Simultaneous Scheduling of Jobs, AGVs and Tools Considering Tool Transfer Times in Multi Machine FMS By SOS Algorithm

N.Sivarami Reddy[1], Dr.D.V.Ramamurthy[2] , Dr.K.Prahlada Rao[3]
1Research scholar,JNTUA,Ananthpuram, AP,India ,Ph.No.+91984898645,
email- siva.narapureddy@gmail.com,
2Principal,GIET,Rajahmundhy,email:rramdwivedula@gmail.com,
3Principal,JNTUACEA,Ananthapuram,email: drkprao1@yahoo.com

Abstract: This article addresses simultaneous scheduling of machines, AGVs and tools where machines are allowed to share the tools considering transfer times of jobs and tools between machines, to generate best optimal sequences that minimize makespan in a multi-machine Flexible Manufacturing System (FMS). Performance of FMS is expected to improve by effective utilization of its resources, by proper integration and synchronization of their scheduling. Symbiotic Organisms Search (SOS) algorithm is a potent tool which is a better alternative for solving optimization problems like scheduling and proven itself. The proposed SOS algorithm is tested on 22 job sets with makespan as objective for scheduling of machines and tools where machines are allowed to share tools without considering transfer times of jobs and tools and the results are compared with the results of existing methods. The results show that the SOS has outperformed. The same SOS algorithm is used for simultaneous scheduling of machines, AGVs and tools where machines are allowed to share tools considering transfer times of jobs and tools to determine the best optimal sequences that minimize makespan.

Keywords: Flexible manufacturing systems, Symbiotic Organisms Search algorithm, AGVs, simultaneous Scheduling of machines, AGVs and Tools, Tool Transporter.

1. Introduction

FMS is an integrated manufacturing system which includes many facilities like computer numerically controlled (CNC) machines, Automated Guided Vehicles (AGVs), Automated Storage/ Retrieval Systems (AS/RSS), Central Tool Magazine (CTM), Robots and Automated inspection under the control of a central computer. Various subsystems flexibilities are integrated together to have an overall flexibility in FMS. One of the recent techniques in industrial automation is FMS, and several researchers have been attracted over the last three decades towards FMS. FMS has many advantages like greater productivity, low work-in-process inventory, high machine utilization, production with least supervision, increased product variety and high quality to satisfy customer requirements. The employing of fixtures, pallets, tool transporter and CTM nearly eliminated the job setting time [1].

The higher flexibility of FMS results in better utilization of resources, better scheduling and routing enhances the productivity. Broadly FMS is categorized into four groups; Single Flexible machines (SFM), Flexible Manufacturing Cells (FMCs), Multimachine FMS(MMFMS) and multi cell FMS (MCFMS). FMS aims at combining the advantages of elevated efficiency in high quantity mass production and better flexibility in low quantity job shop production. In

FMS, in order to achieve the elevated efficiency and flexibility different scheduling decisions like allocation of machines to jobs and selection of tools are made. Proper scheduling plays a critical role in FMS.

2. Literature review

For shop floor productivity improvement, scheduling is recognized to be a crucial task. In scheduling problems, for ‘n’ jobs and ‘m’ machines ‘$(n!)^m$’ different sequences are to be inspected 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 problem of scheduling a NP-hard problem. In FMS different jobs are to be assigned to machines to optimize the FMS performance. This is analogous to job shop scheduling. The main difference between them is that the job shop considers only jobs and machines where as FMS considers resources like AGVs, CTM, AS/RS, Robots, Pallets and Fixtures besides Jobs and machines. Hence problems of scheduling in FMS are also NP-hard.

Jerald and Asokan [2] presented various optimization algorithms for solving FMS scheduling Problems. In the FMS scheduling area, for optimization, earlier the researchers had recognized scheduling of machines and scheduling of tools as two different problems, where as in recent years much interest has been noticed for combined effect of scheduling of machines and scheduling of tools. Several researchers have studied tool scheduling and allocation. Jun, Kim and Sub [3], for provisioning problem and scheduling of tools in FMS, proposed a greedy search algorithm to find the number of required tools from each type for minimizing makespan objective. Also this method gives information about additional number of tools to be purchased when FMS configuration changes due to change of the product mix.

