SEGMENT RANDOM INSERTION PERTURBATION SCHEME (SRIPS) GENETIC ALGORITHM FOR MANUFACTURING INDUSTRY

Dr. Nageswara Rao.M[1*], V.V.Rajan Raju[2], K.Sarath Phani[3], M.B.N.Murthy [4]
[1] *Department of Mechanical Engineering, KLEF, Vaddeswaram, Guntur, AP, India*
[2,3,4] *Final Year Students Department of Mechanical Engineering, KLEF, Vaddeswaram, Guntur, AP, India*

Abstract: This paper lays down a formal framework for simultaneous scheduling of machines- automated guided vehicles (AGVs) and tools in a multi-machine flexible manufacturing system (FMS) while accounting for transport times of parts to minimise makespan. To minimize tooling costs- a central tool magazine (CTM) is suggested so that the tools are 'shared'. AGVs and tool transporter (TT) carry jobs and tools between machines. The complexity of including sequencing of job operations on machines- assignment of AGVs and tools to job operations and corresponding trip operations such as the empty trip and loaded trip times of AGVs and a CTM in scheduling is greater. The scope of this paper is to propose a nonlinear Mixed Integer Programming (MIP) model to minimize makespan. Since the problem is known to be NP hard- it is conjectured and then verified that the intelligent behaviour of chromosomes and genes can be effectively used to lay down a metaheuristic algorithm known as a segment random insertion perturbation scheme genetic algorithm (SRIPSGA) suitable for the problem at hand- and the results have been tabulated and analyzed.

Keywords: FMS- segment random insertion perturbation scheme- makespan- AGVs- Priority rules- Vehicle Scheduling- local search

Introduction

Customer satisfaction is an important challenge in the latest manufacturing scenario- to face which- organizations have shifted their emphasis from producing large quantities of single product to a diverse mix of products with shorter life cycles. Meeting the dead lines is another crucial goal in today‟ s globally competitive manufacturing environment. To meet these challenges the organization should have a variety of flexibilities. In parallel with the developments in computers and automation technology over the past few decades- a new type of production system called the flexible manufacturing system (FMS) has evolved FMS is especially suitable for medium and low-volume industries. The philosophy of FMS is the right answer for unpredictable market environments that demand low-cost solutions for quickly and effectively adapting changes in product mix- demand- and designs (Viswanadham and Narahari- 1994). One of the important aspects in FMS operation is its scheduling policy- which plays a vital role in effective utilization of resources like machines and automated material handling system.

2. **Literature Survey**

To address the complexity of the FMS scheduling problem- Stecke (1985) divided the FMS operation problem into preproduction setup and production operation. Set-up phase includes loading the tools- allocating the operation to the machines- allocating the pallets and fixtures to the different part types. All the issues to be handled next to setup phase are included in the production operation and he is emphasized on pre-production setup of the FMS.Kusiak (1986) presented an FMS scheduling system which uses a rule-based expert system. This system followed priority rules to schedule jobs normally- but when a job cannot be scheduled because of resource conflicts- decision tables were used to select alternative machines- tools- fixtures- material handlers. Chang et al. (1989) - report on a heuristics based beam search technique designed to solve the random FMS scheduling problem. They measured the flexibility of the manufacturing system by flexibility index and for various values of the flexibility indices they compared their algorithm against several dispatching rules. Their algorithm gave better results than the dispatching rules but at the cost of increased computational effort. Pundit and Palekar (1990) proposed branch-and-bound as well as heuristic solution procedures for the simultaneous scheduling of machines and material handling vehicles in a job-shop environment Biegel et al. (1990) applied GA to the job shop-scheduling problem and discussed the GA process for an elementary "n" tasks one-machine problem. Ulusoy et al. (1993) - pointed out to make scheduling of AGVs an integral part of the overall scheduling activity in an FMS. They proposed an iterative solution procedure to generate schedules for machines and AGVs simultaneously Chen et al. (1995) proposed a GA based heuristic for the flow shop problem on the makespan objective and compared the efficiency of the proposed GA with the other GA heuristics reported in the literature. Bilge and Ulsoy (1995) addressed the simultaneous scheduling of machines and material handling problem in an FMS environment by time window approach. **They considered identical automated guided vehicles for material**

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

transfer- which are not allowed to return to load/unload station after each delivery. Cheng et al. (1996) reviewed the research works on classical JSP using genetic algorithms. They divided their review work into two parts. In Part I- they focused their attention on the representation schemes proposed for JSP- where as in Part II- they discussed various hybrid approaches of genetic algorithms and conventional heuristics. Al-Hakim (2001) applied Genetic algorithms for solving job shop scheduling problems. He developed a new coding scheme based on grouping different levels of operations. Abdelmaguid et al. (2004) developed a hybrid GA/heuristic approach and attempted the simultaneous scheduling problem of machines and AGVs. Reddy and Rao (2006) addressed the simultaneous scheduling problem with single as well as multi-objective performance criteria including makespan- mean flow time and mean tardiness. They developed a hybrid GA- to solve the FMS scheduling problem. Jerald et al. (2006) addressed the problem of simultaneous scheduling of parts and AGVs for a particular type of FMS environment by using a non-traditional optimization technique called the adaptive genetic algorithm (AGA). Gonçalves et al. (2002) have developed a hybrid genetic algorithm for the job shop scheduling problem. The chromosome representation of the problem was based on random keys. Aytug et al. (2003) thoroughly reviewed the literature on application of genetic algorithms for production and operations management problems. Nearchou (2004) investigated the effect of various genetic operators on the performance of GAs when applied on permutation flow-shop scheduling problems..

