IJITR
R.Sudha* et al.
(IJITR) INTERNATIONAL JOURNAL OF INNOVATIVE TECHNOLOGY AND RESEARCH
Volume No.2, Issue No. 4, June – July 2014, 1089 – 1097.
Research Article

# An Intelligent Hybrid Scheduling Algorithm for Computer Aided Process Control of Manufacturing System

**R.SUDHA**
Associate Professor
Department of computer Applications
Vasavi College of Engineering
Ibrahimbagh,Hyderabad
Andhra Pradesh, India.

**P.PREMCHAND**
Professor
Department of Computer Science & Engg.
University college of Engineering
Osmania University, Hyderabad,
Andhra Pradesh, India

*Abstract-* **In recent times, maintaining stable and efficient operation, industrial automation and control systems that quickly respond to change is become a tedious task. Although the purpose of process scheduling is different according to the classes of process, the conventional methods have scheduled every process equivalently because they do not know the classes of process. To overcome this limitation, intelligent process scheduling method has to be developed to help the complexity associated with industries. In this paper, an intelligent algorithm is developed to do process scheduling of manufacturing system. Here, the proposed method utilizes a recent soft computing algorithm called, cuckoo search and traditional algorithm, called genetic algorithm. These two algorithms are effectively combined to do intelligent process scheduling. Initially, solutions are encoded effectively by considering the sequential order, set up selection and machine selection. Solution is nothing but the order of process to be carried out sequentially by considering machine availability, set up condition and predefined order of machine ordering. Then, the fitness of the solution are found out using the fitness that considers machine cost of doing task, set up cost and machine change cost. After designing the solution coding and fitness function, the intelligent scheduling will be done with the help of HCGA algorithm which is developed by combining cuckoo search and genetic algorithm. The experimental results showed that, the proposed approach gives fitness rate of 0.82 and which helps to achieve the scheduling in limited time, listed as 22000 sec on an average.**

## I. INTRODUCTION

Modern industrial systems are by nature, distributed, concurrent and stochastic. While maintaining stable and efficient operation, industrial automation and control systems that quickly respond to change. The basic needs for industrial automation and control systems remained relatively consistent with Christensen's list. Ultimately, the goal of the work on reconfiguration in industrial systems is to achieve self-adaptive software. In recent years, Industrial automation has advanced rapidly, yet despite this, further work is required if the goal of self-adaptive software or intelligent reconfiguration is to be realized. As software begins to play an increasingly important role in industrial automation and control systems, it becomes increasingly important to understand the risks associated with these systems and how these risks can be managed [2]. Consequently, to allocate appropriate resources for the required manufacturing tasks and to identify the sequence and timing parameters to accomplish these tasks, production scheduling plays an important role. Limited resources include facilities, personnel, materials, and so on. A good schedule can reduce the efforts in manufacturing, thus improving the competitiveness of products [3].

Although the purpose of process scheduling is different according to the classes of process, the conventional methods have scheduled every process equivalently because they do not know the classes of process. To overcome this limitation, an intelligent process scheduling method has been developed [1, 4]. Here, the intelligent production scheduling is conducted using artificial intelligent techniques in the multi-agent environment to allocate available resources and to the required tasks considering design and manufacturing constraints. Some authors utilized heuristic search algorithm for the intelligent scheduling mechanism search and agent-based negotiation approaches [3]. In heuristic search-based production scheduling, each state represents a partial schedule developed so far using heuristic search. A start state is an empty schedule, while a goal state is the schedule in which all the tasks have been allocated with required resources and timing parameters. These states are described as the nodes in a search tree. So, this heuristic method plays an important role in finding the solution [3].

During the last decade, Genetic Algorithms, a kind of heuristic search algorithm have been applied to many combinatorial problems like process planning. These include job shop scheduling [6, 7], A process planning

  

Research Article

problem (PPP) is similar to a scheduling in that every operation has to be traversed once and once only, although a PPP is more complicated due to precedence constraints among operations and non-fixed "distance" between operations (time required for machine, setup, and tool change). It is expected that GA's can provide a valid option for solving the PPP's so long as a suitable string representation and a corresponding search operator can be devised. Recently, there have been reports on applying GA's to process planning [8]. These developed GA's, however, still suffer from the above-mentioned problems [9]. So, to address the challenges, effective and hybrid algorithms are need in manufacturing system to do the task of process scheduling in an optimal way.

The proposed approach utilizes a recent soft computing algorithm called, cuckoo search and traditional algorithm, called genetic algorithm. These two algorithms are effectively combined to do intelligent process scheduling. Initially, solutions are encoded effectively by considering the sequential order, set up selection and machine selection. Solution is nothing but the order of process to be carried out sequentially by considering machine availability, set up condition and predefined order of machine ordering. Then, the fitness of the solution are found out using the fitness that considers machine cost of doing task, set up cost and machine change cost. After designing the solution coding and fitness function, the intelligent scheduling will be done with the help of HCGA algorithm which is developed by combining cuckoo search and genetic algorithm.

