SIMULTANEOUS SCHEDULING OF MACHINES AND AGVS USING FLOWER POLLINATION ALGORITHM: A NEW NATURE-INSPIRED META-HEURISTIC

[1]Sivarami Reddy N, [2]Dr. Ramamurthy DV and [3]Dr. Prahlada Rao K

[1]Research Scholar, JNTUA, Ananthapuram, Andhra Pradesh
[2]Principal, GIET, Rajahmundhry, Andhra Pradesh
[3]Professor, Mech. Engg., JNTUACEA, Ananthapuram, Andhra Pradesh
[1]E-mail: siva.narapureddy@gmail.com

Abstract: *This paper addresses the problem of simultaneous scheduling of machines and two identical automated guided vehicles (AGVs) in a flexible manufacturing system (FMS). It is a NP–hard problem which is very complex. For solving this problem, a new nature inspired meta-heuristic Flower pollination Algorithm (FPA) is proposed. The problem consists of two interrelated problems, scheduling of machines and scheduling of AGVs. A simultaneous scheduling of these, in order to minimize the makespan will result in an FMS being able to complete all the jobs assigned to it at earliest time possible, thus saving resources. Improvement in performance of FMS can be expected by efficient utilization of its resources, by proper integration and synchronization of their scheduling. The proposed heuristic is tested on problems generated by various researchers and the results are compared with results of existing methods. The results show that the proposed heuristic is outperformed the existing methods.*

Keywords: *Flexible Manufacturing Systems, Flower Pollination Algorithm, Simultaneous Scheduling of Machines and AGVs, Minimization of Makespan*

1. INTRODUCTION

FMS is an integrated manufacturing system which incorporates many modern facilities such as Computer Numerically Controlled (CNC) machines, Automated Guided Vehicles (AGVs), Automated storage/ Retrieval Systems (AS/RS), Central Tool Magazine (CTM), Robots and Automated inspection using machine vision system under the control of a central computer [3,1]. Various subsystems flexibilities are integrated together in creating an overall flexibility in FMS. One of the modern techniques in industrial automation is FMS, and many researchers have been attracted towards FMS over the last three decades. FMS has many advantages such as greater productivity, minimum work-in-process inventory, high machine utilization, production with minimum supervision, increased product variety and high quality to satisfy customer needs. The use of fixtures, pallets, tool transporter and CTM practically eliminated the job setting time [4]. Broadly FMS is classified into four different categories; Single Flexible machines (SFM), Flexible Manufacturing Cells (FMCs), Multi machine FMS (MMFMS) and multi cell FMS (MCFMS) [2]. FMS aims at combining the advantages of higher efficiency in high volume mass production and higher flexibility in low volume job shop production.

In FMS, in order to achieve the higher efficiency and flexibility various scheduling decisions such as allocation of machines to jobs, allocation of AGVs and selection of tools are made. Proper scheduling plays a critical role in FMS in improving productivity.