3. Scheduling Problem

Simultaneous scheduling problems in FMS with four layout configurations as shown in Fig.1 and ten job sets are used (Bilge and Ulsoy, 1995). The AGV travel times and the machine allocation and operation times for the jobs are given in Appendix A

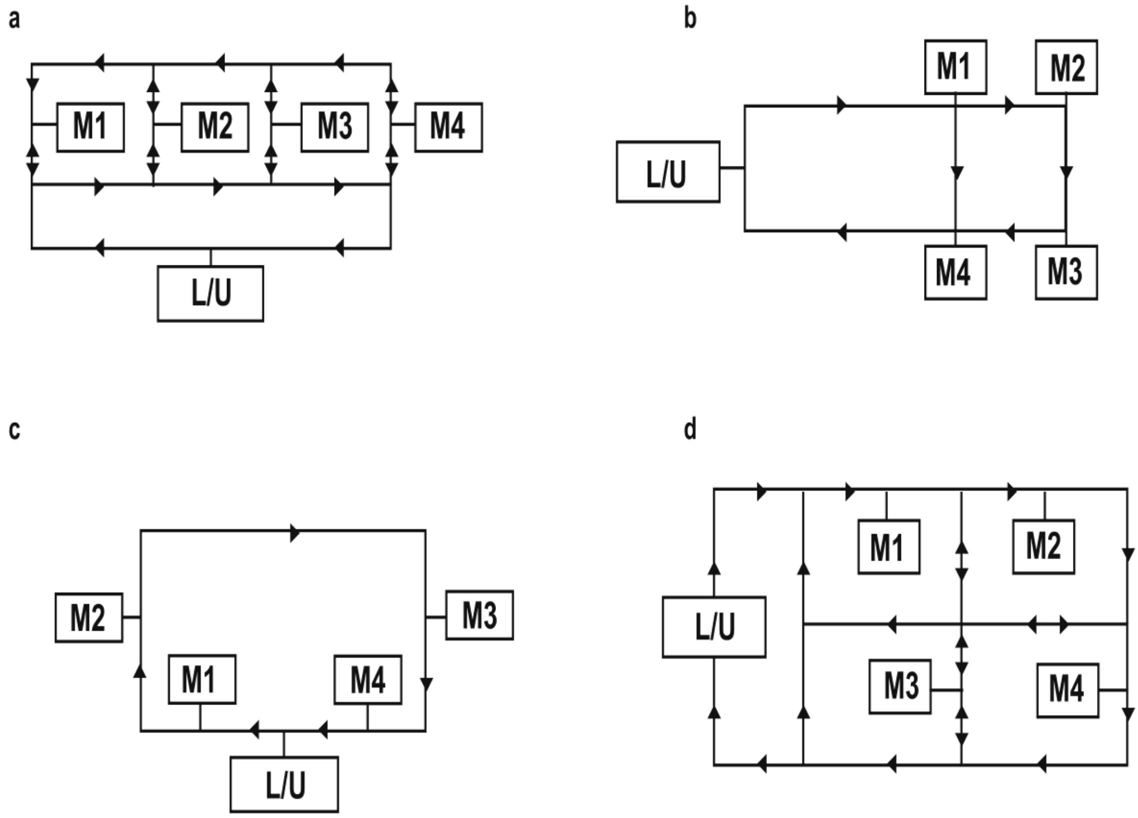

Figure: 1 Layout configurations used for examples

4. Simultaneous Scheduling of machines and AGVs in FMS

Jobs are scheduled based on the operation sequence derived by the algorithms. The problem considered needs scheduling of material handling system along with that of machines. In this paper SRIPSGA are used to solve simultaneous scheduling problems which are discussed below

The steps involved in SRIPSGA are given below:

Step 1: Genetic representation
Step 2: Initial Population
Step 3: Evaluation Function
Step 4: Reproduction selection Scheme
Step 5: Genetic Operators
Step 6: Incorporate a local search in to the GA
Step 7: Receptor Editing
Step 8: Termination criterion
Step 9: Genetic Algorithm parametric setting

4.1 Simultaneous Scheduling through SRIPSGA

For implementation of SRIPSGA- Job set 5 and Layout 2 are considered as an example. SRIPSGA computes the process times for different jobs and the sequences are obtained based

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

on the random manner.

The SRIPSGA is explained in the following steps for the job set 5:

4.1.1 Genetic Representation

Proper representation plays a vital role in successful implementation of any evolutionary algorithm. Here operations based coding (Gen and Cheng- 1997) is used to represent genes in a particular chromosome- where every gene is represented by a number. Hence the chromosome consists of as many numbers of genes as the total number of operations in that job set.

Job Number	1			2			3			4		5	
Operation number	1	2	3	1	2	3	1	2	3	1	2	1	2
Gene Code	1	2	3	4	5	6	7	8	9	10	11	12	13
Machine number	M1	M2	M4	M1	M3	M2	M3	M4	M1	M4	M2	M3	M1
Processing time	8	16	12	20	10	18	12	8	15	14	18	10	15

4.1.2 Initial Population

Double the number of operations is used in this paper- i.e. a list of jobs is itself taken as an operations. For example- if in a flow shop scenario there are 5 jobs with 13 operations it means 26 chromosomes as initial population (double the number of operations)