The main contributions of proposed approach are,

- An intelligent process scheduling method is formulated by combining two algorithms
- The cuckoo search and genetic algorithm are combined to form the proposed approach
- The features of both cuckoo search and genetic algorithm is combined to generate the proposed process scheduling methodology

The rest of the paper is organised as the second section list the review of recent researches, third section includes the problem description, 4th section includes the proposed methodology, 5th section is supplied with experimental results and by 6th section we conclude the paper.

## II. REVIEW OF RELATED WORKS

Literature presents several algorithms for process scheduling in manufacturing system using various systematic algorithms. Here, we review some of the algorithms presented by the researchers recently with the help of heuristic search algorithms. Jun Sun *et al* [3] have introduced an intelligent production scheduling mechanism for an integrated design and manufacturing system. The constraints in design and manufacturing phases were considered for generating the optimal production schedule. Design constraints were modeled based upon a feature-based product representation scheme. Manufacturing constraints have been described as available resources including facilities and persons. Manufacturing requirements for producing a product were also defined using the feature-based product representation scheme. The intelligent scheduling mechanism was developed based upon heuristic search and agent-based negotiation.

Ari Heikkila and Heikki Koivo *et al.* [5] have presented a modular control system of a flexible manufacturing cell based on microcomputers and local area network. Decentralization and modularization as architectural strengths of advanced Shop Floor Control-software were emphasized. The system implementation with corresponding control modules has been described. An expert routing and scheduling system using rule-based simulation was also presented. The use of simulation module tightly connected to the manufacturing process to increase the system throughput was suggested.

Yee-Ming Chen and Shih-Chang Wang [1] have addressed a collaborative framework of a distributed agent-based intelligence system to control and resolve dynamic scheduling problem of distributed projects for practical purposes. A two-stage decision-making process is done with the help of 1) the fuzzy decision-making process and, 2) the compensatory negotiation process. The first stage determined which behavior strategy would be taken by agents while delay event occurs, and prepares to next negotiation process; then the compensatory negotiations among agents were opened related with determination of compensations for respective decisions and strategies, to solve dynamic scheduling problem in the second stage. A prototype system was also developed and simulated with a case to validate the problem solving of distributed dynamic scheduling in the framework.

Yahong YANG *et al* [10] have focused on the dynamic re-configuration and task optimization of holonic manufacturing systems (HMS). The concept of dynamic virtual clustering was extended to the control process of a holarchy or holonic organization. The mediator-based dynamic virtual clustering mechanism have been presented. Then a negotiation strategy based on the Contract Net protocol was made for cooperative action among holons. Finally, a Hybrid Algorithm Based on PSO and Simulated Annealing for Holon task allocation have been described to support the optimum organization of a holarchy. PSO employs a collaborative population-based search, which was inspired by the social behavior of bird flocking. It combines local search(by self experience) and global search(by neighboring experience), possessing high search efficiency. SA employs certain probability to avoid becoming trapped in a local optimum and the search process could be controlled by the cooling schedule. The hybrid algorithm combines the high speed of PSO with the powerful ability to avoid being trapped in local minimum of SA.

Orides Morandin Jr *et al* [11] have found out that the problem for scheduling the manufacturing systems production involves the modelling task and the application of a technique to solve it. There are some ways used to model such problem and some search strategies have been applied on the model to find a solution. The solution has to consider performances parameters like makes pan or another. However, depending on the size and complexity of the system, the response time becomes critical, mostly when it's necessary a reschedule. Researches aim to use Genetic Algorithms as a search method to solve the scheduling problem. These works have been implemented for the use of Genetic Algorithm to solve this problem having as performance criteria the minimum makes pan and the response time.

Guohui Zhang *et al* [12] have developed, a particle swarm optimization (PSO) algorithm and a tabu search (TS) algorithm which have been combined to solve the multi-objective FJSP with several conflicting and incommensurable objectives. PSO which integrates local search and global search scheme possesses high search efficiency. And, TS was a meta-heuristic which was designed for finding a near optimal solution of combinatorial optimization problems. Through reasonably hybridizing the two optimization algorithms, an effective hybrid approach for the multi-objective FJSP has been developed. The computational results have shown that the proposed hybrid algorithm was an efficient and effective approach to solve the multi-objective FJSP, especially for the problems on a large scale.

