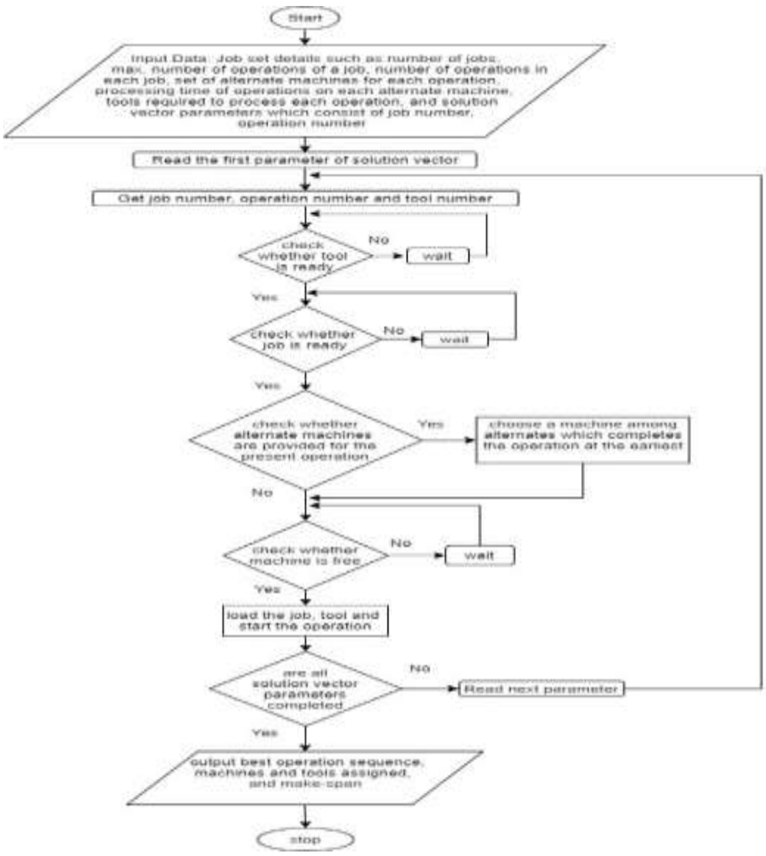

Fig 5. Flow chart for MSN calculation

Figure 2. Flow chart for calculation of makespan with alternate machines and a copy of each tool type

4.3. Input data

Twenty two different job sets are considered for simulation since total jobs, total operations and machines in every job set differ from one to the other. The same mathematical model and the BWO & Jaya algorithms are implemented for each of the twenty two job sets. Such a study establishes that the mathematical model and the BWO & Jaya algorithms are independent of the job sets. Ten Job sets used by M.V. Satish kumar et al are employed for this problem, which are also the job sets employed by Bilge and Ulusoy but with the additional information such as the alternate machines and processing times on those machines to carry out the operations. Remaining 12 job sets are developed and tested. The tools that are used to process the operations of jobs in above job sets are the same tools specified in the job sets employed by Aldrin raj et al which are also standard problems provided by Bilge and Ulusoy but with the additional information such as tools to carry out the operations. The job sets are given in Appendix A. Every job set with 5 to 8 different jobs, each operation of a job is to be processed on alternate machines. Each job's entity in the job set offers information about the alternate machines, operation processing time of a job on those machines and tool needed for the operation. The following data is offered as an input.

(i) Number of jobs, each job's operations and job's maximum operations in job set.
(ii) Alternate machines needed for each job-operation (Alternate machine matrix),
(iii) Time needed to carry out every job-operation on each alternate machine(process time matrix),
(iv) Tool to perform every job-operation (Tool matrix)

5. Metaheuristic algorithms

5.1.Black Widow Optimization Algorithm

Due to the simplicity and flexibility of a novel metaheuristic optimization method based on the mating behaviour of black widow spiders, which was initially suggested by V. Hayyolalam and A. PourhajiKazem in 2020, it has been utilized to tackle a variety of engineering and scientific challenges. The Black Widow Optimization Algorithm (BWO) is influenced by the black widow spider's distinctive mating behaviour. This approach contains a step that is unique to it, called cannibalism. As a result of this phase, species with insufficient fitness are excluded from the circle, resulting in an early convergence. BWO is capable of balancing exploitation and exploration. In other words, it is capable of inspecting a vast region in order to find the optimal global solution; thus, BWO is an excellent option for tackling many optimization issues involving several local optima.