Suresh kumar and sridharan [4] dealt problem of sharing and scheduling of tools for minimising the objectives like mean tardiness, conditional mean tardiness and flow time by employing scheduling priority rules and job scheduling priority rules. Suresh Kumar et al [5] investigated the problem of tool scheduling in FMS by minimising mean flow time, mean tardiness, mean waiting time for tool and percentage of tardy parts by using various priority dispatching rules. Agnetis et al [1] probed a problem of joint part/tool scheduling in FMC. They proposed that all the tools are stored in a central tool magazine and moved throughout the cell by an automatic tool transporter. When the same tool is required by two machines, Tabu search algorithm was employed to address the conflict and prepare production schedules for minimising make span and maximum lateness. Prabhaharan, Nakkeeran and Jawahar [6] attempted on combined operation tool scheduling problem in FMC which consists of a CTM and “m” identical work cells. They proposed simulated annealing algorithm to minimise makespan for combined job and tool scheduling. Udhaykumar and Kumanan [7] proposed ant colony optimisation algorithm for job and tool scheduling problem. J.Aldrin Raj, D.Ravindran et al [8] addressed concurrent machine and tool scheduling in a FMS which has machines and a CTM. They proposed four different algorithms and AIS algorithm, to solve concurrent machine and tool problems with minimum make span as objective.

Most of the 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 [9] addressed the problem as an integer programming problem under resource constraints. It was assumed that after transferring the load, the vehicle always returns to the load/unload station, which reduces

IOP Conf. Series: Materials Science and Engineering **225** (2017) 012294 doi:10.1088/1757-899X/225/1/012294

the AGV flexibility and influences the schedule length. Ulusoy and Bigle [10] 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. Ulusoy et al [11] proposed a genetic algorithm for this problem. Suitable coding scheme was provided, in which chromosome represents both the operation number and AGV assignment. 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 [12] proposed a hybrid genetic algorithm for the problem. The hybrid GA consists of GA and heuristic. The GA addresses the scheduling of jobs and the heuristic called vehicle assignment algorithm handles the vehicle assignment. The hybrid GA is applied on a set of 82 test problems.

Automated tool sharing system is a technological response to high cost of tools in FMS by allowing different machines to employ the same tool by transferring them automatically between machines as tooling needs evolve. In the previous studies some assumptions have been made about concurrent machine and tool scheduling in a FMS consisting of machines, AGVs and a CTM. Earlier researchers had addressed simultaneous scheduling of jobs and AGVs where machines are not allowed to share the tools and simultaneous scheduling of machines and tools where machines are allowed to share the tools but it was assumed that jobs and tools would be transferred instantly with in no time among machines. Omitting these job and tool transfer times will make the result of scheduling impossible to be implemented because these are having considerable influence on makespan.

In this work a new metaheuristic search algorithm SOS is used to minimise makespan by simultaneous scheduling of jobs, AGVs and tools considering transferring times of tools between machines and is explained in the following sections.

3. Problem formulation

Generally CTM is provided in FMS for storage of tools. The tool required by a machine is shared from other machines or transported from the CTM to this machine by a tool transporter (TT) during the machining of job. CTM reduces the number of required tools in the system and hence reduces the tooling cost where as tool transfer time considerably influence the makespan. The FMS has a load/unload (L/U) station. An L/U station serves as a distribution center for parts not yet processed and as a collection center for finished parts. All AGVs start from the L/U station initially and return to there after accomplishing all their assignments. There is sufficient input/output buffer space at the L/U station. Transferring times of jobs between machines influence the makespan so cannot be neglected. The problem definition and assumptions with constraints are given in the following sections.

3.1 Problem Definition

Consider 'n' jobs {J1,J2,J3.........Jn} to be processed through 'm' machines (M1,M2,........Mm} requires 't' tools {T1,T2,........Tt} from CTM, two identical AGVs for transferring jobs between machines and a Tool Transporter to transfer thr tools between the CTM and machines and among the machines. The best sequence by joint selection of jobs, machines and tools is to be found which minimises the make span. In the present work SOS is employed to produce optimal schedule with minimum makespan as objective. The same set of problems that were analyzed with AIS method explained in[11] are considered and the SOS results are compared with those results.

The procedure employed is explained with an example problem. In Table 1 the jobs, tools and machines shown are for job set 1. The job set 1 consists of 5 jobs, the first three jobs have three operations and remaining two jobs have two operations. The system considered has four machines and four tools. An entity in the table gives information about the machine, tool and processing time required for the operation of a job. For example T3-M1[8] shows that Operation I of Job I requires tool T3, machine M1 and 8 units of processing time. The objective is to determine a sequence of jobs that minimises the make span by taking tool and machine constraints into account. Making a decision on selecting a machine and tool for every job is required during the process of scheduling. Both machine and CTM will be having a set of requests from unfinished jobs in the form of queue. A right job with the request has to be selected so as to minimise the make span. Thus, a sequence of operations is formed that minimises the total elapsed time.