Lixin Tang and Xianpeng Wang [13] have presented an improved particle swarm optimization (PSO) algorithm for the hybrid flowshop scheduling (HFS) problem to minimize total weighted completion time. This problem has a strong practical background in process industry. For example, the integrated production process of steelmaking, continuous-casting, and hot rolling in the iron and steel industry, and the short-term scheduling problem of multistage multiproduct batch plants in the chemical industry could be reduced to a HFS problem. To make PSO applicable in the HFS problem, they have used a job permutation that was the processing order of jobs in the first stage to represent a solution, and construct a greedy method to transform this job permutation into a complete HFS schedule. In addition, a hybrid variable neighborhood search (VNS) incorpo-rating variable depth search, a hybrid simulated annealing incor-porating stochastic local search, and a three-level population up-date method were incorporated to improve the search intensification and diversification of the proposed PSO algorithm.

Y.W. Guo *et al* [14] have presented, a unified representation model for Integrated Process Planning and Scheduling (IPPS). Based on this model, a modern evolutionary algorithm, i.e., the Particle Swarm Optimisation (PSO) algorithm has been employed to optimise the IPPS problem. To explore the search space comprehensively and to avoid being trapped into local optima, the PSO algorithm has been enhanced with new operators to improve its performance and different criteria, such as makespan, total job tardiness and balanced level of machine utilisation, have been used to evaluate the job performance. To improve the flexibility and agility, a re-planning method has been developed to address the conditions of machine breakdown and new order arrival. Case studies have been used to verify the performance and efficiency of the modified PSO algorithm under different criteria. A comparison has been made between the result of the modified PSO algorithm and those of the Genetic Algorithm (GA) and the Simulated Annealing (SA) algorithm respectively, and different characteristics of the three algorithms were indicated.

## III. PROBLEM DESCRIPTION

The main intention of my research is to design and develop a technique for computer-aided process control and planning through the help of intelligent scheduling algorithm in manufacturing industries. In manufacturing industries, every machine can do multiple tasks and each process (may be product) need multiple sequence of steps that needs multiple machines. So, allocation of machines to proper processes is increasingly demand in manufacturing system since revenue is completely around in scheduling mechanism. But the major problem here is that handling of real time constraints and criteria such sequential order, set up selection and machine selection. By handling all these criteria, an intelligent scheduling algorithm is urgently needed for manufacturing industries with better process planning in a concurrent manner. Considering the challenges listed above, an intelligent algorithm is developed to do process scheduling of manufacturing system. Here, we have planned to utilize a recent soft computing algorithm called, cuckoo search and traditional algorithm, called genetic algorithm.

## IV. PROPOSED INTELLIGENT METHOD FOR PROCESS SCHEDULING

In the proposed approach, we planned to utilize a recent soft computing algorithm called, cuckoo search and traditional algorithm, called genetic algorithm.

These two algorithms are effectively combined to do intelligent process scheduling. Initially, solutions are encoded effectively by considering the sequential order, set up selection and machine selection. Solution is nothing but the order of process to be carried out sequentially by considering machine availability, set up condition and predefined order of machine ordering. Then, the fitness of the solution are found out using the fitness that considers machine cost of doing task, set up cost and machine change cost. After designing the solution coding and fitness function, the intelligent scheduling will be done with the help of HCGA algorithm which is developed by combining cuckoo search and genetic algorithm. The proposed system can be briefed as a system, which will utilize all the machines used for the manufacturing process in such way that, the most efficient task by the machine is executed in priority. The machines are designed to process some particular tasks. The efficiency of the machines will depent upon the type of tasks they are handling.

According to the proposed approach, some of the definitions are given by, a machine is one part which is used to do the tasks according to the process. A process is defined as a set of task and the completion of a process is depending up on the completion of tasks that possessed by it. The tasks are defined as the basic functional unit of the proposed system. The proposed system is expressed in terms of a sample working model. The sample working model contains, 9 tasks listed as,

$$T = [t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9]$$

The system requires five processes to produce the expected outcome and have seven machines to process the nine tasks. Each of these machines are specialized in doing certain task and that can be listed as,

$$m1 = [t_1, t_7]$$
$$m2 = [t_2, t_8]$$
$$m3 = [t_3, t_9]$$
$$m4 = [t_4, t_1]$$
$$m5 = [t_5, t_4]$$
$$m6 = [t_6, t_5]$$
$$m7 = [t_7, t_2]$$

The above list indicates the machines and their specialized tasks. Here, m is represented as the machined and t represents the tasks. The tasks are assigned to different processes according to which the whole system works. The process includes a set of tasks and which should completed in specific time. The proposed approach is to maximize the efficiency of this process by effectively scheduling the tasks through machines. So, we propose a cuckoo algorithm and genetic algorithm to the scheduling. The 9 tasks are scheduled initially as follows,

$$p1 = [t_1, t_7, t_2, t_3]$$
$$p2 = [t_1, t_4, t_5, t_6]$$
$$p3 = [t_3, t_4, t_5, t_6]$$
$$p4 = [t_6, t_7, t_8, t_9]$$
$$p5 = [t_2, t_4, t_6, t_8]$$

