Application of Genetic Algorithms to Alternative Operation Sequencing for Manufacturing Plants

Aslı Erdiller
Faculty of Engineering
Industrial Engineering Department
Marmara University
Goztepe Kampusu, 34722, Istanbul, Turkey
E-mail: *erdiller@isnet.net.tr*

Â. Yurdun Orbak
Faculty of Engineering and Architecture
Industrial Engineering Department
Uludağ University
Gorukle Kampusu, 16059, Bursa, Turkey
E-mail: *orbak@uludag.edu.tr*

Abstract

In manufacturing plants, it's difficult to prepare applicable schedules with the dynamic constraints of the plant, such as machine breakdowns, dynamic change of demand, etc. Thus, techniques that produce flexible and rapid response are often desirable. In this paper, a method for computing alternative machine sequences for manufacturing plants using genetic algorithms is analyzed. As objective functions makespan, number of tardy jobs and total tardiness are chosen. An example is given to illustrate the results.

1. Introduction

In literature, research on operation planning usually focuses on computer aided process planning (CAPP) or sequencing. Many problems have been constructed such as the 187 CAPP system, etc. [1]. There are also rule-based CAPP systems that can produce the best operation sequence based on local information [2]. In recent years new methods have been implemented such as the hybrid approach that combines the artificial intelligence and decision tree techniques [2, 3].

Furthermore genetic algorithms have been successfully used beginning with the works of Reddy et. al. [1] and Senin et. al.[4,5,6].

Genetic algorithms are optimization tools that simulate evolutionary processes in order to develop solutions for a wide range of optimization problems [7]. In recent years, they have increasingly gained acceptance in industry, as confidence improves in their abilities to solve complex problems that have previously appeared to be unsolvable. Much of the advantage of genetic algorithms lies in the flexibility with which they may be implemented [7].

In this paper, a method is proposed for the computation of alternative machine sequences for manufacturing plants using genetic algorithms. In this approach, makespan, number of tardy jobs and total tardiness are chosen as objective functions.

2. Operation sequencing

Products and their components are designed to perform some certain functions. Functionality aspects of the products and components guarantee the design specifications. Producing components that meet the design specification is main role of manufacturing. The produced components are assembled to form the final product. Process planning acts as a bridge between design and manufacturing by transferring design specifications into manufacturing process details. Therefore, process planning refers to asset of instructions that are used to make a component or a part so that the design specifications are met [1,8].

Some activities must be performed for generating a process plan [8];

1. Analysis of part requirements
2. Selection of raw work piece
3. Determining manufacturing operations and their sequences
4. Selection of machine tools
5. Selection of tolls, work holding devices and inspection equipment
6. Determining machining conditions such as cutting speed, feed, etc., and manufacturing times

Determining manufacturing operations and their sequences, and selection of machine tools can be considered as operations planning. First step of operations planning is to determine the essential types of operations and their sequence to transform raw work piece to a final product. There can be different ways to produce a design. Because of some constraints such as accessibility and setup, machining sequence of some feature may vary. Dependency of operations can also affect the operation sequence. Second step of operations planning is the selection of machine tools on which the operations can be performed.

3. Applied methodology

In this paper, a job shop system with m machines and n type of jobs with k operations is considered. For each type of job, operations are dependent. In the given system, it is assumed that machines can perform different types of operations, but machining speeds may vary. With this assumption, operations can be performed in each machine; machining speed and time is related with the machine. Orders, with their corresponding due dates, arrive to this system through job types.

Main objective for the given system is minimization of makespan, number of tardy jobs and total tardiness. In this paper, an elitist multi-objective genetic algorithm is proposed for the solution of the given problem. With the given algorithm, any of the given objective functions or a weighted sum of objective functions can be minimized.

With the proposed algorithm, active and non-delay schedules can be obtained. Makespan can be minimized by active and non-delay schedules. Minimizing makespan decreases machine idle times and therefore, increases machine utilization. In order to meet due date constraints, both number of tardy jobs and total tardiness must be minimized. Meeting due dates minimizes costs that occur by late delivery and also increases customer satisfaction.

For the given problem a weighted function of makespan, total tardiness and number of tardy jobs is formed as objective function [2]. Weights for the criteria are given as w_p (p=1,2,3), where $0 \le w_p \le 1$ and $\sum_p w_p = 1$. Objective function is given as follows,

$$Min\_Z = w_1 \cdot C_{\max} + w_2 \cdot \sum T_i + w_3 \cdot NT_i \tag{1}$$

where $C_{\max}$ is makespan, T_i is tardiness and NT_i is number of tardy jobs.