2. LITERATURE REVIEW

In scheduling problems, for ‘p’ jobs and ‘q’ machines, $(p!)^q$ different number of sequences are to be examined with respect to any performance measure, to suggest a best sequence. This implies that the search region is increased exponentially for problem of larger size that makes the scheduling problem as NP-hard problem. In FMS various jobs are to be allocated to machines to optimize the performance of FMS. This is similar to job shop scheduling. The main difference between them is that the job shop considers only jobs and machines, where as FMS considers resources such as AGVs, CTM, AS/RS, Robots, Pallets and Fixtures in addition to Jobs and machines. Hence scheduling problems connected with FMS are also NP-hard. Many researchers have addressed the machine and vehicle scheduling as independent problems. However the importance of simultaneous scheduling of jobs and automated guided vehicles (AGVs) has been emphasized by only few researchers. Raman et al [5] addressed the problem as an integer programming problem and procedure for solution based on the concepts of project scheduling under resource constraints. It was assumed that after transferring the load, the vehicle always returns to the load/unload station, which reduces the AGV flexibility and influences the schedule length. Ulusoy and Bigle [6] attempted to make AGV scheduling an integral part of scheduling activity in an FMS. The problem was decomposed into two sub problems i.e. machine scheduling problem and vehicle scheduling problem. At each iteration, a new schedule for machines, generated by heuristic procedure was examined for its feasibility to the vehicle scheduling sub problem. The combined machine and AGVs scheduling problem was formulated as a non-linear mixed integer programming (MIP) model. BILGE and ULUSOY [7] proposed an iterative method based on the decomposition of the master problem into two sub-problems i.e., machine scheduling problem and vehicle scheduling problem. They developed a heuristic, named ‘sliding time window (STW)’, to solve the simultaneous off-line scheduling of machines and material handling in FMS. Ulusoy et al [8] proposed a genetic algorithm for this problem. Suitable coding scheme was provided, in which chromosome represents both the operation number and AGV assignment. Special genetic operators were developed for this purpose. The authors implemented their GA program with this coding and tested it on the 82 test problems that were solved earlier by the STW heuristic. Abdelmaguid et al [9] has presented a new hybrid genetic algorithm for the simultaneous scheduling problem for the makespan minimization objective. The hybrid GA is composed of GA and a heuristic. The GA is used to address the first part of the problem that is theoretically similar to the job shop scheduling problem and the vehicle assignment is handled by a heuristic called vehicle assignment algorithm (VAA). Murayama and Kawata [10] also addressed simultaneous scheduling of machines and AGVs. However it is assumed that AGVs can carry multiple loads instead of single load at a time. The genetic algorithm was applied to the problem. JERALD et al [11] proposed an adaptive GA (AGA) and ants colony optimization (ACO) for a 16-machine and 43-part problem. Their objective function is a combined objective of minimizing penalty cost and minimizing machine idle time. They also examined the speed of the AGV and found that AGA is superior to the ACO algorithm. Jerald et al [12] proposed the two approaches such as genetic algorithm and adaptive genetic algorithm used for scheduling both parts and AGVs simultaneously in an FMS environment. MURAYAMA and KAWATA [13] proposed a simulated annealing method for the simultaneous scheduling problems of machines and multiple-load AGVs to obtain relatively good solutions for a short time. The proposed method is based on a local search method for job shop scheduling problems. They provided a new representation of solutions and neighborhood operation in order to consider the transportation by multiple-load automated guided vehicles. Reddy and Rao [14] addressed the simultaneous scheduling problem as a multi objective problem in scheduling with conflicting objectives and solved by non-dominated sorting evolutionary algorithms. DEROUSSI et al [15] also addressed the problem of simultaneous scheduling of machines and vehicles in FMS. They proposed a new solution representation based on vehicles rather than machines, whereby each solution can thus be evaluated using a discrete event approach. An efficient neighbouring system is then implemented into three different meta-heuristics, namely iterated local search, simulated annealing and their hybridisation. Their results were compared with previous studies and show the effectiveness of the presented approach. Philippe Lacomma et al [16] attempted to model simultaneous scheduling of machines and identical automated guided vehicles using a frame work based on disjunctive graph and used memetic algorithm for scheduling machines and AGVs with the objective of minimum makespan.

3. FMS ENVIRONMENT

The FMS considered consists of 4 machines, a CTM consisting of tools, Automatic tool changer (ATC), Two identical AGVs and tool transporter (TT). On one end there is loading and unloading station. Buffer storage at each machine centre is provided to store the jobs before and after processing. There is an automated storage and retrieval system (AS/RS) for storage of raw materials and retrieval. The system is shown in figure1 with the elements.

4. PROBLEM AND ASSUMPTIONS

Simultaneous scheduling of the machines and the material handling system in an FMS can be defined as follows: For the FMS described above determine the starting and completion times of operations for each job and the trips between workstations together with the vehicle assignment according to the objective of minimizing the make span.

It is assumed that all the design and set-up issues for the FMS as suggested by STECKE [19] have already been resolved. Four layout configurations as shown in figure 2 and ten job sets reported in are Bilge et al [7] are used. The number of automated guided vehicles (AGVs) in the system is two. The types and number of machines are known. There is a sufficient input/output buffer space at each machine. Machine loading has been done i.e., allocation of tools to machines and the assignment of operations to machines. Operations are not pre- emptive. Each job is available at the beginning of the scheduling period. Ready times of all jobs are known. The routing of each job type is available before making scheduling decisions. Limitations on the jobs simultaneously allowed in the shop are ignored. The load/unload (L/U) station serves as a distribution centre for parts not yet processed and as a collection centre for parts finished. All vehicles start from the L/U station initially. There is a sufficient input/output buffer space at the L/U station. Trips follow the shortest path between two points and occur either between two machines or between a machine and the L/U station. Pre-emption of the trips is not allowed. The trips are called loaded or deadheading (empty) trips depending on whether or not a part is carried during that trip, respectively. The duration of deadheading trips is sequence-dependent and is not known until the vehicle route is specified. Processing, set-up, loading, unloading and travel times are available and deterministic. Vehicles move along predetermined shortest paths, with the assumption of no delay due to the congestion. As a result of this assumption, it would follow that the guide paths on segments can be uni-directional or bi-directional. However, on busy segments, two uni-directional paths should be used instead of a bi-directional guide path so that traffic congestion does not reach a critical level leading to the violation of this assumption. Furthermore, such issues as traffic control, machine failure or downtime, scraps, rework and vehicle dispatches for battery changes are ignored here and left as issues to be considered during real time control.