The sample system under consideration uses five process listed as above and the task requirement is also listed near it. The work of the proposed system is to generate a scheduling of the tasks by processing it with the machines. The machine is specialized in doing some task, which we already discussed in the above section. The particular feature of the machine is that, it has a time schedule for processing particular tasks, so we need to feed the tasks to the machines by considering its time quantum. For example, if we consider machine m1, the tasks associated with the machine are t1 and t7. The time for processing t1 by m1 is 10sec and t7 by m1 is 6 sec. similarly, every machine is also possessing particular time quanta for each tasks. The proposed approach has to identify the least time of each machines to process a particular tasks and engage such machine to do the processes. There is also another parameter associated with the machines, known as the quality of task. The quality of task may be different for different machines. In other words, consider the task t1, which is been done by machines m1 and m4. The machine m1 processes the t1 at quality 2 and m4 processes t1 at quality 3. So the proposed approach has to concentrate on the quality parameter also to effectively generate the schedule.

***TABLE.1. PROCESSING TIME OF EACH TASK***

| Tasks/machines | M1 | M2 | M3 | M4 | M5 | M6 | M7 |
|---|---|---|---|---|---|---|---|
| T1 | 10 | | | 8 | | | |
| T2 | | 10 | | | | | 11 |
| T3 | | | 2 | | | | |
| T4 | | | | 2 | 15 | | |
| T5 | | | | | 15 | 8 | |
| T6 | | | | | | 13 | |
| T7 | 10 | | | | | | 2 |
| T8 | | 5 | | | | | |
| T9 | | | 10 | | | | |

***TABLE.2. QUALITY OF TASKS***

| Tasks/machines | M1 | M2 | M3 | M4 | M5 | M6 | M7 |
|---|---|---|---|---|---|---|---|
| T1 | 2 | | | 3 | | | |
| T2 | | 3 | | | | | 2 |

R.Sudha* et al.
(IJITR) INTERNATIONAL JOURNAL OF INNOVATIVE TECHNOLOGY AND RESEARCH
Volume No.2, Issue No. 4, June – July 2014, 1089 – 1097.

**Research Article**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| T3 | | | 5 | | | | |
| T4 | | | | 5 | 2 | | |
| T5 | | | | | 3 | 2 | |
| T6 | | | | | | 5 | |
| T7 | 2 | | | | | | 1 |
| T8 | | 2 | | | | | |
| T9 | | | 5 | | | | |

The tables 1 and 2 represent the processing time list and quality of task list of the machines and tasks of the sample system under consideration. Now the system deals with the processes in action. The process, as discussed above, contains one or more tasks and which are associated with the machines. The idea of the proposed method is to conduct each process by effectively utilizing the machines with ability to process the tasks. Consider the table

***TABLE.3. MACHINE ALLOCATION***

| Tasks/ Process | P1 | P2 | P3 | P4 | P5 |
|---|---|---|---|---|---|
| T1 | M1 | M4 | | | |
| T2 | M2 | | | | M7 |
| T3 | M3 | | M5 | | |
| T4 | | M4 | M5 | | M4 |
| T5 | | M5 | M6 | | |
| T6 | | M6 | M6 | M6 | M6 |
| T7 | M1 | | | M7 | |
| T8 | | | | M2 | M2 |
| T9 | | | | M3 | |

The table 3 represents the machines allocation in reference to the different process. In the sample we considered, each process possesses four different tasks and according to the task in the process, the machines are allocated. As each machine is associated with a time span to execute a process, the tasks that require same machines will form a queue and wait till the current task to be finished by the machine. Consider the following,

$$P1 \xrightarrow{conisists\ \ of} [t1, t2, t3, t7]$$

$$p1 \xleftarrow{requires} [m1, m2, m3]$$

$$t1, t7 \in m1\ \&\ t2 \in m2, t3 \in m3$$

$$while\ \ m1 \xrightarrow{executes} t1, t7\ will\ \ be\ in\ queue\ \ to\ m1$$

Once all the processes are allocated with machines and tasks, the proposed method subjects optimization process for scheduling the process and machines. The scheduling is provided to efficiently executing a process by make use of the machines with its best time to execute a tasks. The major steps associated with the proposed approach are to associate a cuckoo formulae and mutation process to optimize the scheduling process. The scheduling process is also associated with the fitness of each process.