Chromosome number	Chromosome
1	4- 10- 12- 1- 7- 2- 13- 5- 11- 8- 3- 6- 9
2	4- 10- 12- 1- 7- 2- 8- 13- 11- 5- 9- 6- 3
3	10- 12- 1- 7- 4- 13- 11- 2- 8- 5- 9- 6- 3
4	4- 10- 12- 7- 1- 11- 13- 2- 8- 5- 6- 3- 9
5	4- 12- 1- 7- 10- 2- 5- 11- 13- 8- 9- 6- 3
6	1- 10- 7- 4- 12- 8- 2- 11- 13- 5- 3- 6- 9
7	7- 12- 1- 4- 10- 13- 2- 8- 11- 5- 6- 9- 3
8	7- 10- 12- 4- 1- 13- 2- 5- 11- 8- 3- 9- 6
9	1- 4- 10- 12- 7- 8- 13- 2- 5- 11- 3- 9- 6
10	12- 10- 1- 7- 4- 8- 13- 2- 5- 11- 9- 6- 3
11	10- 7- 12- 4- 1- 11- 8- 13- 5- 2- 9- 3- 6
12	1- 4- 12- 10- 7- 2- 13- 11- 8- 5- 6- 3- 9
13	7- 4- 10- 12- 1- 5- 11- 2- 13- 8- 6- 3- 9
14	7- 10- 1- 4- 12- 8- 5- 11- 13- 2- 3- 6- 9
15	4- 1- 7- 12- 10- 5- 13- 2- 11- 8- 6- 3- 9
16	7- 1- 10- 12- 4- 11- 13- 8- 5- 2- 6- 9- 3

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

17	4- 12- 7- 10- 1- 8- 13- 11- 2- 5- 6- 3- 9
18	12- 4- 10- 7- 1- 11- 5- 8- 13- 2- 9- 3- 6
19	12- 7- 10- 1- 4- 2- 5- 8- 13- 11- 9- 3- 6
20	7- 12- 10- 4- 1- 8- 11- 2- 13- 5- 3- 6- 9
21	12- 1- 7- 10- 4- 5- 2- 13- 8- 11- 9- 3- 6
22	10- 4- 12- 7- 1- 2- 8- 5- 13- 11- 9- 3- 6
23	12- 10- 4- 7- 1- 2- 11- 13- 8- 5- 9- 3- 6
24	7- 4- 12- 1- 10- 5- 8- 13- 11- 2- 3- 6- 9
25	1- 4- 10- 12- 7- 13- 11- 8- 2- 5- 3- 6- 9
26	4- 7- 1- 12- 10- 13- 11- 8- 2- 5- 3- 9- 6

4.1.3 Evaluation Function

For implementation of SRIPSGA Algorithm- job set 5 and layout 2 are considered as an example. From the above population job order operation sequence '1' is

4- 10- 12- 1- 7- 2- 13- 5- 11- 8- 3- 6- 9

For identifying the maximum operational completion time of the above sequence- the steps discussed in below are executed.

Step 1: Considering the machine number (M.No) of the given sequence for the job

M1-M4-M3-M1-M3- M2-M1-M3-M2-M4-M4-M2-M1

Step 2: The AGV '1' is selected

Step 3: The vehicle's previous location (VPL) is identified

For example considering first operation VPL=L/U

Step 4: The previous operation machine number (POMN) is identified as

POMN=L/U

Step 5: The vehicle ready time (VRT) is identified as VRT=0

Step 6: The previous operation completion time (POCT) is found to be '0'

Step 7: Vehicle empty trip time (VET) is calculated with

VET = VRT + TRT1=0+0=0

Where TRT1 = VPL to POMN= L/U to L/U= 0

Step 8: The maximum vehicle empty travel time is found from

Max (VET) = Maximum (POCT and VET)= Max(0-0)=0

Step 9: The total travel time of vehicle (TT) is evaluated from

TT=VET+ Time taken from previous machine to latest machine.

= 0 + L/U to M1 (from travel time data for layout '2')

= 0 + 2(half of the travel time) = 2

Step 10: Found machine ready time (MRT) from

MRT = Time until the job is completed on the assigned job operation = 0

Step 11: Identified the maximum of TT and MRT from

Maximum travel time of AGV = Maximum (TT-MRT)

= Max (3- 0) = 3

Step 12: The maximum travel time is added to the process time to get the operational completion time (OCT) or makespan.

OCT = Maximum TT+ Process Time = 3 + 36 (Process time double) = 27

Step 13: Repeated the steps from 4 to 14 for all other operations.

Step 14: Identified the maximum operational completion time. It represents the possible completion time (makespan) of given job set.

The calculated values of various parameters for all operations are shown in table 1

Table 1: Completion time through SRIPSGA (for Problem set 5 and layout-2)

O.No	M.No	V.No	VPL	POMN	VRT	POCT	VET	Max(7,8)	VLT	MRT	Max(10,11)	Process Time	Make Span

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)
4	M1	1	L/U	L/U	0	0	0	0	2	0	2	18	38
10	M4	2	L/U	L/U	0	0	0	0	3	0	3	6	15
12	M3	2	4	L/U	3	0	5	5	9	0	9	3	15
1	M1	1	1	L/U	2	0	5	5	7	38	38	6	50
7	M3	1	1	L/U	7	0	10	10	14	15	15	9	33
2	M2	2	3	1	9	50	14	50	51	0	51	12	75
13	M1	1	3	3	14	15	14	15	20	50	50	9	68
5	M3	1	1	1	20	38	20	38	40	33	40	6	52
11	M2	1	3	4	40	15	41	41	46	75	75	15	105
8	M4	1	2	3	46	33	47	47	48	15	48	3	54
3	M4	2	2	2	51	75	51	75	77	54	77	9	95
6	M2	1	4	3	48	52	54	54	60	105	105	15	135
9	M1	1	2	4	60	54	62	62	66	68	68	12	92

Table 1 shows operation scheduling of through SRIPSGA rule for job set 5 layout 2 is shown. From the vehicle heuristic algorithm for first two operations AGVs are selected sequentially in case of third operation AGV ‘2’ is selected basing on the availability of AGV with minimum travel time this constraint will be taking care in the algorithm. For job set 5 and layout 2 the operational completion time (makespan) is 135

In similar way make span for all 26 sequences are calculate and identify the best sequences based on evaluation function for the reproduction selection scheme

4.1.4 Reproduction Selection Scheme

Receptor Editing Genetic Algorithm uses tournament selection procedure is used in this work- which picks randomly “Chromosomes and the winner (based on their fitness) among them is selected for next operation.