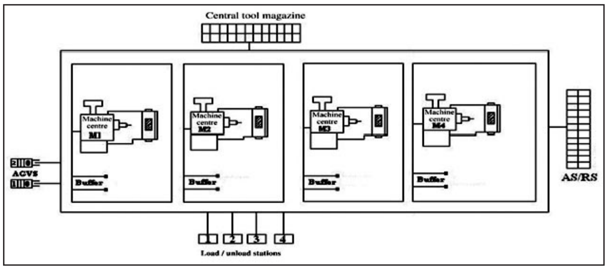


Fig 1. FMS Environment

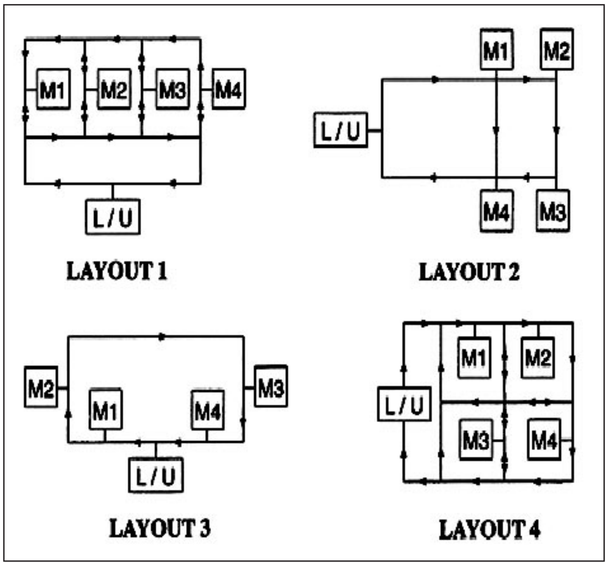


Fig 2. Layout Configurations Used for Examples

The following constraints are to be satisfied by the AGV travel when scheduling these FMSs:

(i) For each operation j, there is a corresponding loaded trip whose destination is the machine where operation j is to be performed and its origin is either the machine where the operation preceding j is assigned or the L/U station;

(ii) Operation j of job I can start only after the trip to load has been completed

(iii) An AGV trip cannot start before the maximum of the completion time of the previous operation of a job and the deadheading trip of the AGV to the job is obtained. The AGV travel times and the machine allocation and operation times for the jobs are given in Appendix A.

5. SIMULTANEOUS SCHEDULING METHODOLOGY

5.1 Algorithm / Procedural Steps in Simultaneous Scheduling methodology

Step 1: Enter the input data: Job set details, solution vector, AGV travelling time matrix.

Step 2: Read parameter of solution vector one after another.

Step 3: Get job no, operation no, machine no, AGV no.

Step 4: Whenever AGV is ready move the AGV to the job, AGV waits till the job is ready then the AGV moves the job from its current location to the machine on which the job next operation is to be performed.

Step 5: Check whether machine is ready or not. If machine is ready load the job, else the job waits in the buffer till machine becomes ready.