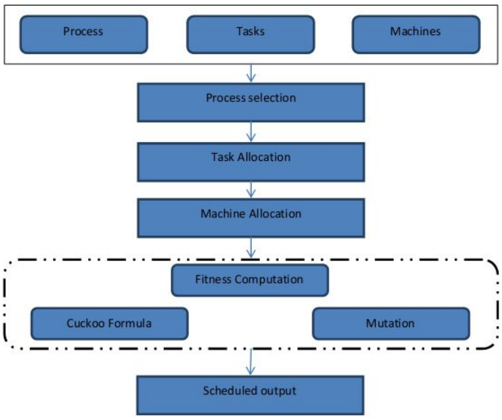


***Fig.1. Basic architecture of the proposed approach***

The figure 1 shows the basic architecture of the proposed approach. The main section or the key part of the proposed approach is the block followed by the machine allocation process. Although we have enough said on the machine allocation part and still it is complex one. The machine allocation is a tedious process to deal with in this part we actually do the optimization. In the machine allocation phase, each machine allocated according to the task that it can process. A machine can process only one task at a time, so the program has to be designed such way that the other task that requires the same machine should wait until the initial one gets finished. Once all the processes are allocated with the machines, we move one to the fitness computation process, which will be benchmark to analyse the efficiency of each machine.

### *4.1. Fitness Computation*

The fitness is defined as the measure to find the stability of the scheduling based on the entire processes in the defined system. According to the proposed approach, the fitness function is associated with parameters like tasks, maximum time taken by a machine, the quality of executing a task by a machine. The fitness is calculated by considering all the process as a single system. The fitness defined by proposed approach is given as,

$$Fitness = \alpha\left[\frac{MaxTime}{TotalTime}\right] + \beta\left[1 - \left(\frac{\sum Q}{N_t \times Q_{MaxAvg}}\right)\right]$$

Here, the MaxTime defines the maximum time of a machine; TotalTime defines the total time consumed by the whole process, Q represents the quality of the machines, $Q_{MaxAvg}$ defines maximum average quality and are $\alpha, \beta$ constants. According to the above equation, the quality of the whole process is calculated, for example,

  

Research Article

Let's assume time for each machine taken to complete the entire task. M1 = 48, M2 = 54, M3 = 10, M4 = 15, M5 = 60, M6 = 20, M7 = 8; Max Time = 60 (M5); Total Time = (M1 + M2 + M3 + M4 + M6 + M7) = 155; $N_t$ = 9 ( No : of Task). Let's assume M1 has Quality 2 in T1 and 4 in T7 , M2 has 5 in T3 and 5 in T4.

***TABLE.4***

| Tasks/machines | M1 | M2 | M3 | M4 | M5 | M6 | M7 |
|---|---|---|---|---|---|---|---|
| T1 | 2 | | | 3 | | | |
| T2 | | 3 | | | | | 2 |
| T3 | | | 5 | | | | |
| T4 | | | | 5 | 2 | | |
| T5 | | | | | 3 | 2 | |
| T6 | | | | | | 5 | |
| T7 | 2 | | | | | | 1 |
| T8 | | 2 | | | | | |
| T9 | | | 5 | | | | |

From the above table, we get, $\sum Q$ = 42; $Q_{MaxAvg}$ = 5; α = 0.6 (user-defined); β = 0.4 (user-defined). So based on these values the fitness becomes,

$$Fitness = 0.6\left[\frac{60}{155}\right] + 0.4\left[1 - \left(\frac{42}{9 \times 5}\right)\right] = 0.232 + 0.028 = 0.260$$

The above listed fitness is the value derived for the four processes that we mention in the example give above. The proposed approach will consider similar kind of process by varying their machines, i.e. the same task will be done with different set of machines. The major role of the defined approach is select best schedule among the different schedules that we consider.

### *4.2. Selecting the best schedules*

The main phase in the proposed approach is the selection of best schedules among the set of schedules generated. We generate a number schedules with process byvarying the machines in it, in order to find the efficient one. The proposed approach uses two different processes for finding the effective scheduling strategy. The initial one is addiction of cuckoo formulae to optimize the fitness values. We consider a set of schedules for the processes namely a set of 10 schedules. In each set there will be 4 processes and 4 tasks each for every process. The tasks are marked to be executed by different machines among the seven machines. The initial process in the selection of schedule is the fitness calculation. The fitness of each set process are calculated and stored. The fitness is calculated based on the fitness equation defined the above sections. Later on, the cuckoo formula is cooperated with the processes and fitness is again calculated based on the cuckoo formulae. Thus we get another 10 values for fitness values. The cuckoo formulae can be given as,

$$X_i^{t+1} = X_i^t + \alpha \oplus Levy(\lambda)$$

Here, the X represents the machine to be allocated for the task according to the cuckoo algorithms. $\alpha$ is a constant value for a single iteration and it will be greater than 0. $Levy(\lambda)$ is the constant by the definition of cuckoo algorithm. According to the proposed approach the cuckoo algorithm wil alter the machine which is to be used for processing a particular task. The value of $X_i^{t+1}$ represents the newly created machine value. The machine corresponding to this value will be used for executing the task possessed by the particular process. This will cause a significant change in the fitness calculation. Consider the following table,