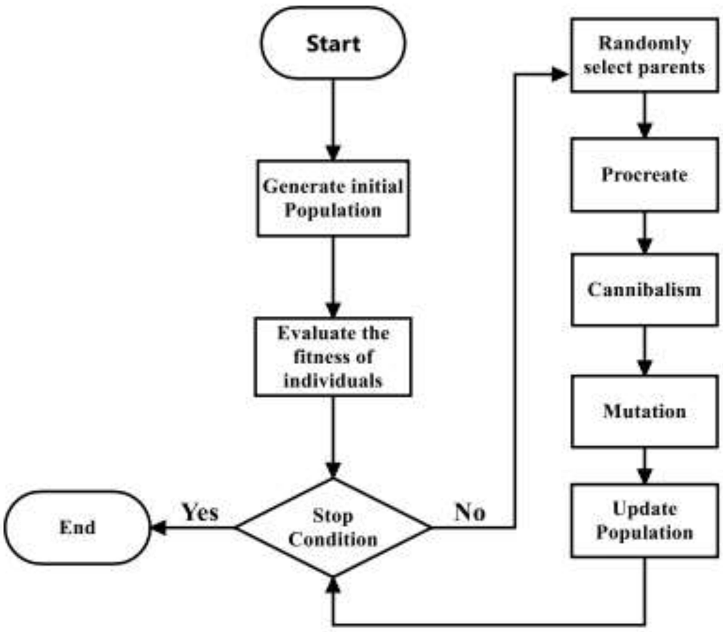

Figure 3: BWO flowchart

5.1.1.Steps in BWO Algorithm

5.1.1.1. Initial population

The variables required to solve an optimization issue are referred to as " widow" in this technique, and the possible solution to each issue is referred to as a black widow spider. Each Black widow spider demonstrates the values of the issue variables. To solve benchmark functions in this work, the structure should be treated as an array. In a Nvar dimensional problem a widow is an array of 1X N_{var}representing the solution of the problem. The widow's fitness is determined by evaluating the fitness function f at the widow.

$$Fitness= f(widow)$$

To begin the optimization process, an initial population of spiders is used to construct a candidate widow matrix of size N_{pop} XN_{var}. Next, at random, pairs of parents are chosen to undertake the procreative stage of mating, wherein the female black widow consumes the male black widow.

5.1.1.2. Procreate

This stage results in the reproduction of a new generation via the mating process of pairs of autonomous widows inside its web. Now, in order for this algorithm to reproduce, an array named alpha must also be generated as long as the widow array contains random numbers; after which, offspring are generated using the following equation, wherein the x_1 and x_2 are now the parents and y_1 and y_2 are indeed the offspring.

$$\begin{Bmatrix} y_1 = \alpha \; X \; x_1 + (1-\alpha) \; X \; x_2 \\ y_2 = \alpha \; X \; x_2 + (1-\alpha) \; X \; x_1 \end{Bmatrix} \quad \ldots\ldots\ldots\ldots(1)$$

This procedure is done $N_{var}/2$ times, with the exception that randomly chosen integers must not be duplicated. Eventually, the offspring and mother are added to an array and ordered as per their fitness value, which is now determined by the cannibalism rating; a subset of the best individuals gets added to the newly created population. These procedures are universally applicable to all pairs.

5.1.1.3. Cannibalism

The female black widow consumes her partner during or after mating in the first case of cannibalism (sexual). Female and male will be distinguished in this algorithm based on their fitness scores. Second cannibalism (sibling) occurs when powerful spiderlings consume their weaker siblings. The algorithm will establish a cannibalism rating (CR) based on which the number of survivors will be decided. In certain instances, a third kind of cannibalism is often witnessed, in which newborn spiders consume their mothers. The fitness value is being used to classify spider lings as strong or weak.