4.1.5 Genetic Operators

In order to alter the genetic alignment and to reproduce new chromosomes in every generation genetic operation like crossover and mutation are performed.

4.1.6 Crossover operators

The crossover operator is an important component of GA. The crossover operation generates offspring from randomly selected pairs of individuals within the mating pool- by exchanging segments of the chromosome strings from the parents. Different types of crossover operators are available in the literature and here in this work- single point crossover and two-point crossover are considered.

i) Single point crossover:

In single point crossover randomly a cut point is selected and one side of the genes of the cut point in the first parent is exchanged with the genes of second parent on the same side. Chromosomes before single point crossover:

4- 10- 12- 1- 7-| 2- 13- 5- 11- 8- 3- 6- 9
10- 12- 1- 7- 4-| 13- 11- 2- 8- 5- 9- 6- 3

Chromosomes after single point crossover:

4- 10- 12- 1- 7- 2- | 11- 2- 8- 5- 9- 6- 3
10- 12- 1- 7- 4- 13 |**- 13- 5- 11- 8- 3- 6- 9**

As it can be seen the off springs produced after crossover may violate the precedence constraints and also some genes may be missing whereas others are duplicated. In the present case operations 2 is duplicated whereas operations 13 is missing in offspring 1. The reverse is true in offspring 2 i.e.- 13 duplicated and 2 is missing. To take care of these problems repair and replace functions are used. Repair function exchanges the genes to yield valid off-springs whereas replace function removes the duplicate genes and replaces them with missing genes.

Chromosomes after single point crossover replace and repair:

4- 10- 12- 1- 7- 2- 11- 13- 8- 5- 9- 6- 3
10- 12- 1- 7- 4- 13- **13- 5- 11- 8- 3- 6- 9**

ii) Two-point crossover:

As the name indicates- in this case two cut points are selected randomly and the genes in between these cut points are exchanged.

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

Chromosomes before two-point crossover:
4- 10- 12- 1 | 7- 2- 11- 13 | 8- 5- 9- 6- 3
10- 12- 1- 7 | 4- 13- 13- 5 | 11- 8- 3- 6- 9

Similar to single point crossover here also the problems like violation of precedence constraints- duplication and missing of genes may arise which needs repair and replace.

Chromosomes after two-point crossover- replace and repair
4- 10- 12- 1- 7- 2- 13- 8- 11- 5- 9- 6- 3
10- 12- 1- 7- 4- 2- 11- 13- 8- 5- 3- 6- 9

4.1.7 Mutation operators

For exploring the search space effectively- now mutation operation is performed on the selected chromosomes. Different mutation operators are available in the literature and among them random mutation- inverse mutation- adjacent mutation and shift mutation are used in this work and the performance of the GA for different mutations in combination with the above two crossover operators is studied.

Random Mutation

In this mutation two genes are randomly selected and their positions are exchanged. If necessary- chromosome is repaired.

Chromosome before random mutation:
10- 12- 1- 7- 4- 2- 11- 13- 8- 5- 3- 6- 9
Chromosome after random mutation:
10- 12- 1- 7- 13- 2- 11- 4- 8- 5- 3- 6- 9

Adjacent Mutation

In this case two genes which are adjacent to each other are exchanged.

Chromosome before adjacent mutation:
10- 12- 1- 7- 13- 2- 11- 4- 8- 5- 3- 6- 9
Chromosome after adjacent mutation:
10- 12- 1- 7- 2- 13- 11- 4- 8- 5- 3- 6- 9

Inverse Mutation:

Here a set of successive genes are selected and the entire set is reversed and if this leads to infeasible chromosome then it is repaired.

Chromosome before inverse mutation:
10- 12- 1- 7- 2- 13- 11- 4- 8- 5- 3- 6- 9

Chromosome after inverse mutation and before & after repair

10- 12- 1- 4- 11- 13- 2- 7- 8- 5- 3- 6- 9

Shift Mutation

In this mutation a gene is selected randomly- and it is shifted to another randomly selected position and again if necessary- it is repaired.

Chromosome before shift mutation:
10- 12- 1- 4- 11- 13- 2- 7- 8- 5- 3- 6- 9
Chromosome after shift mutation:
10- 12- 1- 4- 11- 5-13- 2- 7- 8- 3- 6- 9

Similar to crossover- here also there is a possibility of violation of precedence constraints after mutation operation which is taken care of again by the repair function.

4.1.8 Incorporate a local search into the GA

In this work a local search called Segment Random Insertion Perturbation Scheme (SRIPS) is used. According to this local search for each sequence obtained after mutation, few more neighbors are created and the best among them will be transferred in to the next generation. The following steps explain the local search procedure.