***TABLE.5***

| Tasks/ Process | P1 | P2 | P3 | P4 | P5 |
|---|---|---|---|---|---|
| T1 | M1 | M4 | | | |
| T2 | M2 | | | | M7 |
| T3 | M3 | | M5 | | |
| T4 | | M4 | M5 | | M4 |
| T5 | | M5 | M6 | | |
| T6 | | M6 | M6 | M6 | M6 |
| T7 | M1 | | | M7 | |
| T8 | | | | M2 | M2 |
| T9 | | | | M3 | |

When considering the table 5, task t1 possess two machines, which can possibly execute T1. Now we provide index values for each machine and according to the index values cuckoo formulae generate a new machine index to execute T1. For T1, we consider the $X_i^t$ as index of initial machine which is used to execute T1 and as per the definitions of cuckoo search, the value $\alpha \oplus Levy(\lambda)$ becomes the second choice of machine, which can execute T1. Let us consider, index value of M1 is 1 and M4 is also 1. Thus according to cuckoo formulae,

$$X_i^{t+1} = X_i^t + \alpha \oplus Levy(\lambda) \Rightarrow 1+1 \Rightarrow 2$$

Thus, machine with index value two will be used for executing task T1. Similarly the entire tasks will be refreshed with different machine according to the above listed process. Because of this change in process, the fitness will have considerable change in the system. Thus we get ten new fitness values and now we make a comparison with basic fitness values and the fitness values obtained through cuckoo method. We select five set of process after comparison, which possess the better fitness in both streams.

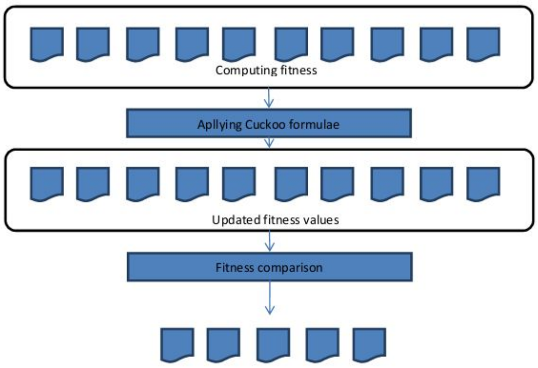


*Fig.2. cuckoo formulae*

The figure 2 represents processing of proposed approach based on the cuckoo formulae. After the selection of best five set of process from the comparison, it is time to apply the second major step of the proposed approach. The selected five set of process will undergo on more update in the machine allocation part. The process we add is adopted from the genetic algorithm and the process is termed as mutation. In mutation step, we replace any one of the machine possessed by the task with another. I.e. the task is designed to be processed by one particular machine and that will be replaced by one machine from the list of machines randomly. The mutation will help the process to be more efficient as the change in machine on a task is expected to deliver improved performance. The mutation is also done in an iterative manner on the processing.

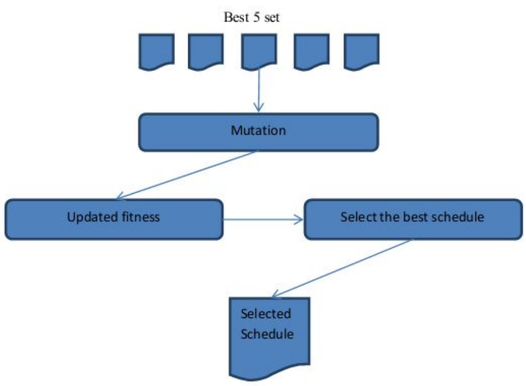


*Fig.3. Mutation process*

The figure 3 shows the application of mutation process in the proposed approach. In the above listed process, after applying the mutation on the process, we conduct the fitness calculation. As all the iterations come up to an end and the system will provide a schedule with optimized functionalities, which is intended by the proposed approach.