In the given objective function, makespan is expected to have larger values relative to other criteria, which will cause an undesired situation unless all criteria are normalized [6]. For this purpose, two normalization values are defined, largest total value and number of orders. Largest total value, *TD*, is the longest time needed to process all jobs and used for normalizing makespan and total tardiness. Number of orders, *NS* is used for the normalization of number of tardy jobs. Normalized objective function is given as (2).

$$Min\_Z = w_1 \cdot \frac{C_{\max}}{TD} + w_2 \cdot \frac{\sum T_i}{TD} + w_3 \cdot \frac{NT_i}{NS} \tag{2}$$

Like many of the scheduling and sequencing problems, this given problem is a minimization problem. On the other hand, the proposed algorithm is also suitable for the maximization problems. So, the given objective function is transformed to a fitness function f and stated as follows:

$$f = 1 - Z = 1 - \left(w_1 \cdot \frac{C_{\max}}{TD} + w_2 \cdot \frac{\sum T_i}{TD} + w_3 \cdot \frac{NT}{NS} \right) \tag{3}$$

4. Example

In order to determine the efficiency of the proposed algorithm, a shop with 4 machines is formed. It is assumed that 5 types of jobs, each with different number of operations, are processed in this shop. Details about operations and processing times are given Table 1.

As an example, an order list with 5 orders and its sequenced list obtained by the Hudgson's Rule is given in Table 2 [6].

Table 1. Shop floor details

Job Type	Oper. No.	Processing time			
		Mach. 1	Mach. 2	Mach. 3	Mach. 4
A	1	50	3	50	4
	2	5	2	3	50
	3	10	50	2	50
	4	50	3	50	4
	5	8	50	7	6
B	1	2	50	2	50
	2	50	5	4	50
	3	50	50	7	7
C	1	5	50	8	50
	2	10	12	50	10
	3	10	50	9	50
D	1	13	50	13	50
	2	2	3	50	5
	3	8	50	50	6
	4	50	10	12	50
E	1	50	8	50	11
	2	2	50	9	50

Table 2. Order data of example

Order Data		
Order No	Order Type	Due Date
1	C	200
2	B	100
3	B	25
4	D	30
5	E	150

Sequenced Order Data		
Order No	Order Type	Due Date
3	B	25
4	D	30
2	B	100
5	E	150
1	C	200

A simple binary representation is used for individual representation. Each couple of code indicates the machine number, and job type, operation number and machine number determine processing time of an operation. An example of an individual for the given example is shown in Table 3.

Table 3. Individual representation

Order no	3	4	2	5	1
Type	B	D	B	E	C
Individual	00 01 11	10 01 00 01	00 10 10	01 11	00 00 01
Mach. no	1 2 4	3 2 1 2	1 3 3	2 4	1 1 2
Proc. time	2 5 7	13 3 8 10	2 4 7	8 50	5 10 50

All operations of a job are dependent and must be processed in their given order. As seen in the given example, 3rd job, which is type B and has three operations, is the first job to be processed according to due date. In order to form a schedule, a matrix like chart is used. Starting from the beginning of the chart, all operations are placed in the corresponding machines with given processing times. Formation steps of given example is given in Figures 1 through 5.

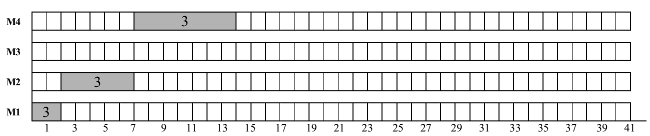

Figure1. Formation of schedule for given individual – Step 1

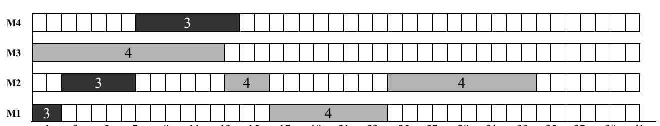

Figure 2. Formation of schedule for given individual – Step 2

When placing the operations on the chart, main problem is possible overlaps. In order to avoid overlaps, chart is searched for each operation and, operation can be placed only if needed space is present.

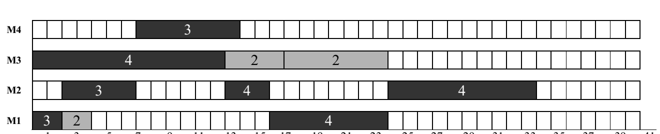

Figure 3. Formation of schedule for given individual – Step 3

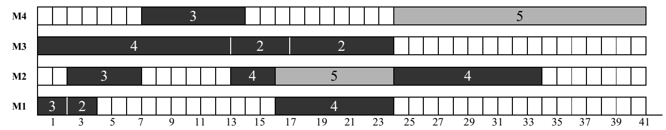

Figure 4. Formation of schedule for given individual – Step 4

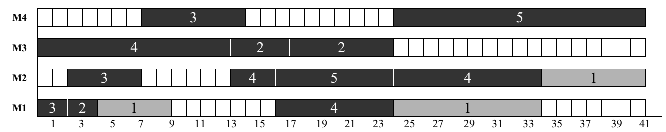

Figure 5. Formation of schedule for given individual – Step 5

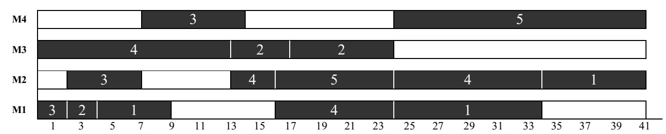

Figure 6. Final schedule for given individual

For the solution of the problem a multi-objective algorithm, modified version of Goldberg's Simple Genetic Algorithm, is used. In the given example string length is 30 for five orders and population size is chosen as 30. Crossover and mutation probabilities are 0.6 and 0.01 respectively, which are most common values [7].