Step 1: Consider the chromosome which is generated after crossover and mutation.
10- 12- 1- 4- **11**- 5-13- 2- 7- 8- 3- 6- 9
Step 2: Select randomly a sub-segment (S), of size “p”.
S= 11- 5-13- 2- 7 and P = 5
Step 3: Insert any one member of the sub-segment at one of the randomly selected position, to the left or right of the sub-segment.
10- 12- 11-1- 4--5- 13- 2- 7- 8- 3- 6- 9
Insertion to left creates a neighbor to the sequence and Insertion to right creates another neighbor to the sequence.
10- 12- 1- 4- 5-13-11- 2- 7- 8- 3- 6- 9
Step 4: Repeat step 3, till all the members in the sub-sequence are selected. This creates “2p” neighbors.
Step 5: Compute the fitness of all the neighbors created and sort them in the descending order of their fitness. Select the fittest members among the neighbors

4.1.9 Receptor Editing Scheme for creating new generation:

The editing of the chromosomes in the population after the cross over operation is known as receptor editing. In this process a number of worst makespan value chromosomes are eliminated from the population and randomly generated chromosomes are added in those places. After editing the chromosomes in the population- the new population has gone to next iteration until termination criterion is reached. This concept is adopted from

4.1.10 Termination

The process of selection- crossover and mutation are repeated till the termination criterion is satisfied. In this work the number of generations is taken as the termination criterion- which is varied from 100 to 1000 generations and its effect on the performance of the algorithm is noted down.

4.1.11 Genetic Algorithm parametric setting

Genetic algorithm’s evolutionary procedure has been implemented in JAVA language and simulated for various problem sets. Population size is taken as twice the chromosome length (total number of operations in the job set). The results are obtained after repeating the evolutionary procedure for 20 runs and the number of generations is varied from 100 to 1000. Crossover rate 20%, 30% , 40% and 50%, Mutation rate is 0.02%, 0.04%, 0.06% & 0.08% ,

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

Receptor Editing 10% and 20%

5. Computational Analysis

The proposed method has now been applied to the benchmark problems given by Bilge and Ulusoy (1995) with extra data on the tools needed to perform the operation. These benchmark problems were generated at different levels of travel times to processing times (t/p) ratio. These benchmark problems were produced with the proportion of processing times (t/p) at distinct rates of travel times. Ten job sets, 4 layouts and 2 AGVs were combined to design the 120 test problems, 40 with t/p > 0.25 and 80 with t/p < 0.25. Three cases with 4 distinct layouts (LY1, LY2, LY3 and LY4) were considered here for makespan calculation with growing processing times. Original processing times were employed in Table 2 where the processing times were doubled and tripled respectively, as in Table 3 and Table 4.

Table 2: Performance analysis (t/p>0.25)

Job. No	t/p	FCFS	SPT	LPT	SRIPSGA
1.1	0.59	173	193	177	94
2.1	0.61	158	158	177	104
3.1	0.59	202	224	198	112
4.1	0.91	263	267	264	113
5.1	0.85	148	164	148	84
6.1	0.78	231	240	227	124
7.1	0.78	195	210	201	120
8.1	0.58	261	261	266	185
9.1	0.61	270	277	268	120
10.1	0.55	308	308	310	174
1.2	0.47	143	173	165	76
2.2	0.49	124	124	130	77
3.2	0.47	162	188	160	87
4.2	0.73	217	223	224	85
5.2	0.68	118	144	131	73
6.2	0.54	180	169	165	104
7.2	0.62	149	160	149	85
8.2	0.46	181	181	198	159
9.2	0.49	250	249	244	105
10.2	0.44	290	288	287	152
1.3	0.52	145	175	167	82
2.3	0.54	130	130	136	82
3.3	0.51	160	190	162	89

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

4.3	0.8	233	237	230	94
5.3	0.74	120	146	133	70
6.3	0.54	182	171	167	112
7.3	0.68	155	166	151	87
8.3	0.5	183	183	200	169
9.3	0.53	252	251	246	108
10.3	0.49	293	294	293	158
1.4	0.74	189	207	189	104
2.4	0.77	174	174	174	107
3.4	0.74	220	250	212	126
4.4	1.14	301	301	298	128
5.4	1.06	171	189	171	97
6.4	0.78	249	252	237	133
7.4	0.97	217	242	151	135
8.4	0.72	285	285	200	195
9.4	0.76	292	311	290	125
10.4	0.69	350	350	345	182

In the optimal sequence of machines and AGVs are determined by using FCFS- SPT- LPT and SRIPSGA for T/P>0.25 and shown in Table no 2. From table 2- out of 40 problems 40 problems gives improved results using SRIPSGA in comparison with FCFS- SPT and LPT
.

Table 3: Performance analysis (t/p<0.25)

Job.No	t/p	FCFS	SPT	LPT	SRIPSGA
1.10	0.15	207	248	252	121
2.10	0.15	217	217	225	130
3.10	0.15	257	327	282	162
4.10	0.15	303	328	317	119
5.10	0.21	152	190	187	93
6.10	0.16	304	281	297	194
7.10	0.19	231	240	264	137
8.10	0.14	338	338	347	292
9.10	0.15	390	367	359	185

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

10.10	0.14	452	429	444	267
1.20	0.12	194	238	246	122
2.20	0.12	194	194	206	130
3.20	0.12	241	311	270	159
4.20	0.12	285	312	298	116
5.20	0.17	142	180	184	**92**
6.20	0.12	292	260	284	174
7.20	0.15	212	218	249	136
8.20	0.11	306	319	334	287
9.20	0.12	380	355	347	179
10.20	0.11	445	423	439	260
1.30	0.13	195	239	247	117
2.30	0.13	197	197	209	126
3.30	0.13	240	312	271	160
4.30	0.13	292	317	301	117
5.30	0.18	141	181	183	91
6.30	0.24	296	261	285	192
7.30	0.17	215	221	250	137
8.30	0.13	307	320	335	288
9.30	0.13	381	356	348	180
10.30	0.12	448	426	442	270
1.40	0.18	213	255	254	120
2.40	0.13	221	221	228	136
3.40	0.18	261	330	282	162
4.40	0.19	315	336	323	120
5.40	0.18	155	197	186	98
6.40	0.19	310	288	299	180
7.40	0.24	239	251	270	137
8.40	0.18	343	343	349	293