Algorithm.1.
Input: P, T, M, Q, $time_M$
Output: Process schedule

Step1. **Select** set of task, T,

$$T = [t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9]$$

Step2. **Select** set of machines, M

$$M = [m_1, m_2, m_3, m_4, m_5, m_6, m_7]$$

Step3. **Initialize** M, with process time, $time_{machine}$ and Q, quality of task
Step4. Populate P, processes list
Step5. For each element in P, assign $t_i \in T$
Step6. Define fitness function

$$Fitness = \alpha \left[ \frac{MaxTime}{TotalTime} \right] + \beta \left[ 1 - \left( \frac{\sum Q}{N_t \times Q_{MaxAvg}} \right) \right]$$

Step6. Calculate fitness for entire set of P
Step7. Store the fitness in a set, $F_i$
Step8. Update the process with cuckoo formulae,
$X_i^{t+1} = X_i^t + \alpha \oplus Levy(\lambda)$
Step9. Update the fitness values, and store in $F_{new}$
Step10. Compare $F_i$ and $F_{new}$
Step11. Select best values after comparison
Step12. Apply mutation
Step13. Update fitness and select the best
Step14. Continue the process till stopping criteria
Step15. End

***Psuedocode for the proposed approach***

## V. RESULTS AND DISCUSSION

In the above part of the paper include the proposed methodology of the scheduling process in system containing multiple tasks and multiple machines. The idea behind the proposed approach is to define a timely effective scheduling by utilizing the maximum efficiency of each machine in processing the tasks. The following section includes the experimental analysis of the proposed approach on sample dataset.

### *5.1. Experimental setup*

The whole system is designed and developed in java programming language under JDK 1.7.0. The systems used for the experimentation is powered with i5 processor,4 GB RAM and 500 GB hard disk.The proposed approach defines the sample system for the experimentation process. The defined system is constructed with seven machines and nine tasks. The nine tasks are randomly assigned to four distinct processes. In the input time, the processing time of each machines and quality of executing of process is also is defined from the user side. Later, the experimental results are analysed based on the iterations, time and fitness values.

Research Article

### *5.2. Performance Analysis*

The performance of the proposed approach is evaluated based on the fitness values under different iterations. The fitness of the system through a set of varied iterations is recorded to evaluate the performance of the proposed approach. The iteration used in the analysis range from 50 to 100. The corresponding fitness values are store and plotted in the following graph.

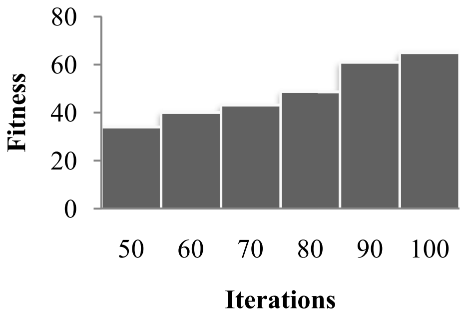


*Fig.4. performance based on fitness*

The figure 4 shows the average fitness values obtained by selecting all the set of process. The analysis from the figure shows that, as the iterations increases the fitness value also increases. We can make a conclusion that the proposed approach performs well at high iterations and will have a saturation point after 100.The following graphs, list the performance of a single system of process having four tasks.

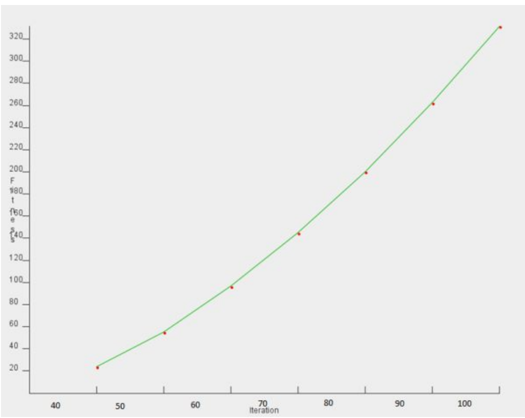


*Fig.5. fitness list*

Now, we discuss about performance based on the scheduling process. The parameter used for the analysis is the time consumed for the scheduling process. The analysis concentrates on two different types of time consumption based on the scheduling. The method concentrated on scheduling time of process and scheduling time of machines.

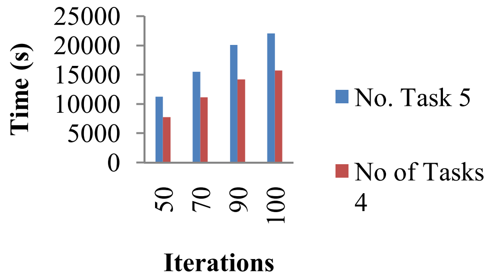


*Fig.6. Time regarding process*

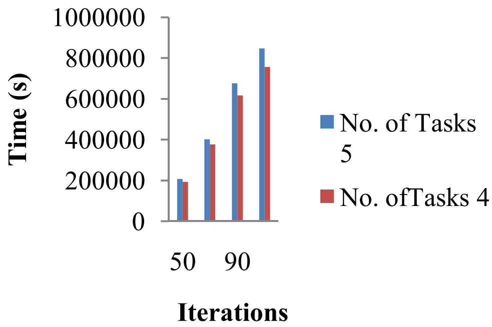


*Fig.7. Time regarding machine*

In figure 6 and 7, we represent the analysis of time based on the scheduling happened in process and machines. Considering the figure 6, we can see that, in similar to the fitness, as the iterations increases the time consumed is also increases. Even though, number of process may vary, there is no significant change in the time. As the program approaches to 100 iterations, the time consumed reaches about 900000, which significantly improves the scheduling process.