It should be noted that the simulations were performed using a Intel Pentium Celeron based PC having ~567MHz speed and 64MB RAM. Additionally, the Virtual Pascal program is used to compile the necessary codes.

The simulation results of this example can be seen in Figures 7 and 8. The results indicate that the proposed algorithm produces appropriate schedules effectively.

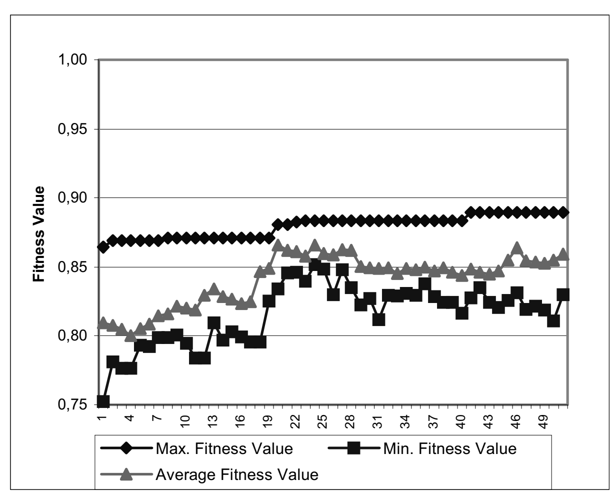

Figure 7. Simulation results

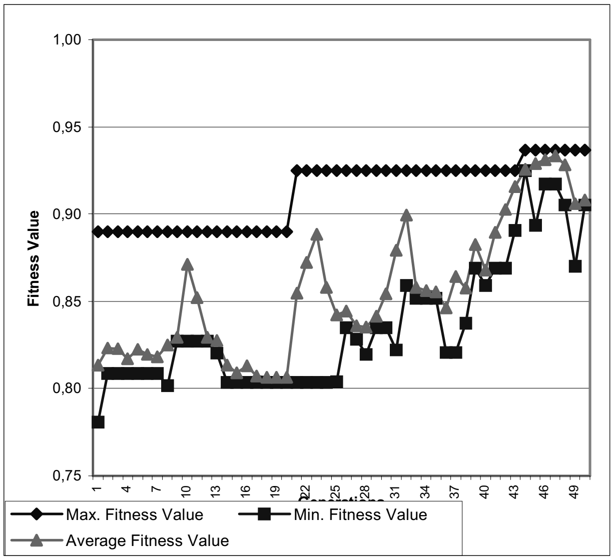

Figure 8. Simulation results

5. Conclusion

This work presents an algorithm to solve the operation sequencing problems with the following objective functions: makespan, number of tardy jobs and total tardiness. The main difference of this algorithm compared to the ones in the literature is that, the objective functions mentioned above are used at the same time. In the literature, two or more step methodologies are used to obtain active, non-delay, alternative schedules by the use of alternative machines. It is noted that these techniques increase the computation time. The use of the proposed algorithm produces active, non-delay schedules in one stage. One generic example is used to illustrate the results.

6. References

[1] S. V. B. Reddy, M. S. Shunmugam and T. T. Narendran, "Operation Sequencing in CAPP using Genetic Algorithms", *International Journal of Production Research,* vol.37 (5), 1999, pp. 1063-1074.

[2] J. Błaewicz, W. Domschke and E. Pesch, "The Job Shop Scheduling Problem: Conventional and New Solution Techniques", *European Journal of Operational Research*, vol.93, 1996, pp.1-33.

[3] W. Tan and B. Khoshnevis, "Integration of Process Planning and Scheduling - A Review", *Journal of Intelligent Manufacturing*, vol. 11, 2000, pp.51-63.

[4] N. Senin, R. Groppetti, and D. R. Wallace, "Concurrent Assembly Planning with Genetic Algorithms", *Robotics and Computer Integrated Manufacturing*, vol.16, 2000, pp.65-72.

[5] H. Zhou, Y. Feng, and L. Han, "The Hybrid Heuristic Genetic Algorithm for Job Shop Scheduling", *Computers and Industrial Engineering*, vol. 40, 2001, pp.191-200.

[6] J. K. Cochran, S. M. Horng and J. W. Fowler "A Multi-Population Genetic Algorithm to Solve Multi-Objective Scheduling Problems for Parallel Machines", *Computers & Operations Research*, vol. 30 (7), 2003, pp. 1087-1102.

[7] Goldberg, D.E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Pub., USA, 1989

[8] Baker, K. R., *Elements of Sequencing and Scheduling*, Amos Tuck School of Business Administration, Hanover; 1994.