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

9.40	0.19	396	379	370	182
10.40	0.17	466	445	455	266

In the optimal sequence of machines and AGVs are determined by using FCFS- SPT- LPT and SRIPSGA for T/P<0.25 and shown in Table no 3. From Table 3 - out of 40 problems 40 problems gives improved results using HGA in comparison with FCFS- SPT and LPT

Table 4: Performance analysis (t/p<0.25)

Job.No	t/p	FCFS	SPT	LPT	SRIPSGA
1.11	0.15	290	349	361	177
2.11	0.15	299	299	316	196
3.11	0.15	366	473	411	239
4.11	0.15	426	467	448	175
5.11	0.21	215	262	271	138
6.11	0.16	443	398	433	286
7.11	0.19	325	334	379	203
8.11	0.14	488	488	508	433
9.11	0.15	560	521	509	269
10.11	0.14	652	617	641	395
1.21	0.12	280	339	358	178
2.21	0.12	276	276	297	183
3.21	0.12	350	457	399	236
4.21	0.12	407	450	429	172
5.21	0.17	205	252	268	137
6.21	0.12	432	377	420	290
7.21	0.15	299	315	364	202
8.21	0.11	469	469	495	428
9.21	0.12	550	509	497	266
10.21	0.11	645	612	638	386
1.31	0.13	279	340	357	175
2.31	0.13	279	279	300	192
3.31	0.13	349	458	400	237
4.31	0.13	412	453	430	173

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

5.31	0.18	204	253	267	136
6.31	0.24	433	378	421	276
7.31	0.17	302	318	365	203
8.31	0.13	470	470	496	429
9.31	0.13	551	510	498	267
10.31	0.12	648	615	641	387
1.41	0.18	296	356	363	176
2.41	0.13	307	307	319	193
3.41	0.18	370	476	411	239
4.41	0.19	434	471	451	176
5.41	0.18	218	269	270	137
6.41	0.19	445	405	433	281
7.41	0.24	329	344	385	203
8.41	0.18	493	493	508	434
9.41	0.19	560	533	520	269
10.41	0.17	666	633	652	391

In the optimal sequence of machines and AGVs are determined by using FCFS- SPT- LPT and SRIPSGA for T/P<0.25 and shown in Table no 4. From Table 4 out of 40 problems 40 problems gives improved results using SRIPSGA in comparison with FCFS- SPT and LPT

6. Conclusions

The purpose of this research work is to provide an integrated schedule for minimum makespan and also to study the impact of considering machines, AGVs and job transfer times in a multi machine FMS for simultaneous scheduling.FMS scheduling plays an effective and vital role in efficient utilization of resources. This problem being an NP hard, a metaheuristic optimisation algorithm, called SRIPSGA is developed for solving this optimization problem. This scheduling problem is about integrating the assignment of the appropriate machine-AGV-tool combination into each job operation and sequencing and timing of those job operations while constraints are imposed on the system. The results show that tool waiting time and job transfer time between machines have significant influence on makespan. Hence, it is confirmed that by omitting the tool waiting and job transfer times will make the result of scheduling non-realistic. The program developed can be adapted to any layout configuration provided change in AGV travel time matrix in input data. The work can be explored further by considering the transfer times of tools between machines and tool shift activities caused by tool wear.

7. Acknowledgments

Financial support from DST-SERB- Govt.of India (Sanction No: SB/EMEQ-501/2014)

References

[1]. Viswanadham N and Narahari Y, “Performance Modeling of Automated Manufacturing Systems”, Prentice Hall of India, 1994.

[2]. Stecke K E, “Design, Planning, Scheduling and control problems of Flexible Manufacturing Systems”, Annals of Operations Research, No. 3, 1985, pp 3-12.

[3]. Kusiak A, “FMS scheduling: a crucial element in expert system control architecture”, IEEE International Conference on Robotics and Automation, 1986, pp.653-658

[4]. Chang Y, Matsuo H, and Sullivan R S, “A bottleneck- based beam search for job scheduling in a flexible manufacturing system”, International Journal of Production Research, 27,1989, pp1949-1961.

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

[5]. Pundit R, and Palekar U S, “Job shop scheduling with explicit material handling considerations”, Working paper, Dept.of M and IE, Univ. of Illinois at Urbana-Champaign,Urbana,IL61801 1990.

[6]. Biegel J E and Davern J J, “Genetic algorithms and job shop scheduling”, Computers and Engineering, Vol.19, 1990, pp 81-91.

[7]. Ulusoy G, and Bilge U, “Simultaneous Scheduling of machines and automated guided vehicles”, International Journal of Production Research, Vol.31, No.12, 1993, pp 2857-2873.

[8]. Chen C L, Vempati V S, and Aljaber N, “Theory and methodology an application of Genetic algorithms for flow shop problems”, European Journal of Operations Research, Vol 80, 1995, pp 389-396.

[9]. Bilge U and Ulusoy G, “A time window approach to simultaneous scheduling of machines and material handling system in an FMS”, Journal of Operations Research, Vol.43, No.6, 1995, pp1058-1070.

[10]. Cheng R, Gen M, and Sujimura Y T, “ A Tutorial survey of job shop scheduling problems using genetic algorithms- I, Representation”, Computers and Industrial Engineering, No.4, 1996, pp983-997.

[11]. Al-Hakim L, “An analogue genetic algorithm for solving job shop scheduling problems”, International Journal of Production Research, Vol39, No 7, 2001, pp 1537-1548.