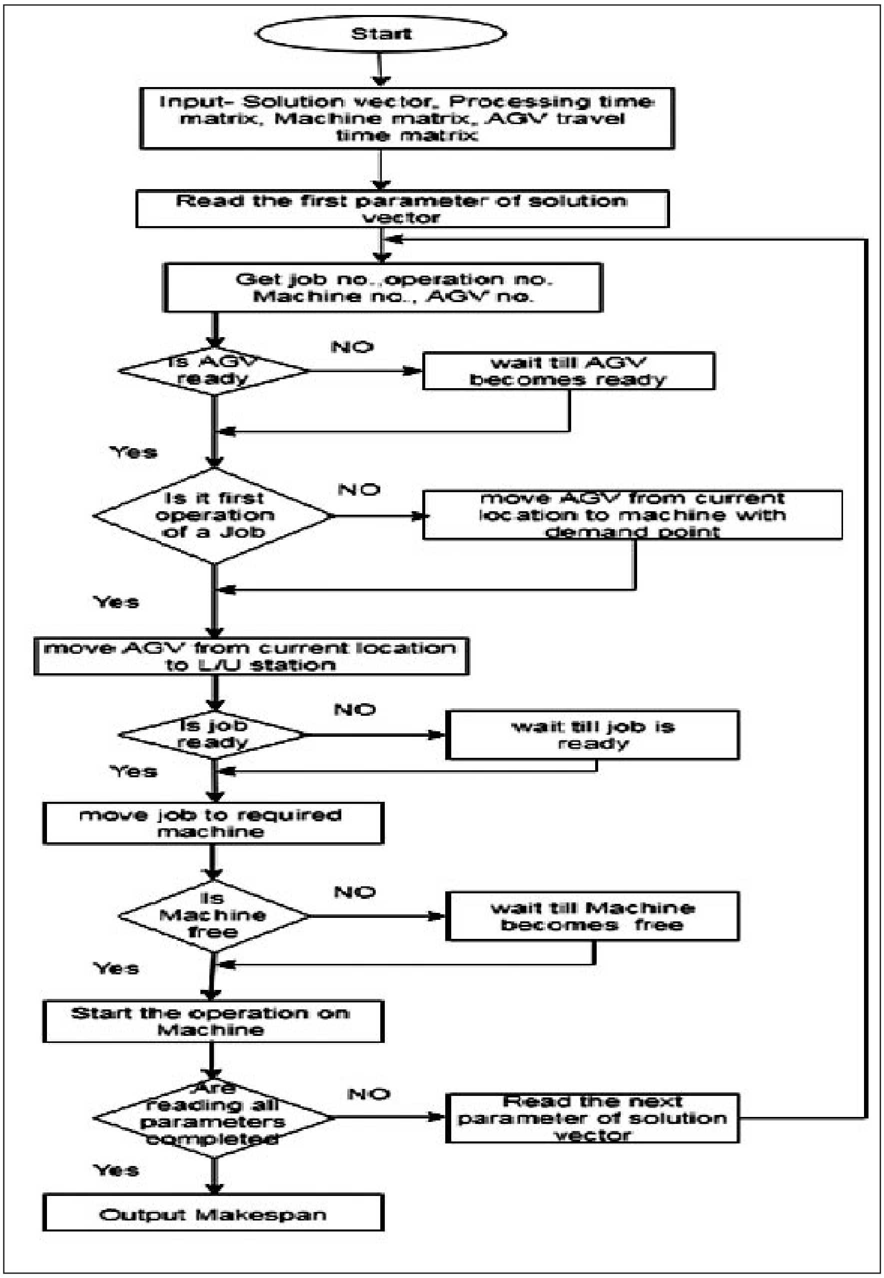


Fig 3. Simultaneous Scheduling Methodology Flow Chart

Step 6: Start the operation on the machine.

Step 7: Check whether all parameters of solution vector are completed. If not, repeat from step2 onwards.

Step 8: If all the parameters are completed output the makespan.

5.2 Limits Function and Bounds Function

Limits function is used to make sure that the operations in the vector so generated using random numbers follows precedence requirement constraints of the operations. If the precedence is not followed, the limits function corrects the vector so that the operations of the vector follows precedence requirement constraints. Bounds function is used to make sure that the AGVs in the vector so generated using random numbers are within bounds. If the AGVs are not within the bounds which will be corrected by bounds function so that the AGVs of the vector are within the bounds.

The flow chart for simultaneous scheduling methodology is shown in figure 3.

6. FLOWER POLLINATION ALGORITHM [18]

Flower pollination is a process related to transferring flowers' pollens. Transferring flowers' pollens are carried by birds, bats, insects and other animals. There are two categories of pollination i.e., abiotic and biotic. In biotic pollination, pollinators transfer pollen and abiotic pollination does not require pollinators. Some pollinators tend to visit some specific type of flowers and at the same time, other species of flowers will be bypassed. This phenomenon is known as flower constancy. All the flowers with flower constancy property guarantee the reproduction maximisation.

Pollination can be achieved through cross-pollination or self-pollination. In cross-pollination, pollens are transferred from a different plant (Yang, 2012).

The biotic and cross-pollination take place at long distances, so they are carried out by pollinators that can fly for long distances. The moves of pollinators like birds and bees can be considered as discrete jumps that obey levy distribution. In self-pollination, same flower pollens or different flowers pollen of the same plant are responsible for fertilization process. The above pollination process characteristics, flower constancy and behavior of pollinator can be idealized as the following rules:

- biotic and cross-pollination is interpreted as global pollination where the pollinators carrying pollens performs levy flights
- the abiotic and self-pollination can be recognized as local pollination
- flower constancy can be considered as a reproduction capability and is proportional to the similarity of the two flowers involved
- due to the wind and physical proximity, local pollination has a little advantage over global pollination.

Flower pollination algorithm imitates the above process to find the optimal solution of a problem. Initially 'n' feasible solutions known as flowers pollens are generated randomly. Switch probability *p* € [0, 1] controls local pollination and global pollination.

If a randomly generated number is less than Switch probability p, global pollination takes place otherwise local pollination takes place.

In global pollination, next generation solution Y_i^{t+1} is calculated using the following equation.

$$Y_i^{t+1} = Y_i^t + L\left(Y_i^t - h^*\right)$$

where Y_i^t is the pollen i (i = 1 to number of pollens) or solution vector at iteration t, h* is the best solution among all solutions at iteration t, L is the step size and is derived from levy distribution.

In local pollination i.e., when the random number is greater than switch probability p, next generation solution Y_i^{t+1} is calculated using the following equation.

$$Y_i^{t+1} = Y_i^t + \in\left(Y_j^t - Y_k^t\right)$$

Where Y_j^t and Y_k^t are two different pollens or solutions from the same iteration. The variable ϵ is drawn from a uniform distribution in [0, 1]. The pseudo code of the above explanation is shown in Figure 4.