Table 1: Job set 1

Jobs	**Operation I**	**Operation II**	**Operation III**
I Job (J1)	T3-M1[8]	T4-M2[16]	T1-M4[12]
II Job (J2)	T2-M1[20]	T3-M3[10]	T1-M2[18]
III Job (J3)	T1-M3[12]	T4-M4[8]	T2-M1[15]
IV Job (J4)	T3-M4[14]	T4-M2[18]	-
V Job (J5)	T2-M3[10]	T1-M1[15]	-

3.2 FMS Environment Considered

The FMS considered has four machines, a CTM consisting of four tools, Automatic tool changer (ATC), Two identical AGVs and TT. There is a L/U station on one end. Jobs are stored in the buffer storage provided at each machine before and after processing. The system is shown in fig.1 with the elements.

Assumptions and Constraints:

The following assumptions are made for the problem under study.

- Each job has J different operations.
- Required machines and tools are known in advance before scheduling to process each operation.
- Operations in a job have its own processing order and there are no technological constraints.
- Each job has the pre-specified sequence of operations and its corresponding processing times.
- Only one job can be processed on each machine at a time.
- Tools are stored in CTM.
- Tool transporter moves the tools throughout the system.
- Tools are shared among the machines in the system.
- The two AGVs are identical in speed and load carrying characteristics.
- AGVs carry a single job at a time.
- AGVs will move along shortest predetermined paths with the assumption of no delay due to congestion.

- Issues such as traffic control, scraps, rework, downtime and vehicle dispatches for battery change are ignored here.

The constraints of the problem are as follows.

- Precedence constraints exist, that is a set of pre-specified sequence of operations will be there for every job that cannot be changed.

  Consider the operation (4143)
  4 – Job number
  1 – First operation of J4
  4 – First operation of J4 is performed on machine 4.
  3 - First operation of J4 requires Tool 3.
  The second operation of J4 cannot be processed before completion of first operation and hence the operation 42XX cannot be processed before 41XX. This restriction in job processing is called precedence constraints.
- A same job cannot be processed on two different machines at a time.

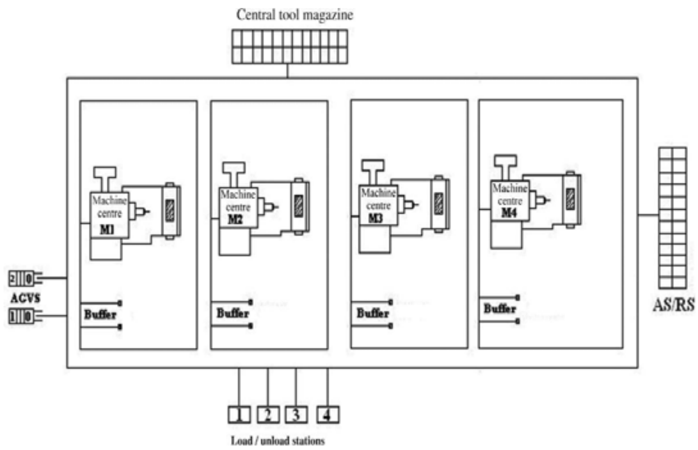

Fig. 1: Considered FMS Environment

4. SYMBIOTIC ORGANISMS SEARCH ALGORITHM

The SOS algorithm, proposed by Cheng and Prayogo [12], is a simple and powerful meta-heuristic algorithm. The SOS algorithm works on the interdependent behaviour seen among organisms in nature. Some organisms do not live alone because they are interdependent on other species for survival and food. The liaison between two different species is known as symbiotic. The SOS algorithm starts with a randomly initiated population, where system has 'n' number of organisms (i.e. eco size) in the ecosystem. The population will be updated in each generation 'g' by 'the mutualism phase', 'the commensalism phase', and 'the parasitism phase' respectively. Moreover, the updated solution in the each phase is accepted if it only has a better functional value. The course of optimization is repeated until it satisfies the termination criterion. The detailed description of all three phases of the SOS algorithm is explained in the subsequent sections.