[12]. Abdelmaguid T F ,Nasef A O, Kamal B A and Hassan M F, “A hybrid GA / heuristic approach to the simultaneous scheduling of machines and automated guided vehicles”, International Journal of Production Research ,Vol.42, No.2, 2004, pp 267-281.

[13]. Reddy B S P and Rao C S P, “A hybrid multi-objective GA for simultaneous scheduling of machines and AGVs in FMS”, International Journal of Advanced Manufacturing Technology, 31, 2006, pp 602-613.

[14]. Jerald J, Asokan P , Saravanan P , Rani D C, “Simultaneous scheduling of parts and automated guided vehicles in an FMS environment using adaptive genetic algorithm”, International Journal Advanced Manufacturing Technology, 29, 2006, pp 584- 589.

[15]. Goncalves J F, and Mendes J J M, “A Hybrid Genetic Algorithm for the Job Shop Scheduling Problem”, AT&T Labs Research Technical Report TD-5EAL6J, September 2002.

[16]. Aytug H, Khouja M, and Vergara F E, “Use of genetic algorithms to solve production and operations management problems: a review”, International Journal Production Research, Vol.41, No.17, 2003, pp 3955-4009.

[17]. Nearchou A C, Omirou S L, “Differential evolution for sequencing and scheduling optimization”, J Heuristics 12, 2006, pp 395–411.

[18]. Sreekara Reddy- M. B. S.- Ratnam- C.- Rajyalakshmi- G.- & Manupati- V. K. (2018). An effective hybrid multi objective evolutionary algorithm for solving real time event in flexible job shop scheduling problem. Measurement- 114- 78–90.

[19]. Patle- B.K.; Parhi- D.R.K.; Jagadeesh- A.; Kashyap- S.K(2017). Matrix-Binary Codes based Genetic Algorithm for path planning of mobile robot. Comput. Electr. Eng- 67- 708–728.

[20]. L. Pappula and D. Ghosh (2018). Cat swarm optimization with normal mutation for fast convergence of multimodal functions. Appl. Soft Comput. 66- 473–491.

[21]. Durga Rajesh K.V.- Chalapathi P.V.- Nageswara Rao M.- Krishna C.E.- Anoop K.- Neeraj Y. (2017)-’An efficient sheep flock heredity algorithm for the cell formation problem. ARPN Journal of Engineering and Applied Sciences-12(21)- 6074-6079.

[22]. Nageswara Rao M.- Sai Bharath C.- Venkatesh Y.- Lokesh K.- Harish V.- Vara Kumari S. (2017).

[23]. Simultaneous scheduling in FMS through priority rules. Journal of Advanced Research in Dynamical and Control Systems-9-1995-2008.

[24]. M. Nageswararao et al. (2017) Scheduling of Machines and Automated Guided Vehicles in FMS using Shuffled Frog Leap Algorithm. International Journal of Mechanical Engineering and Technology- Vol. 8(5)- pp. 496–503.

[25]. P.A. Kumar- M. Nageswararao- T.Y.Kumar-M.M.Kumar- V. Bhanu (2017). Scheduling of machines and automated guided vehicles in FMS using scatter search algorithm- International Journal of Mechanical Engineering and Technology- 8(5)-471-481.

[26]. M. Nageswara Rao et al (2017). Application of BPSO in flexible manufacturing system scheduling- International Journal of Mechanical Engineering and Technology- 8(5)- 186-195.

[27]. K. Prakash Babu- V.Vijaya Babu- M.Nageswara Rao (2018). Implementation of heuristic algorithms to synchronized planning of machines and AGVs in FMS- Management Science Letters- 8(6)- 543-554

[28]. P.B..Kanakavalli-V.B.Vommi-M.Nageswara Rao(2018). Fuzzy heuristic algorithm for simultaneous scheduling problems in flexible manufacturing system- Management Science Letters-

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

8(12)- 1319-1330.
[29]. M.Nageswara Rao- K.Dileep- Shaik Khadar Basha- Vara Kumari.S (2020). Modrak Algorithm to Minimize Completion Time for n-Jobs m-Machines Problem in Flexible Manufacturing System- International Journal of Emerging Trends in Engineering Research- 8(8)- 4560-4566.
[30]. M. Nageswara Rao, S.Vara Kumari,P. Manohar, B. Madesh, P. Naveen Krishna, R. Suraj Krishna (2020). Simultaneous Scheduling of Machines and AGVs in FMS Through Ant Colony Optimization Algorithm. International Journal of Engineering and Advanced Technology (IJEAT), 9(3), 1392-1397
[31]. M. Nageswara Rao, Vara Kumari S., Praneeth I., Gaya Prasad K., D.Venkata Reddy E.Vineeth, D. Maheshwar Reddy(2020). Simultaneous Scheduling of Machines and AGVs in FMS through Simulated Annealing Algorithm. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 9(4), 2235-2240
[32]. Kanakavalli Prakash Babu, Vommi Vijaya Babu, Medikondu Nageswara Rao (2019). Scheduling of Machines and AGVs Simultaneously in FMS through Hybrid Teaching Learning Based Optimization Algorithm. International Journal of Engineering and Advanced Technology (IJEAT), 9(2), 2048-2055
[33]. M. Nageswara Rao, K. Lokesh, V. Harish, Ch. Sai Bharath, Y. Venkatesh, Vara Kumari. S (2018)- Simultaneous Scheduling through Heuristic Algorithm, International Journal of Engineering & Technology, 7 (2.32), 125-130
[34]. Nageswara Rao, M., Narayana Rao, K., & Ranga Janardhana, G. (2017). Integrated Scheduling of Machines and AGVs in FMS by Using Dispatching Rules. Journal of Production Engineering, 20(1), 75-84.
[35]. Mr. Dharmesh Dhabliya, Dr.S.A.Sivakumar. (2019). Analysis and Design of Universal Shift Register Using Pulsed Latches . International Journal of New Practices in Management and Engineering, 8(03), 10 - 16
[36]. Nageswara Rao, M., Narayana Rao, K., & Ranga Janardhana, G. (2014). Machines and AGVs Scheduling in Flexible Manufacturing System with Mean Tardiness Criterion. International journal of Advanced Materials Manufacturing and Characterization, Vol. 4, pp. 100-105.
[37]. Subbaiah, K, V., Nageswara Rao, M., & Narayanarao, K. (2010). Scheduling of AGVs and Machines in FMS with Make Span Criteria Using Sheep Flock Heredity Algorithm. International Journal of Physical Sciences, 4(2), 139-148.
[38]. MedikonduNageswararao, Dr.K.Narayanarao, Dr.G.Ranagajanardhana “Simultaneous Scheduling of Machines and AGVs in Flexible Manufacturing System with Minimization of Tardiness Criterion” International Conference on Advances in Manufacturing and Materials Engineering (ICAMME-2014),2014.
[39]. Nageswara Rao, M., Narayana Rao, K., & Ranga Janardhana, G. (2017). Scheduling of Machines and Automated Guided Vehicle in FMS using Gravitational search algorithm. *Applied Mechanics and Materials*, 867, 307-313.