7. RESULTS AND DISCUSSIONS

The proposed algorithm is tested on 10 job sets with four different layouts (LY1, LY2, LY3 and LY4) for different traveling time/processing time (t/p)

```
Initialize a population of n flowers/pollen gametes with random solutions
Find the best solution h* in the initial population
Define a switch probability p € [0,1]
while (t <MaxGeneration)
        for i = 1 : n (all n flowers in the population)
            if      rand < p,
                    Draw a (d-dimensional) step vector L which obeys a Levy distribution
                    Global pollination via Yi^(t+1)=Yi^t + L (Yi^t – h*)
            else
                    Draw € from a uniform distribution in [0,1]
                    Randomly choose j and k among all the solutions
                     Do local pollination via Yi^(t+1)= Yi^t + € (Yj^t –Yk^t)
            end if
            Evaluate new solutions
            If new solutions are better, update them in the population
        end for
        Find the current best solution h*
end while
```

Fig 4. Pseudo Code of the Proposed Flower Pollination Algorithm

ratios reported in Bilge et al. [17] Three cases are considered here for makespan calculation with increasing processing times with four different layouts (LY1, LY2, LY3 and LY4). In case I original processing times are used, in case II processing times are taken as double the original processing times and in case III processing times are taken as triple the original processing times. For LY1, LY2 and LY3 Case I and II are considered, for LY4 all the three cases are considered. For case II and case III AGV travelling times are halved. The three cases are grouped into two sets, one with relatively high t/p ratio >0.25 (case I) and the other with relatively low t/p <0.25 (case II and case III). Results of proposed algorithm for case I of LY1, LY2, LY3 and LY4 in Table No. 1 , case II of LY1, LY2, LY3 and LY4, case III of LY4 in Table No.2 are presented. From these Tables it is observed that the proposed algorithm outperforms the other methods.

Table 1 consist of make span of problems whose t_i/p_i ratios greater than 0.25 while Table 2 consist of make span of problems whose t_i/p_i ratios are less than 0.25. A code is used to designate the problems which are given in the first column. The digits that follow EX indicates the job set in the layout. In table 2, another digits is appended to the code. Here having zero or one as the last digit implies that the process times are doubled or triple respectively, where as travel times are halved in both cases.

The results of the proposed FPA are better over the STW on 31 problems, the UGA on 24 problems, the AGA on 16 problems, the RGA on 4 problems, the PDE1 on7 problems and the PDE2 on 8 problems, same on 9 problems in STW, 16 problems in UGA, 23 problems in AGA, 35 problems in RGA, 28 problems in PDE1and 25 problems in PDE2 and poor on 1 problem in AGA , 1 problem in RGA, 5 problems in PDE1 and 7 problems in PDE2 for the case t/p ratio >0.25.

For t/p ratio <0.25 the proposed FPA performs better on 17 problems in STW, 5 problems in UGA, 4 problems in AGA, 4 problems in RGA, 4 problems in PDE1 and 4 problems in PDE2, same on 24 problems in STW, 35 problems in UGA, 38 problems in AGA, 38 problems in RGA, 37 problems in PDE1 and 37 problems in PDE2 and poor on 1 problem in STW, 2 problems in UGA, 1 problem in PDE1 and 1problem in PDE2.

At the outset, out of the 82 problems the proposed FPA performs better than STW on 48 problems, UGA on 29 problems, AGA on 20 problems, RGA on 8 problems, PDE1 on 11 problems and PDE2 on 12 problems.

7.1 Gantt Chart

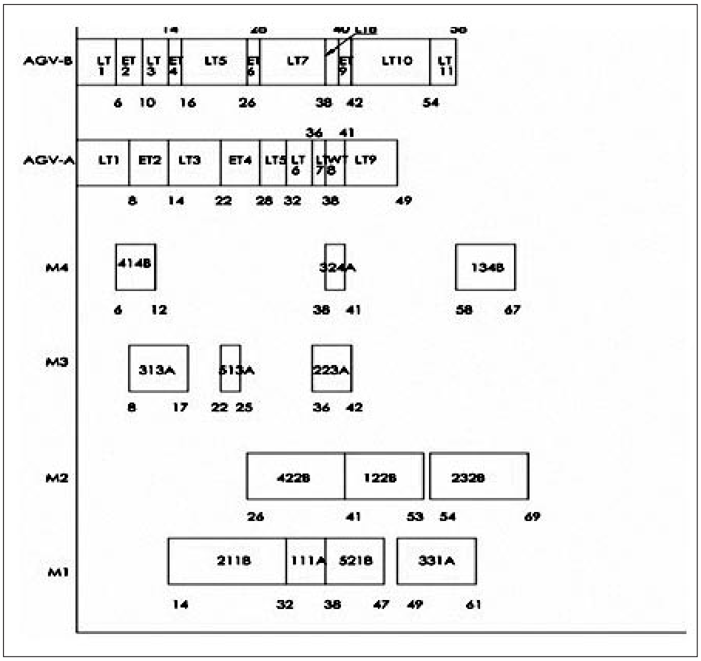
The Gantt chart for the sequence generated