5.1.1.4. Mutation

At around this point, we arbitrarily choose Mutepop from the population. As seen in Figure 2, each one of the selected solutions swaps two members in the array at random. The mutation rate is used to compute the mutation population.

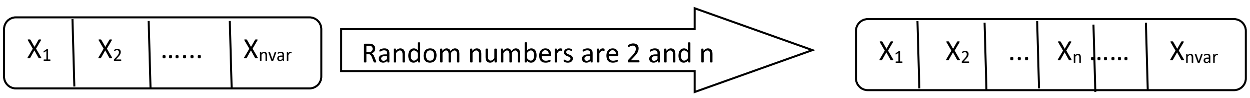

Figure2: Mutation rate

5.1.1.5. Convergence

Three stop conditions, similar to those used by other evolutionary algorithms, may be considered: (a) a predetermined iteration count. (B) Maintaining a constant fitness value for the best widow for a number of iterations. (C) Attaining the necessary degree of precision. The first-stop condition is employed in the current work.

5.1.1.6. Parameter setting

Parameter setting is crucial to maximize the algorithm's performance in finding better solutions. The more parameters are tuned, the greater the possibility of leaping out of any local optimum and the greater the opportunity to explore the search space worldwide. Thus, the appropriate number of factors may guarantee that the balance amongst exploitation and exploration phases is maintained. The BWO algorithm incorporates three critical control parameters, namely Procreate rate (PP), Cannibalism rate (CR) and Mutation rate (PM) with the values 0.6, 0.44 and 0.4 respectively.

5.2. Jaya Algorithm

A simple yet powerful optimization algorithm is proposed in this paper for solving theconstrained and unconstrained optimization problems. This algorithm is based on the concept that the solution obtained for a given problem should move towards the best solution and shouldavoid the worst solution. Jaya algorithm is a powerful parameter less algorithm for finding optimal solutions. In this algorithm also only common controlparameter is required like TLBO algorithm that is population size and number of generation size. JAYA algorithm always tries toimprovise the solution to avoid the worst solution. JAYA algorithm has only one phase and it is relatively simpler to use for anyspecific problems. First of all, needs to initialize the population size, iteration number which is a common control parameter, thenidentify the best and worst solution for a given population. The important step of the algorithm is a modification of the solution byevaluating the best and worst result for the particular objective function and problem defined for minimization or maximization. r_1 andr_2 are any two random numbers in between 0 and 1.