APPENDIX A

Travel time matrix for this particular problem

Layout-1					
From/To	L/U	M1	M2	M3	M4
L/U	0	6	8	10	12
M1	12	0	6	8	10
M2	10	6	0	6	8
M3	8	8	6	0	6
M4	6	10	8	6	0

Layout-2					
From/To	L/U	M1	M2	M3	M4
L/U	0	4	6	8	6
M1	6	0	2	4	2
M2	8	12	0	2	4
M3	6	10	12	0	2
M4	4	8	10	12	0

Layout-3					
From/To	L/U	M1	M2	M3	M4

Layout-4					
From/To	L/U	M1	M2	M3	M4

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

L/U	0	2	4	10	12
M1	12	0	2	8	10
M2	10	12	0	6	8
M3	4	6	8	0	2
M4	2	4	6	12	0

L/U	0	4	8	10	14
M1	18	0	4	6	10
M2	20	14	0	8	6
M3	12	8	6	0	6
M4	14	14	12	6	0

Data for the Job Sets Used in Example Problems

JobSet-1 Job 1: Ml(8); M2(16); M4(12) Job 2: Ml(20); M3(10); M2(18) Job 3: M3(12); M4(8); Ml(15) Job 4: M4(14); M2(18) Job 5: M3(10); Ml(15)	JobSet-2 Job 1: Ml(10); M4(18) Job 2: M2(10); M4(18) Job 3: Ml(10); M3(20); Job 4: M2(10); M3(15); M4(12) Job 5: Ml(10); M2(15); M4(12) Job 6: M1(10); M2(15); M3(12)
JobSet-3 Job 1:Ml(16); M3(15) Job 2:M2(18); M4(15) Job 3:Ml(20); M2(10) Job 4:M3(15); M4(10) Job 5:Ml(8);M2(10);M3(15);M4(17) Job 6: M2(10);M3(15);M4(8);Ml(15	JobSet-4 Job1: M4(11); Ml(10); M2(7) Job2: M3(12); M2(10); M4(8) Job3: M2(7); M3(10); Ml(9); M3(8) Job4: M2(7); M4(8); Ml(12);M2(6) Job5:Ml(9);M2(7);M4(8);M2(10);M3(8)
JobSet-5 Job 1: Ml(6);M2(12);M4(9) Job 2: Ml(18);M3(6); M2(15) Job 3: M3(9);M4(3);Ml(12) Job 4: M4(6);M2(15) Job 5: M3(3);Ml(9)	JobSet-6 Job 1: Ml(9); M2(11); M4(7) Job 2: Ml(19); M2(20); M4(13) Job 3: M2(14); M3(20); M4(9) Job 4: M2(14); M3(20); M4(9) Job 5: Ml(11); M3(16); M4(8) Job 6: Ml(10); M3(12); M4(10)

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

JobSet-7 Job 1: Ml(6); M4(6) Job 2: M2(11); M4(9) Job 3: M2(9); M4(7) Job 4: M3(16); M4(7) Job 5: Ml(9); M3(18) Job 6: M2(13); M3(19); M4(6) Job 7: Ml(10); M2(9); M3(13) Job 8: Ml(l1); M2(9); M4(8)	JobSet-8 Job 1: M2(12); M3(21);M4(11) Job 2: M2(12); M3(21);M4(11) Job 3: M2(12); M3(21);M4(11) Job 4: M2(12); M3(21);M4(11) Job 5: Ml(10); M2(14);M3(18);M4(9) Job 6: Ml(10);M2(14); M3(18);M4(9)
JobSet-9 Job 1: M3(9);Ml(12);M2(9);M4(6) Job 2: M3(16);M2(11); M4(9) Job 3: Ml(21); M2(18); M4(7) Job 4: M2(20); M3(22); M4(11) Job 5:M3(14);Ml(16);M2(13); M4(9)	JobSet-10 Job1:Ml(11);M3(19);M2(16);M4(13) Job2: M2(21);M3(16); M4(14) Job3:M3(8); M2(10); Ml(14); M4(9) Job4: M2(13); M3(20); M4(10) Job5: Ml(9); M3(16); M4(18) ; Job6:M2(19);Ml(21); M3(11);M4(15)

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731