## VI. CONCLUSION

In this paper, an intelligent algorithm is developed to do process scheduling of manufacturing system. Here, the proposed method utilizes a recent soft computing algorithm called, cuckoo search and traditional algorithm, called genetic algorithm. These two algorithms are effectively combined to do intelligent process scheduling. Initially, solutions are encoded effectively by considering the sequential order, set up selection and machine selection. Solution is nothing but the order of process to be carried out sequentially by considering machine availability, set up condition and predefined order of machine ordering. Then, the fitness of the solution are found out using the fitness that considers machine cost of doing task, set up cost and machine change cost. After designing the solution coding and fitness function, the intelligent scheduling will be done with the help of HCGA algorithm which is developed by combining cuckoo search and genetic algorithm. The experimental results showed that, the proposed approach gives fitness rate of 0.82 and

  

Research Article

which helps to achieve the scheduling in limited time, listed as 22000 sec on an average.

## REFERENCES

[1] Yee-Ming Chen and Shih-Chang Wang, "Framework of agent-based intelligence system with two-stage decision-making process for distributed dynamic scheduling, " *IEEE Transactions on Applied Soft Computing*, vol.7, no.12, , 2007, pp.229–245

[2] R.W. Brennan, A. Zoitl, C.Sunder, T. Strasser, V. Marik, "Developments in dynamic and intelligent reconfiguration of industrial automation," *IEEE Transactions on Computers in Industry*, vol.20, no.12, 2008 , pp.533-547.

[3] Jun Sun, Deyi Xue, and Douglas H. Norrie, "An intelligent production scheduling mechanism considering design and manufacturing constraints," *Third International Conference on Industrial Automation*, vol.18, no.12, 1999,pp.651-667.

[4] Sungsoo Lim and Sung-Bae Cho, "Intelligent OS Process Scheduling Using Fuzzy Inference with User Models," *Proceedings of the 20th international conference on Industrial, engineering, and other applications of applied intelligent systems,* 2007,pp. 725-734.

[5] [5]Ari Heikkila and Heikki Koivo, "Modular control system with intelligent scheduling," *international conference on Computers in Industry*, vol 36, 1998,pp.75-81.

[6] L. Davis, "Job shop scheduling with genetic algorithms," *in Proc. 1st Int. Conf. Genetic Algorithms and Their Applications, J. J. Greferstette, Ed. Hillsdale, NJ: Lawrence Erlbaum,* 1985, pp. 136–140.

[7] J. E. Biegel and J. Davern, "Genetic algorithms and job scheduling," *Comput. Indust. Eng.,* vol. 19, 1990,pp. 81–91.

[8] D. Yip-Hoi and D. Dutta, "A genetic algorithm application for sequencing operations in process planning for parallel machining," *IIE Trans.,vol.* 28, 1996,pp. 55–68.

[9] F. Zhang, Y. F. Zhang, and A. Y. C. Nee, "Using Genetic Algorithms in Process Planning for Job Shop Machining", *IEEE Transactions on evolutionary computation*, Vol. 1, No. 4, November 1997.

[10] Yahong YANG Fuqing ZHAO , Yunping YAO and Aihong ZHU, "A PSO and Simulated Annealing Hybrid Algorithm to Task Allocation Problem for Holonic Manufacturing System,"*Proceedings of the 6th World Congress on Intelligent Control and Automation*, vol. 28, 2006,pp. 6767–6771.

[11] Orides Morandin Jr, Edilson R. R. Kato,Ana C. Deriz and Danilo S. Sanches, "A Search Method using Genetic Algorithm for Production Reactive Scheduling of Manufacturing Systems,"*International conference on Industrial electronics*, 2008,pp. 1843 – 1848.

[12] Guohui Zhang, Xinyu Shao, Peigen Li and Liang Gao, "An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem,"*International conference on Computers & Industrial Engineering*, vol. 56 , 2009,pp.1309–1318.

[13] Lixin Tang, Member and Xianpeng Wang, "An Improved Particle Swarm Optimization Algorithm for the Hybrid Flowshop Scheduling to Minimize Total Weighted Completion Time in Process Industry,"*IEEE transactions on control systems technology*, vol. 18,no.6, 2010,pp.1303–1314.

[14] Y.W. Guo, W.D. Li, A.R. Mileham, G.W. Owen,"Optimisation of Integrated Process Planning and Scheduling Using a Particle Swarm Optimisation Approach,"*International Journal of Production Research*, 2010, pp.1-55.