4.1 The Mutualism Phase

An association between two organisms of dissimilar species results into individual benefits of the symbiotic interaction is called mutualism. A typical example for mutualism the symbiotic interaction between the bee and the flower. In this phase, the design vector (X_i) of organism '*i*' (i.e. population) interacts with a randomly chosen organism '*k*'s design vector (X_k) of the ecosystem (where $k \neq i$). This mutualistic relationship improves individual functional values of the organisms in ecosystem. Therefore, new organisms are governed by a Mutual Vector (MV) and Benefit Factors (BF1 and BF2). The organism with the best functional value is considered as the best organism (X_{best}) of ecosystem. In this phase, organisms 'X_i' and 'X_k' also interact with the best organism. The organism is updated if its new fitness value is better than existing only. The mathematical formulations of the new solutions are given in Equations (1) and (2).

$$X_i' = X_i + rand * (X_{best} - MV * BF_1) \quad (1)$$
$$X_k' = X_k + rand * (X_{best} - MV * BF_2) \quad (2)$$
$$MV = mean(X_i, X_k) \quad (3)$$
$$BF_1 = 1 \text{ or } 2 \quad (4)$$
$$BF_2 = 1 \text{ or } 2 \quad (5)$$

4.2 The Commensalism Phase

When a relationship established by an organism with another organism of a different species results into benefits for this organism while having no influence on the other organism, such symbiotic interaction is called commensalism. The relationship between the remora fish and sharks is a standard example of commensalism phenomenon. In this commensalism phase, design vector (*Xi*) of the organism '*i*' (i.e. population) interacts a randomly chosen organism '*k*'s design vector (X_k) of the ecosystem (where $k \neq i$). This commensalism relationship improves the functional value of the organism 'i'. However, the organism '*k*' has neither benefit nor loss from this relationship. Moreover, the organism '*Xi*' also interacts with the best organism of the ecosystem. The organism is updated only if it's new fitness value is fitter than existing. The mathematical formulation of the new population is given in Equation (6).

$$X_i' = X_i + rand * (X_{best} - X_k) \quad (6)$$

4.3 The Parasitism Phase

A relationship established by an organism with another organism of a different species either benefits or harms the other organism, such symbiotic phenomenon is called parasitism. The symbiotic interaction between the plasmodium parasite and the anopheles mosquito is an example of this phenomenon. In this phase, the design vector (*Xi*) of the organism '*i*' (i.e. population) is assumed to be the anopheles mosquito. The anopheles mosquito produces an artificial parasite called *Parasite_Vector*. Parasite vector is produced by changing values of some randomly selected design variables of the organism '*X*i', the randomly selected design variables are modified using a random generated number within its bounds. Therefore, parasite vector is a fusion of design variables of the organism '*i*' and randomly generated design variables. The design vector (*Xk*) of a randomly selected organism '*k*' of the ecosystem (where $k \neq i$) works as a human host to the parasite vector. The interaction between these organisms results in a parasitism relationship. If the parasite vector has a better functional value than functional value of organism '*k*', the parasite will kill organism '*k*' and acquire its position in the ecosystem. If the functional value of organism '*k*' is better, organism '*i*' will have immunity from the parasite and the parasite will die. The pseudo code of the above explanation is shown in fig.2.

- Initialization
- REPEAT
    - Mutualism phase
    - Commensalism phase
    - Parasitism phase

UNTIL (termination criterion is met)

Fig. 2.Pseudo code for SOS algorithm

5. Results and discussion

Initially makespan optimization in the FMS by concurrent scheduling of jobs, tools and machines without considering AGVs and tool transfer times has been carried out by the proposed algorithm. Totally 22 job sets are considered in the work and the data of these job sets is given in [11]. The job sets with different number of jobs, machines and tools with its processing times have been taken into account to test the efficacy of the proposed SOS method. The results are compared with the results of existing methods [11] and are shown in Table 2.

From Table 2 it is obvious that the proposed SOS method is yielding better results. The minimum makespan is represented with bold letters and it is observed that the proposed SOS algorithm outperforms all the existing methods for all the 22 job sets. For majority job sets improvement is noticed. The proposed method has given same result for 7 job sets out of 22 job sets. For job set 22 the improvement is maximum and is 45.43%. For job set 21 the improvement is 44.16%.