5.2.1. Flow chart

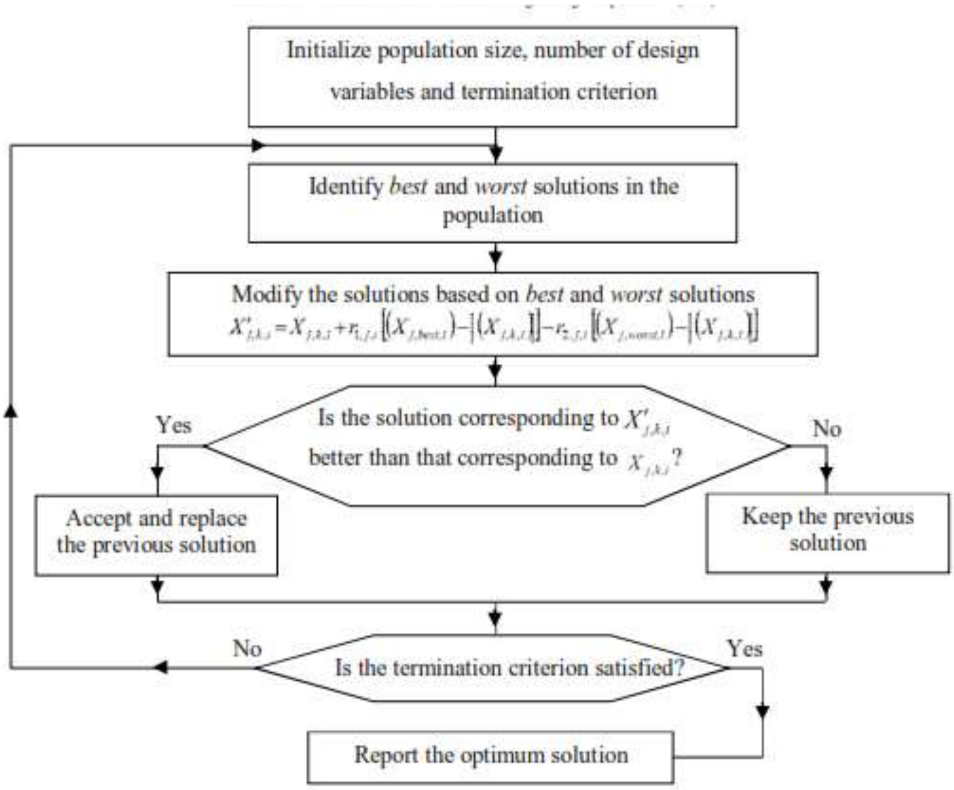

Figure4: Jaya algorithm flow chart

6. Results and discussion

Proposed Black widow optimization and Jaya algorithms discussed areemployed to accomplish Makespan optimization in FMS for simultaneous job and machine scheduling with alternate machines.Totally 22 jobsets are considered in the work and Table 2 represents the job set information consisting of number of jobs and their details of operation. The step-by-step procedure for evaluating each solutionvector of the simultaneous scheduling problem withalternate machines is shown in the form of a flow chart inFig. 5.

Table. 2

Job Set	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
No. of Jobs	5	6	6	5	5	6	8	6	5	6	7	8	5	9	10	8	7	9	5	10	15	20
Min no. of operations in a job	3	3	4	5	3	3	3	4	4	4	3	3	5	3	3	3	3	3	5	3	8	8
Max no. of operations in a job	13	15	16	19	13	18	19	20	17	21	18	19	19	21	22	19	17	20	19	24	110	151
No. of Machines	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	6	8
No. of tools	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	6	8

The objective function developed by us is tested on different benchmark problems provided by Aldrin Raj et al.,2014 and the makespans obtained by proposed methods are tabulated. From Table 3 it is observed that BWO yielded better makespan for simultaneous scheduling of jobs and tools with alternate machines. Makespans obtained by the proposed methods is same for all jobsets except for jobsets 6,17, 21 & 22.From table 3, for problems with more operations BWO is outperforming Jaya algorithm.The makespan obtained by proposed methods forall 22 job sets is illustrated as a bar chart in fig. 5. Figure 6 shows the Gantt chart for the sequence generated for job set 1 by BWO, in which the operations that are assigned to each machine as well as the start and finish times of each operation and utilization of tools for various operations of jobs is shown. The Gantt chart shows the correctness of the solution provided by the proposed BWO algorithm.

Table 3: Best makespan obtained by Jaya and BWO algorithms along with mean runs and standard deviation

Job Set	Jaya algorithm			BWO		
	Best Makespan	Mean runs	Standard deviation	Best Makespan	Mean runs	Standard deviation
1	53	55.00	1.9365	53	53.24	0.8307
2	54	56.88	1.8330	54	54.00	0
3	70	71.04	1.5937	70	70.00	0
4	54	54.56	0.7681	54	54.00	0
5	42	42.96	0.5385	42	42.24	0.4359
6	86	90.68	2.0761	85	85.12	0.4397
7	62	62.16	0.3742	62	62.00	0
8	84	85.20	0.9574	84	84.40	0.6455
9	95	97.68	1.2490	95	95.08	0.2769
10	107	111.80	1.9149	107	107.04	0.2000
11	81	81.00	0	81	81.00	0
12	48	50.64	1.2207	48	48.24	0.6633
13	93	95.44	1.4457	93	93.00	0
14	66	66.12	0.3317	66	66.00	0
15	91	91.00	0	91	91.00	0
16	58	60.60	1.4720	58	58.08	0.2769
17	46	47.04	0.7895	45	45.12	0.3317
18	50	52.48	0.9626	50	50.00	0
19	64	67.12	1.7635	64	64.00	0
20	75	75.60	0.7638	75	75.00	0
21	225	234.56	4.7617	192	195.04	4.7739
22	286	292.20	3.8514	235	236.64	1.9553

