International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 - 5161

Available at http://inpressco.com/category/ijcet

Research Article

Iterative Genetic Algorithm for Task Scheduling in Non-Flexible Manufacturing Process

D.L. Bhombe[Ȧ] and N.B. Bhawarkar[Ȧ*]

[Ȧ]Department of Electronics & Telecommunication Engineering, SSGMCE Shegaon, Amravati University, India

Accepted 10 April 2014, Available online 16 April 2014, **Vol.4, No.2 (February 2014)**

Abstract

Every manufacturing system has its increasing demand for well planned and scheduled operations for various products. A Genetic algorithm observed its great application for finding the best schedule for the flexible manufacturing System (FMS). But these flexible operations are having big limitations for real time processing in wide industries. Every real time operation demands non-flexibility with tasks dependent on each other, for this case the machine is fixed for a particular operation jointly called as Task. Hence, for this case we can generate the optimal solution showing the Tasks which can be executed in parallel. This paper gives an iterative genetic algorithm (IGA) for achieving solution for these limitations and to find optimum schedule for the various tasks to minimize the total makespan.

Keywords: *Iterative Genetic Algorithm (IGA), Task Scheduling (TS), FMS, Optimized Task plan.*

1. Introduction

Many industries require their production without any penalties and due dates with minimum production costs. These have already being achieved by applying various scheduling criterion by Genetic algorithm to reduce the total makespan of the production (C. Moon, et al, 2004). But every algorithm has considered the production with flexible tasks and without any dependence where any task can be executed on any machine (KJ. Chen, et al, 2007). It is not the case for real time production where the number of tasks with its prescribed machines are used also there is precedence of tasks and thousands of tasks with dependence on each other. In these cases the algorithm with flexible operations fails to find the optimum solution of schedule.

It means for real time production we can simply achieve the scheduling for these machines with tasks to be executed in parallel with each other. In this paper, an iterative based genetic algorithm (IGA) is discussed to find the best schedule for non-flexible tasks. The optimal solution generated at the end of simulation shows only the Tasks which can be simply executed in parallel with best fitness value. An IGA is found to be effective one which generates the best chromosomes iteratively and increases its space to search beyond the conditional limit. In section 2 we have discussed the Task scheduling model for a given data set considered from one of the small industry, in section 3 we have discussed the precedence of Task, an iterative genetic algorithm (IGA) concept is discussed in section 4 and the algorithm and results are discussed in further sections.

*Corresponding author: **N.B. Bhawarkar**

2. Tasks Scheduling Model

This model shows the parameters used to minimize an objective function. The various notations are used to describe the problem.

i, j : index of tasks assigned to k machines

N : Number of tasks

C_{ki}= Completion time of each task

Pki=Processing time of task for each machine

T=Total makespan = $\sum_{i=1}^{N}$ Cki

The dataset collected from one of the industry is as shown in table 1 which requires scheduling.

Table 1 Total task showing their operation time

Task Number	Completion Time (min)	Task Number	Completion Time (min)
Task1	38	Task11	05
Task2	15	Task12	30
Task3	15	Task13	17
Task4	24	Task14	47
Task5	11	Task15	27
Task6	68	Task16	10
Task7	22	Task17	30
Task8	20	Task18	27
Task9	06	Task19	15
Task10	11	Task20	05

3. Task Precedence

These total 20 tasks with the precedence of execution are as shown in fig.1

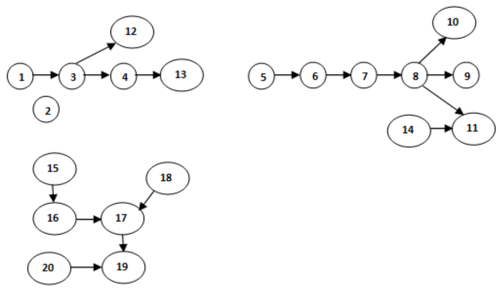

Fig.1 Precedence of tasks

4. Iterative Genetic Algorithm (IGA)

An iterative genetic algorithm is used here having following parameters. The main objective is to reduce the total makespan and is achieved by IGA. The IGA works mainly with initial chromosomes generation, fitness determination, and genetic operators (Azuma Okamoto, et al, 2010, C. Moon, et al, 2006).
Number of populations: 30
Number of iterations: 10

A] Chromosomes Generations

The chromosomes coding mainly generates the initial random chromosomes in terms of initial solutions for the number of tasks and populations. Two dimensional coding concept is applied here similar as mentioned by every researchers, it produces various solutions for execution of tasks.

B] Fitness Determination:

The fitness defines the measure of optimality of a chromosome for continuing in the next generation. It can be simply calculated as the reciprocal of the objective function.

fitness = cost/MaxCost;
mean_fitness = fitness/NUM_POPULATION

C] Genetic operators

The genetic operators are mainly Selection, Crossover and mutations are used here.