The same algorithm is now applied for scheduling jobs, machines and tools by considering AGVs and tool transfer times between machines. The transfer times of tools between machines are taken as 70 percent of AGV travelling times. It is tested on first 10 job sets of aforementioned 22 job sets with four different layouts(LY1,LY2,LY3 and LY4) and with different processing times. These are the bench mark instances in the literature [11]. These results are presented in Table 3.Three cases are considered here to show the influence of AGVs and tool transfer times on makespan with different processing times and travelling times. In case I original processing times and AGVs travelling times are used. In case II processing times are taken as double the original processing times and AGV travelling times are takes as half of the original travelling times. And in case III processing times are taken as triple the original processing times and AGV travelling times are takes as half of the original travelling times. In all the above three cases tool transfer times remain same. The Gantt chart for optimal sequence produced by SOS algorithm for job set 5 and layout 2 in case I is shown in fig. 4. The operations that are assigned to each machine as well the start and finish times of each operation are shown in the Gantt chart. Utilization of tools for various operations of jobs are also shown in the Gantt chart. The Gantt chart also indicates loaded trip times, empty trip times and waiting times of AGVs and TT. The loaded trips are labeled as 'L', empty trips are labeled as 'E' and waiting times are labeled as 'W' in fig. 3. The Gantt chart shows the correctness of the solution provided by the proposed SOS method.

6 Conclusion

Scheduling of jobs, machines and tools without considering tool transfer times is performed with the proposed SOS algorithm. It is noticed that SOS algorithm outperforms the other algorithms in minimizing makespan without considering AGVs and tool transfer times. The proposed algorithm is tested on 22 job sets to show its consistency. It is observed that AGVs and tool transfer times have a considerable impact on makespan in all three cases and hence any schedule without considering AGVs and tool transfer times cannot be implemented in reality. The work can be extended further by considering downtime and AGVs dispatches for battery change.

Table 2: Makespan comparision of proposed method and existing methods for 22 job sets

Job set	MWR	LWR	LPT	SPT	MOR	LOR	ND-MWR	ND-LWR	ND-LPT	ND-SPT	ND-MOR	ND-LOR	MGTA	AIS	SOS	% Improvement
1	104	116	77	100	125	116	90	101	86	83	88	73	77	69	69	0.00
2	112	133	111	125	153	133	96	107	98	90	98	87	90	82	80	2.44
3	90	151	148	139	121	141	87	115	108	105	115	105	87	80	80	0.00
4	80	152	119	132	77	154	74	83	89	73	73	78	84	72	61	15.28
5	72	105	78	66	60	87	66	66	75	66	72	75	66	48	48	0.00
6	100	109	100	100	108	140	95	95	95	104	115	101	98	95	88	7.37
7	120	150	107	139	89	127	84	74	101	74	74	84	87	74	70	5.41
8	215	213	211	204	215	213	160	153	153	151	154	165	204	145	131	9.66
9	182	146	184	156	158	158	139	126	160	134	144	130	145	122	113	7.38
10	239	238	244	183	224	217	164	152	182	158	165	164	158	149	136	8.72
11	128	218	153	171	150	153	105	137	109	124	142	101	104	96	91	5.21
12	134	134	134	134	134	134	71	83	77	76	75	82	72	71	65	8.45
13	209	226	195	204	211	236	137	152	139	161	166	136	161	126	113	10.32
14	100	160	121	127	105	132	84	92	82	71	83	84	132	70	70	0.00
15	140	177	162	178	177	184	104	130	119	129	139	106	127	104	100	3.85
16	123	137	96	114	108	123	89	83	90	88	90	80	89	75	75	0.00
17	109	89	75	121	94	99	74	82	76	82	81	74	83	72	61	15.28
18	82	146	98	116	130	151	71	68	88	81	86	70	79	64	64	0.00
19	113	187	148	142	157	187	91	109	109	127	121	104	127	89	89	0.00
20	115	183	163	139	172	160	107	125	115	114	127	112	98	107	92	14.02
21	862	876	755	655	709	623	652	661	661	623	623	644	626	582	319	45.19
22	779	1300	768	757	1264	1255	784	742	781	748	780	804	787	733	409	44.20

Table 3: Makespan for job sets with and without AGVs and tool transfer times

Job set	Makespan Without AGVs &Tool Transfer Times		Makespan with AGVs &Tool Transfer Times by SOS								
			Case I				Case II				Case III
	AIS	SOS	LY1	LY2	LY3	LY4	LY1	LY2	LY3	LY4	LY4
1	69	69	124	105	103	139	182	161	161	182	------
2	82	80	129	101	116	149	191	174	184	199	271
3	80	80	134	106	115	151	194	181	183	204	273
4	72	61	142	117	125	160	162	154	161	186	238
5	48	48	101	84	88	116	137	120	137	147	194
6	95	88	127	105	108	137	192	183	186	200	------
7	74	70	139	111	118	158	185	171	181	200	194
8	145	131	173	156	163	191	284	273	275	291	------
9	122	113	156	140	144	151	258	252	253	254	------
10	149	136	196	176	183	210	306	308	314	322	------