When makespan of both the algorithms in the above table is compared, BWO is outperforming Jaya algorithm for the jobs which have more operations. The standard deviation is zero for 11 problems in case of BWO and for 2 problems in case of Jaya algorithm. If one looks closer to the final solution of 100 runs for these problems, one finds that distinct solutions with the same makespan value exist. Therefore, BWO finds many optima alternatives than Jaya algorithm.

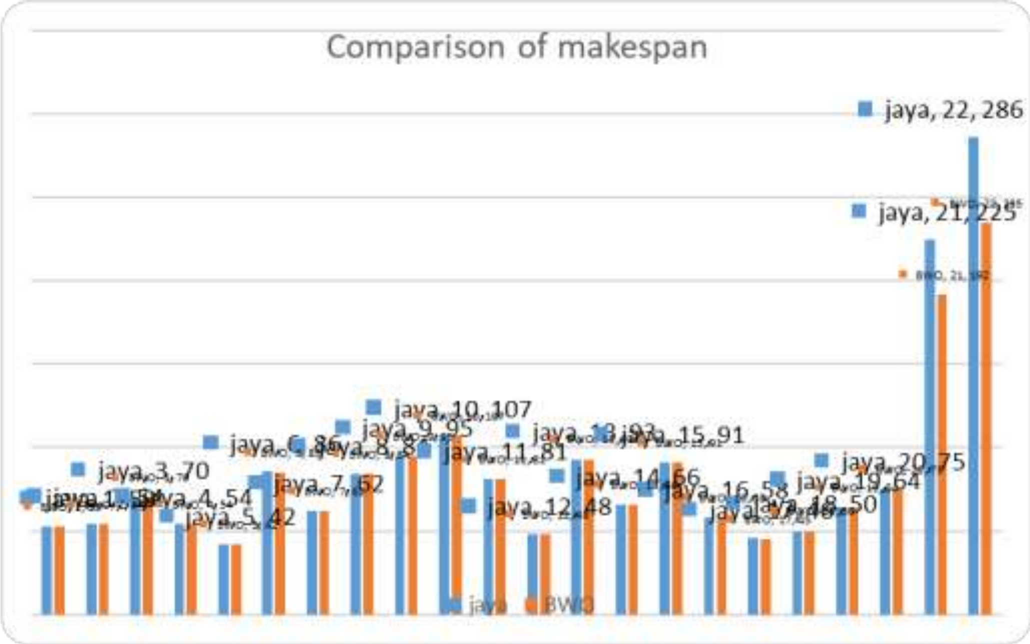

Figure 4: Makespan comparison obtained by proposed methods

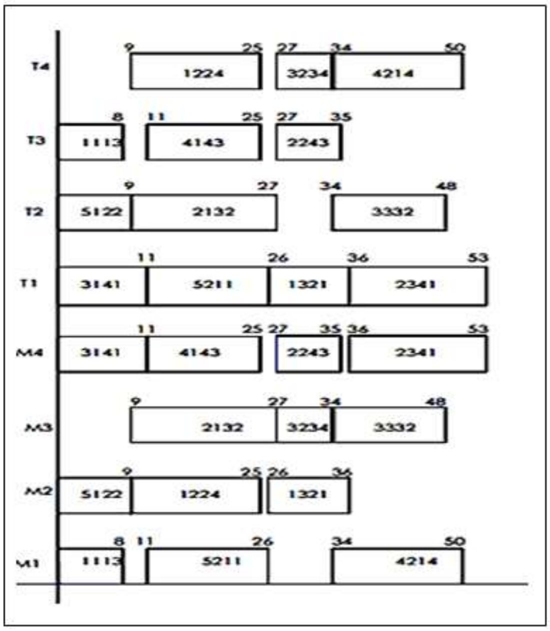

Figure 5: Gantt chart for job set 1 by Jaya algorithm

7. CONCLUSI ON

The proposed algorithms are used to schedule jobs, tools, and alternate machines withmakespan as an objective. To demonstrate accuracy, the suggested algorithms are checked on 22 job sets.Makespan obtained by BWO for job set 21 and job set 22 is less compared with Makespan obtained by Jaya algorithm. From table 3, it is found that standard deviation values obtained by BWO are small compared with Jaya algorithm.Therefore, BWO is outperforming Jaya algorithm.

REFERENCES

[1] Giffler, B. and Thompson, G.L. (1960), “Algorithms for solving production scheduling problems”, Operations Research, 8(4), 487-503.

[2] Chandra, P. and S. LI &M. STAN (2007), “ Jobs and tool sequencing in an automated manufacturing environment” 2911-2925

[3] A. K. Jain &H.A. Elmaraghy (2010), “ Production scheduling/rescheduling in flexible manufacturing” 281-309

[4] Prabaharan, T., Nakkeeran, P.R. and Jawahar, N. (2006) ‘Sequencing and scheduling of job andtool in flexible manufacturing cell’, *International Journal of Advanced ManufacturingTechnology*, Vol. 29, No. 3, pp.729–745.

[5] Udhayakumar, P. and Kumanan, S. (2010) ‘Sequencing and scheduling of job and tool in flexible manufacturing system using ant colony optimization algorithm’, International Journal of Advanced Manufacturing Technology, Vol. 50, No. 9, pp.1075–1084.