The selection operator finds the best chromosome having the best fitness among all the chromosomes generated. The probability of selection of a chromosome is given by

P = maximum fitness/(Sum of fitness for all chromosomes)

The crossover operator is the main operator which creates next generation and applies the search space in finding the best generations iteratively.

Finally, the mutation is applied which check the availability of the other optimal solution other than the first one by simply changing some positions at a very small probability.

5. Algorithm

procedure: Iterative Genetic algorithm for Task scheduling (TS)
input: TS data set, GA parameters.
output: best solution
begin
$t \leftarrow 0$;
Generate initial chromosomes P(t);
create Pareto E(P)
fitness F(P);
select the best solution σ* and its corresponding P(t)*
while (not termination condition) do
select P(t+1) from P(t) by roulette wheel selection;
crossover to P(t) yield C(t) by uniform
crossover;
mutation to P(t) yield C(t) by swap mutation;
mutation;
fitness F;
update the best solution σ* and P(t)*, i.e., if $\sigma < \sigma^*$,
then $\sigma^* = \sigma$, $P(t)^* = P(t)$
$t \leftarrow t+1$;
end
output Pareto optimal solution E(P);
end

Fig.2 Overall procedure for iterative genetic algorithm

6. Result

The simulation result is as shown in fig.3. It shows the actual makes span before scheduling, the makespan after scheduling, the optimized task plan and total percentage time saved after scheduling.

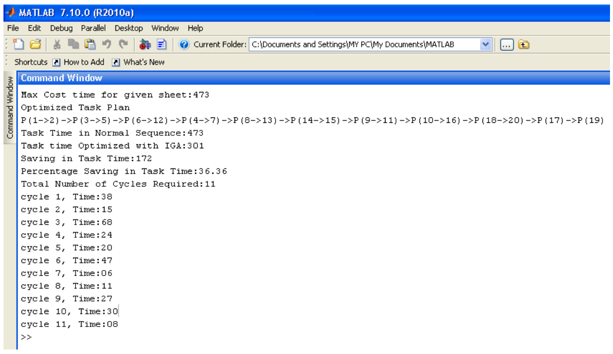

Fig.2 Simulation results showing optimized task plan.

Table 2 Result showing tasks can be executed in parallel.

Cycle Number	Task executed in Parallel
Cycle 1	Task 1 || Task 2
Cycle 2	Task 3 || Task 5
Cycle 3	Task 6 || Task 12
Cycle 4	Task 4 || Task 7
Cycle 5	Task 8 || Task 13
Cycle 6	Task 14 ||Task 15
Cycle 7	Task 9 || Task 11
Cycle 8	Task 10 || Task 16
Cycle 9	Task 18 || Task 20
Cycle 10	Task 17
Cycle 11	Task 19

Conclusions

The real time manufacturing problem to minimize the total makespan is having its wide need now a day. Here a concept based on an iterative genetic algorithm (IGA) for tasks scheduling is given where there is no non-flexibility of operations with machines. An optimized Task plan can be achieved which shows the tasks which can be executed in parallel with best fitness function. Hence an IGA can be applied to the big industries where there is thousands of Tasks with certain precedence of execution. IGA finds best solution within few seconds of time without any limitation of dependency and flexibility. For the data set considered here the IGA saves 36.36% time.

References

C. Moon, J. S. Kim, M. Gen (2004), Advanced planning and scheduling based on precedence and resource constraints for e-plant chains, International Journal of Production Research, vol. 42, no. 15, pp. 2941– 2954.

KJ. Chen, P. Ji (2007), Development of a genetic algorithm for scheduling products with a multilevel structure, International Journal of Advanced Manufacturing Technology, vol. 33, no. 8, pp. 1229–1236.

Azuma Okamoto, and Mitsumasa Sugawara (2010), Solving Composite Scheduling Problems using Hybrid Genetic Algorithm, The 11th Asia Pacific Industrial Engineering and Management System Conference.

C. Moon, Y. Seo, Y. Yun, M. Gen (2006), Adaptive genetic algorithm for advanced planning in manufacturing supply chain, Journal of Intelligent Manufacturing, vol. 17, no. 4, pp. 509–522, 2006.

H Zhang (2006), Analyzing the structure of Integrated Manufacturing System (IMS), and extraction of the mathematical model from the scheduling problems, D space at Waseda University.

J. Yang, W. Tang (2009), Preference based Adaptive Genetic Algorithm for Multi-objective Advanced Planning and Scheduling Problem, "IEEE international conference, pp-1935-1940.

D. Liu, Y. Pu, Y. Ji, Development of a multi-objective GA for advanced plan-ing and scheduling problem, International Journal of Production Research, in press.