Table 1: Comparison of FPA Results with Other Methods for t_i/p_i>0.25

Job set	FPA	STW	UGA	AGA	RGA	PDE1	PDE2
EX 1.1	96	96	96	96	96	96	96
EX 1.2	82	82	82	82	82	82	82
EX 1.3	84	84	84	84	84	84	84
EX 1.4	103	108	103	103	103	103	103
EX 2.1	100	105	104	102	100	100	100
EX 2.2	76	80	76	76	76	76	76
EX 2.3	86	86	86	86	86	86	86
EX 2.4	108	116	113	108	108	108	106
EX 3.1	99	105	105	99	99	99	99
EX 3.2	85	88	85	85	85	85	85
EX 3.3	86	86	86	86	86	86	86
EX 3.4	111	116	113	111	111	111	110
EX 4.1	112	118	116	112	112	112	112
EX 4.2	87	93	88	88	87	85	86
EX 4.3	89	95	91	89	89	89	89
EX 4.4	89	95	91	89	89	89	89
EX 5.1	87	89	87	87	87	87	87
EX 5.2	69	69	69	69	69	69	69
EX 5.3	74	76	75	74	74	74	74
EX 5.4	96	99	97	96	96	96	95
EX 6.1	118	120	121	118	118	115	118
EX 6.2	98	98	98	98	98	98	98
EX 6.3	103	104	104	104	103	103	103
EX 6.4	120	120	123	120	120	120	120
EX 7.1	111	119	118	115	111	112	114
EX 7.2	79	90	85	79	79	79	79
EX 7.3	83	91	88	86	83	83	84
EX 7.4	126	136	128	127	126	126	126
EX 8.1	151	161	152	161	161	161	161
EX 8.2	141	151	142	151	151	153	151
EX 8.3	143	153	143	153	153	152	153
EX 8.4	153	163	163	163	163	163	163
EX 9.1	116	120	117	118	116	114	114
EX 9.2	102	104	102	104	102	104	104
EX 9.3	105	110	105	106	105	103	103
EX 9.4	122	125	123	122	122	123	123
EX 10.1	150	153	150	147	147	147	147
EX 10.2	135	139	137	136	135	135	135
EX 10.3	139	143	143	141	139	139	139
EX 10.4	158	171	164	159	158	158	158

Table 2: Comparison of FPA Results with Other Methods for t_i/p_i>0.25

Job set	FPA	STW	UGA	AGA	RGA	PDE1	PDE2
EX 1.10	126	126	126	126	126	126	126
EX 1.20	123	123	123	123	123	123	123
EX 1.30	122	122	122	122	122	122	122
EX 1.40	124	124	124	124	124	124	124
EX 2.10	148	148	148	148	148	148	148
EX 2.20	143	143	143	143	143	143	143
EX 2.30	146	146	146	146	146	146	146
EX 2.41	217	217	217	217	217	217	217
EX 3.10	150	148	150	150	150	150	150
EX 3.20	145	148	145	145	145	145	145
EX 3.30	146	149	146	146	146	146	146
EX 3.41	221	221	221	221	221	221	221
EX 4.10	119	121	119	119	119	119	119
EX 4.20	114	116	114	114	114	114	114
EX 4.30	114	116	114	114	114	114	114
EX 4.41	172	179	172	172	172	171	171
EX 5.10	102	102	102	102	102	102	102
EX 5.20	100	100	100	100	100	100	100
EX 5.30	99	99	99	99	99	99	99
EX 5.41	148	154	148	148	148	148	148
EX 6.10	186	186	186	186	186	186	186
EX 6.20	181	181	181	181	181	181	181
EX 6.30	182	184	182	182	182	182	182
EX 6.40	184	185	184	184	184	184	184
EX 7.10	137	137	137	137	137	137	137
EX 7.20	136	136	136	136	136	136	136
EX 7.30	137	137	137	137	137	137	137
EX 7.41	203	203	203	203	203	203	203
EX 8,10	272	292	271	292	292	292	292
EX 8.20	267	287	268	287	287	287	287
EX 8.30	268	288	270	288	288	288	288
EX 8.40	273	293	273	293	293	293	293
EX 9,10	176	176	176	176	176	176	176
EX 9.20	173	174	173	173	173	173	173
EX 9.30	174	176	174	174	174	174	174
EX 9.40	175	177	175	175	175	175	175
EX10.10	238	238	236	238	238	238	238
EX10.20	236	236	238	236	236	236	236
EX10.30	237	237	241	237	237	237	237
EX10.40	240	240	244	240	240	240	240



LT-Loaded Trip Time ET-Empty Trip Time WT-Waiting Time

Fig 5. Gantt Chart for Jobset 5 Layout 2

for job set 5, Layout 2 of Case I by FPA is shown in figure 5. The operations that are assigned to each machine as well as the start and finish times of each operation are shown in the Gantt chart. AGVs Loaded trip times (LT), Empty Trip times (ET) and waiting times (WT) are also shown in Gantt chart. The Gantt chart shows the correctness of the solution provided by the proposed FPA method.

Each operation is denoted as three digit number followed by an alphabet. For example in the operation - 211B

2 - represents Job number,

1 – represents operation number,

1 – represents machine that is used for performing operation and

B – represents AGV that is used for moving job.