[6] Udhayakumar, P. and Kumanan(2011) ,”Integrated scheduling of flexible manufacturing system u8sing evolutionary algorithms” International Journal of Advanced Manufacturing Technology, Vol. 61, pp.1621–635.

[7] Aldrin Raj, J., Ravindran, D., Saravanan, M. and Prabaharan, T. (2014) ‘Simultaneous scheduling of machines and tools in multimachine flexible manufacturing system using artificial immune system algorithm’, International Journal of Computer Integrated Manufacturing, Vol. 27, No. 5, pp.401–414

[8]Costa, A., Cappadonna and Fichera,S(2016).“Minimizing the total completion time on a parallel machine system with tool changes”, Computers & Industrial Engineering, 91, 290–301.

[9] Sivarami Reddy, N., Ramamurthy,D.V., PrahladaRao,K (2017) “.Simultaneous scheduling of machines and tools to mini mi se makespan in multi machine FMS using new nature inspired algorithms”, Manufacturing Technology Today, Vol. 16, No. 3, pp.19–27

[10]Nasr N, Elsayed EA (1990) Job shop scheduling with alternative machines. Int J Prod Res 28(9):1595–1609

[11] JeekyoungAhn, Weihua He and Andrew Kusiak(1993), “Scheduling with Alternative Operations”,IEEE Transactions on Robotics and Automation, 9:3 pp.297-303.

[12] KIM K H, EGBELU P J. Scheduling in a production environment with multiple process plans per job [J]. International Journal ofProduction Research, 1999, 37(12): 2725-2753.

[13] Choi I C, Choi D S. A local search algorithm for job shop scheduling problems with alternative operations and sequence-dependent setups [J]. Computers & Industrial Engineering, 2002, 42(1): 43-58.

[14] MEHRABAD M S, FATTAHI P. Flexible job shop scheduling with tabu search algorithms [J]. International Journal of AdvancedManufacturing Technology, 2007, 32(5/6): 563-570.

[15][Satish Kumar, M.V., Ranga Janardhana& Rao, C.S.P(2011)., “Simultaneous scheduling of machines and vehicles in an FMS environment with alternative routing” International Journal of Advanced Manufacturing Technology, 2011, 53:339–351