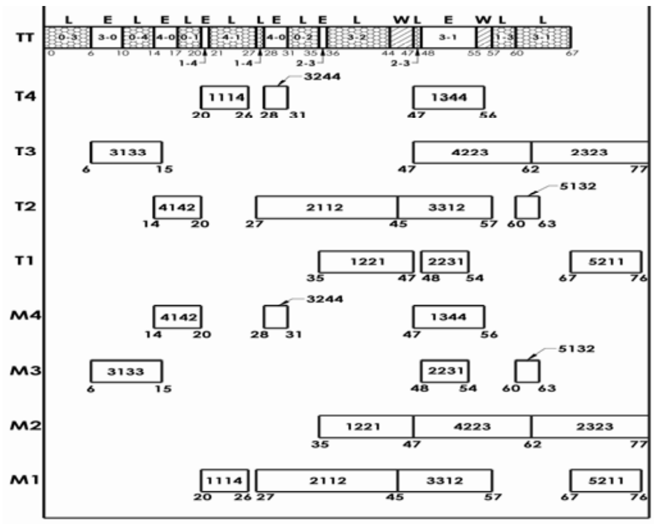

Fig. 3: Gantt chart for job set 5 and layout 2 in case I

References

[1] Agnetis, A., A. Alfieri, P. Brandimarte, and P. Prinsecchi. "Joint Job/Tool Scheduling in a Flexible Manufacturing Cell with No On-Board Tool Magazine." *Computer Integrated Manufacturing System* **10 (1)**,pp. 61–68(1997).

[2] Jerald, J., and P. Asokan."Simultaneous Scheduling of Parts and Automated Guided Vehicles in an FMS Environment using Adaptive Genetic Algorithm." *International Journal of Advanced Manufacturing Technology,* **29 (5)**,pp. 584–589 (2006).

[3] Jun, H., Y. Kim, and H. Sub. "Heuristics for a Tool Provisioning Problem in a Flexible Manufacturing System with an Automatic Tool Transporter." *IEEE Transactions on Robotics and Automation,* 15 (3), pp-488–497 (1999).

[4] Sureshkumar, N., and R. Sridharan. "Simulation Modeling and Analysis of Tool Flow Control Decisions in Single Stage Multimachine Flexible Manufacturing System." Robotics and Computer Integrated Manufacturing 23,pp-361–370(2007).

IOP Conf. Series: Materials Science and Engineering **225** (2017) 012294 doi:10.1088/1757-899X/225/1/012294

[5] Sureshkumar, N., and R. Sridharan "Simulation Modeling and Analysis of Tool Flow Control Decisions in a Flexible Manufacturing System." *Robotics and Computer Integrated Manufacturing* **25**,pp.829–838 (2009).

[6] Prabaharan, T., P. R. Nakkeeran and N. Jawahar "Sequencing and Scheduling of Job and Tool in Flexible Manufacturing Cell." *International Journal of Advanced Manufacturing Technology,* **29 (3)**, pp.729–745 (2006).

[7] Udhayakumar, P., and S. Kumanan. "Sequencing and Scheduling of Job and Tool in Flexible Manufacturing System Using Ant Colony Optimization Algorithm." *International Journal of Advanced Manufacturing Technology,***50 (9),**pp.1075–1084 (2010).

[8] J.Aldrin Raj, D.Ravindran, M.Saravanan and T.Prabaharan "Simultaneous scheduling of machines and tools in multimachine flexible manufacturing system using artificial immune system algorithm" *International Journal of Computer Integrated Manufacturing,***27(5)**,pp.401-414 (2014).

[9] N.Raman, F.B Talbolt, R.V.Rachamadugu, (**1986**) Simultaneous scheduling machines and material handling devices in automated manufacturing,Proceedings of the Second ORSA/TIMS Conference on Flexible Manufacturing Systems.

[10] Ulusoy G, Bilge U (1993) Simultaneous Scheduling of machines and automated guided vehicles. Int J Prod Res 31(12):2857–2873

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

[12] Abdelmaguid TF, Nasef AO, Kamal BA, Hassan MF (2004) A hybrid GA / heuristic approach to the simultaneous scheduling of machines and automated guided vehicles. Int J Prod Res 42 (2):267–281

[13] Min-Yuan Cheng ,Doddy Prayogo, "Symbiotic Organisms Search: A new metaheuristic optimization algorithm" *Computers and Structures* 139 ,pp.98–112(2014).