For AGV-A

LT1 for 313A, ET2 for 513A, LT3 for 513A, ET4 for 111A, LT5 for 111A,

LT6 for 223A, LT7 for 324A, WT at machine 4 for 331A , LT9 for 331A

For AGV-B

LT1 for 414B, ET2 for 211B, LT3 for 211B, ET4 for 422B, LT5 for 422B,

ET6 for 521B, LT7 for 521B, LT8 for 122B, ET9 for 232B, LT10 for 232B

And LT11 for 134B

8. CONCLUSIONS

Scheduling of jobs and AGVs is carried out for minimizing the makespan objective by Flower Pollination algorithm (FPA). The proposed algorithm is tested on 10 job sets with four different layouts and it is noticed that proposed algorithm outperforms the existing methods in minimizing makespan. The work can be extended by considering down time and AGVs dispatch time for battery change.

REFERENCES

1. Baker, KR: Introduction to Sequencing and Scheduling. New York, Wiley, 1974.
2. Lee, D and DICESARE, F: Integrated Scheduling of FMSs Employing Automated Guided Vehicles, 'IEEE Transactions on Industrial Electronics', 41(6), 1994, 602–610.
3. Agnetis, Alfieri, AA; Brandimarte, P and Prinsecchi, P: Joint Job/Tool Scheduling in a Flexible Manufacturing Cell with No On-Board Tool Magazine” 'Computer Integrated Manufacturing System', 10(1), 1997, 61–68.
4. Jerald, J; and Asokan, P: Simultaneous

Scheduling of Parts and Automated Guided Vehicles in an FMS Environment using Adaptive Genetic Algorithm. International Journal of Advanced Manufacturing Technology, 29(5), 2006, 584–589.

5. Raman, N; Talbot, FB, Rachamadgu, RV: Simultaneous scheduling of machines and material handling devices in automated manufacturing [C]// Stecke KE, Suri R: Proceedings of the Second ORSA/TIMS Conference on Flexible Manufacturing Systems. University of Michigan, Ann Arbor, MI, USA, 1986: 455−466.

6. Ulusoy, G; Bilge U: Simultaneous scheduling of machines and automated guided vehicles [J], 'International Journal of Production Research', 1993, 31(12): 2857−2873.

7. Bilge, U; Ulusoy, G: A time window approach to simultaneous scheduling of machines and material handling system in FMS [J]. 'Operations Research', 1995, 43: 1058−1070.

8. Ulusoy, G; Sivrikaya-Serifoglu F; Bilge,U: A genetic algorithm approach to the simultaneous scheduling of machines and automated guided vehicles [J]. 'Computers & Industrial Engineering', 1997, 24(4): 335−351.

9. Abdelmaguid, TF; Nassef, ON; Kamal, BA; Hassan, MF: A hybrid GA/heuristic approach to the simultaneous scheduling of machines and automated guided vehicles [J]. 'International Journal of Production Research', 2004, 42: 267−281.

10. Murayama, N; Kawata, S: A genetic algorithm approach to simultaneous scheduling of processing machines and multiple-load automated guided vehicles [J]. 'Transactions of the Japan Society of Mechanical Engineers C', 2005, 71(712): 3638−3643. (in Japanese)

11. Jerald J; Asokan P, Saravanan R, Rani, A D C. Simultaneous scheduling of parts and AGVs in an FMS using non-traditional optimization algorithms [J], 'International Journal of Applied Management and Technology, 2005, 3(1): 305−315.

12. Jerald, J; Asokan, P; Saravanan, R, RANI A D C: Simultaneous scheduling of parts and automated guided vehicles in an FMS environment using adaptive genetic algorithms [J], 'International Journal of Advanced Manufacturing Technology', 2006, 59: 584−589.

13. Murayama, N; Kawata, S: Simulated annealing method for simultaneous scheduling of machines and multiple-load AGVs [C]// IJCC Workshop on Digital Engineering. Pyeongchang-gun, Gangwon-do, SouthKorea, 2006: 55−62. (in Japanese)

14. Reddy, BSP; Rao, C S P: A hybrid multi-objective GA for simultaneous scheduling of machines and AGVs in FMS [J]. 'International Journal of Advanced Manufacturing Technology', 2006, 31(5/6): 602−613.

15. Deroussi, L; Gourgand, M; Tchernev, N: A simple met heuristic approach to the simultaneous scheduling of machines and automated guided vehicles [J]. 'International Journal of Production Research', 2008, 46(8): 2143−2164.

16. Philippe Lacomme, Mohand Larabi, Nikolay Tchernev Job-shop based framework for simultaneous scheduling of machines and automated guided vehicles, 'Int. J. Production Economics', 143, 2013, 24–34

17. Stecke, KE: Design, planning, scheduling, and control problems of flexible manufacturing systems [J]. 'Annals of Operations Research', 1985, 3(1): 3−12.

18. Xin-She Yang, Flower pollination algorithm for global optimization, in: Unconventional Computation and Natural Computation 2012, Lecture Notes in Computer Science, vol. 7445, 2012, 240-249 ◘

APENDIX A

A. Travel time matrix for the example problem

	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	1	0	2
M4	4	8	10	12	0

A. Data for the job sets used:

Job Set 1		**Job Set 2**		**Job Set 3**	
Job1	M1- (8);	Job1	M1-(10);	Job1	M1-(16);
	M2- (16);		M4-(18)		M3-(15)
	M4-(12)				

Job2	M1- (20);	Job2	M2-(10);	Job2	M2- (18);
	M3-(10);		M4- (18)		M4- (15)
	M2- (18)				
Job3	M3-(12);	Job3	M1- (10);	Job3	M1- (10);
	M4-(8);		M3- (20)		M2- (10)
	M1- (15)				
Job4	M4- (14);	Job4	M2- (10); M3-(15);	Job4	M3- (15);
	M2-(18)		M4-(12)		M4- (10)
Job5	M3-(10);	Job5	M1- (10); M2-	Job5	M1-(8); M2- (10); M3-(15);
	M1-(15)		(15);		M4-
			M4- (12)		(17)
		Job6	M1-(10);	Job6	M2-(10); M3- (15); M4-
Job Set 4			M2-(15);		(8); M1-
			M3- (12)		(15)
Job1	M4- (11);				
	M1-(10);	**Job Set 5**		**Job Set 6**	
	M2- (7)				
Job2	M3- (12);	Job1	M1-(6);	Job1	M1- (9);
	M2-(10);		M2-(12);		M2-(11);
	M4-(8)		M4- (9)		M4-(7)
Job3	M2- (7);	Job2	M1- (18);	Job2	M1-(19);
	M3-(10);		M3-(6);		M2-(20);
	M1- (9);		M2-(15)		M4-(13)
	M3- (8)				
Job4	M2-(7);	Job3	M3- (9);	Job3	M2- (14);
	M4- (8);		M4-(3);		M3-(20);
	M1- (12);		M1- (12)		M4- (9)
	M2- (6)				
Job5	M1- (9);	Job4	M4-(6);	Job4	M2-(14);
	M2-(7);		M2-(15)		M3- (20);
	M4- (8);				M4-(9)
	M2- (10);				
	M3- (8)				
		Job5	M3-(3);	Job5	M1-(11);
Job Set 7			M1- (9)		M3- (16);
					M4-(8)

Job1	M1-(6);			Job6	M1-(10);
	M4- (6)	**Job Set 8**			M3-(12);
					M4-(10)
Job2	M2-(11);	Job1	M2-(12);		
	M4-(9)		M3- (21);	**Job Set 9**	
			M4- (11)		
Job3	M2-(9);	Job2	M2-(12);	Job1	M3- (9);
	M4-(7)		M3-(21);		M1-(12);
			M4-(11)		M2-(9); M4-(6)
Job4	M3- (16);	Job3	M2-(12);	Job2	M3-(16);
	M4-(7)		M3-(21);		M2- (11);
			M4-(11)		M4-(9)
Job5	M1-(9);	Job4	M2-(12);	Job3	M1-(21);
	M3-(18)		M3-(21);		M2-(18);
			M4-(11)		M4-(7)
Job6	M2-(13);	Job5	M1-(10);	Job4	M2- (20);
	M3-(19);		M2-(14);		M3- (10);
	M4-(6)		M3-(18);		M4-(11)
			M4-(9)		
Job7	M1-(10);	Job6	M1-(10);	Job5	M3-(14);
	M2-(9);		M2-(14);		M1- (16);
	M3-(13)		M3-(18);		M2-(13);
			M4-(9)		M4-(9)
Job8	M1-(11);				
	M2-(9);				
	M4-(8)				
Job Set 10					
Job1	M1-(11);	Job3	M3-(8);	Job5	M1-(9);
	M3-(19);		M2-(10);		M3-(16);
	M2-(16);		M1-(14);		M4-(18)
	M4-(13)		M4-(9)		
Job2	M2-(21);	Job4	M2-(13); M3-(20);	Job6	M2-(19);
	M3-(16);		M4-(10)		M1-(21);
	M4-(14)				M3-(11);
					M4-(15)

N Sivarami Reddy Graduated in Mechanical Engineering from Andhra University in 1989. He completed M.Tech (Advanced Manufacturing Engineering) at NITK, Surathkal in 1992. He is pursuing research in JNTUA, Ananthapuramu. He has 24 years of experience in teaching and administration in engineering colleges. Currently he is a Professor, Mechanical Engineering Department, AITS, Rajampet. He has 10 papers to his credit. His research areas include Scheduling of FMS and Softcomputing Techniques. (E-mail: siva.narapureddy@gmail.com)

Dr. D V Ramamurthy is working currently as Principal, Godavari Institute of Engineering and Technology, Rajahmundry. He graduated from College of Engineering, Andhra University in 1987 and obtained his Post Graduation from IIT, Delhi in 1992. He obtained his Ph.D from Oklahama State University, USA. He has 28 years of teaching experience. He has 20 Publications to his credit. His research areas include Optimization, Design manufacturing and Control systems.

Dr. K Prahlada Rao is presently working as Professor, Department of Mechanical Engineering, JNTUACEA, Ananthapuramu. He has 31 years teaching experience. He Published 80 papers in reputed International Journals and presented 50 papers in International Conferences. His areas of interests include Artificial Intelligence, Expert systems applications in design, Production and industrial Engineering Domains.