Research Article

A Heuristic Scheduling Algorithm for Minimizing Makespan and Idle Time in a Nagare Cell

M. Muthukumaran and S. Muthu

Faculty of Mechanical Engineering, Anna University of Technology Coimbatore, Coimbatore 641 047, India

Correspondence should be addressed to M. Muthukumaran, profmmk@gmail.com

Adopting a focused factory is a powerful approach for today manufacturing enterprise. This paper introduces the basic manufacturing concept for a struggling manufacturer with limited conventional resources, providing an alternative solution to cell scheduling by implementing the technique of Nagare cell. Nagare cell is a Japanese concept with more objectives than cellular manufacturing system. It is a combination of manual and semiautomatic machine layout as cells, which gives maximum output flexibility for all kind of low-to-medium- and medium-to-high- volume productions. The solution adopted is to create a dedicated group of conventional machines, all but one of which are already available on the shop floor. This paper focuses on the development of heuristic scheduling algorithm in step-by-step method. The algorithm states that the summation of processing time of all products on each machine is calculated first and then the sum of processing time is sorted by the shortest processing time rule to get the assignment schedule. Based on the assignment schedule Nagare cell layout is arranged for processing the product. In addition, this algorithm provides steps to determine the product ready time, machine idle time, and product idle time. And also the Gantt chart, the experimental analysis, and the comparative results are illustrated with five (1×8 to 5×8) scheduling problems. Finally, the objective of minimizing makespan and idle time with greater customer satisfaction is studied through.

1. Introduction

In the present competitive business environment, many businesses focus their attention especially on the rapidity for responding to their customers' needs. For this reason, continuous improvements are needed to increase response times to customer changes. One of the strategies followed are cellular manufacturing (CM). It is a manufacturing philosophy that has attracted a lot of attention because of its positive impact on the flow shop production. The design of this system has been viewed as an important manufacturing strategy that has contributed to improve productivity. Similar parts are grouped into families and associated machines into groups so that one or more part families can be proc-essed within a cell. The process of determining the part families and machine groups is referred to as the cell formation (CF) problem [1]. CM has been considered as an alternative to conventional flow shop manufacturing where different parts are produced intermittently in small lot sizes. In a conventional flow shop group scheduling problem, each stage has only one machine.

Thus, when group of products complete their processing requirements on a machine representing a preceding stage they will need to wait their turn to get their processing requirements completed on the machine representing the next stage. In contrast, one or more stages in a flow shop scheduling problem can have two or more machines in that stage. Therefore, in a flow shop system products that belong to two different groups may be processed simultaneously, yet separately, on two different machines representing that stage, thus offering reduction in completion time [2]. Though, for increasing the number of machines in each stage it is difficult for some companies to invest more cost on machines. Moreover, the reduction in process time leads to empty running of machines and the product has to wait because of improper scheduling. So the problem of completing the products with the available resources by the company itself by changing the layout with new scheduling order is discussed in this paper. In detail the following assumptions should be made (1) Only a single-setup is performed on a machine for processing the products that belong to a group. (2) When

sequence-dependent setup is performed on a machine, the entire set of products that belong to the group are processed on the same machine and moved to the next stage. (3) The products that belong to the various groups are assumed to be available for processing on machines at the beginning of scheduling [3–5]. In this paper, a new Nagare cell flow shop scheduling problem is taken as a case study and discussed. The literature survey details, introduction of Nagare cell, and its characteristics are explained in Section 2. Problem definition and methodology used are explained in Section 3. Proposed work, algorithm, and experimental illustrations are explained in Sections 4 and 5. Discussions and conclusion are explained in Sections 6 and 7.

2. Literature Review

Like any other sector, nowadays manufacturing sector has been striving to survive against the economic downturn all over the world. Hence, firms look for more options for survival of the fittest. Undoubtedly, one of the most effective manufacturing strategies is reconfiguration of the production facilities [6]. Many types of production configurations have been proposed and provide revolution in the manufacturing industry. If demand structure is characterized by high volume and less variety, a product type layout is the most efficient. In the case of low-to-medium volume and unstable demand, a functional type layout is suggested. It is well known that while a product type layout limits flexibility, a functional type layout suffers from deficiency.

In the 1960s, cellular manufacturing system (CMS) based on group technology principles was introduced to cope with the drawbacks of product type layouts. In CMS, parts with similar processing requirements are grouped together to form product families and machines with different processing capabilities are grouped to create machine cells. The production operations are therefore simplified by processing a family through only one cell. Because of its processing advantages, CMS provides the basis for implementing more sophisticated manufacturing methods such as flexible manufacturing systems, computer-aided manufacturing, just-in-time production, and robotics [7–9]. The performance of CMS depends significantly on the stability of demand with respect to volume and mix. In the cell formation phase of CMS design it is assumed that the demand pattern is relatively stable over a long product life cycle. One method is allowing products to visit more than one cell to complete its processing requirements. But this increases the scheduling complexity and the primary principle of CMS is violated [10, 11]. Consider the cases where all multiple product types are produced in a single cell. All the products have different processing routes. To process these product types multiple machine types are required. Every machine type has more than one individual machine. It is assumed that all individual machines of any type have identical processing capabilities (i.e., processing times). The machines are located as layout in the shop floor. Since the problem is a flow shop scheduling problem, makespan minimization is a relevant objective [12]. On the other hand, integration of different machine raises the issue of minimizing the machine idle time. Since

product mix is permitted in the system scheduling, decisions must consider the assignment of machines to the product, the products start time at each machine, and the product quantity processed on different machines.

In 2000, Logendran and Sirikrai [13] stated that, while manufacturing cells are setup the products assigned to a cell would be processed completely on machines assigned with in the same cell. But in most industrial-size problems a nonideal situation prevails, in that one or more products assigned to a cell would require one or more operations to be performed on machines assigned to a cell other than its home cell. Such parts are called bottleneck (exceptional) parts and machines that cater to their needs are called bottleneck machines, thus creating disaggregated manufacturing cells. In 1989, Wemmerlov and Hyer [9] and, in 1998, Suresh and Kay [14] stated that in group scheduling problem the objective is to identify the sequence of parts that belong to each group as well as the sequence of groups themselves to optimize some measure of performance. Thus, unlike in conventional scheduling problems the issues in group scheduling must be dealt with on two levels. On level 1, the sequence of parts that belong to each group is determined, and, at level 2, the sequence of groups themselves is determined. The existing literature on scheduling groups of jobs with setup times on single or parallel machines was reviewed by Liaee and Emmons [15] in 1997. They consider scheduling with and without the group technology assumptions (GTAs) under a variety of performance measures. For minimizing the makespan in level 1 problem, heuristic methods based on optimal and heuristic decision rules for solving conventional flow shop scheduling problems have been proposed by Radharamanan, 1986 [16]; Al-Qattan, 1988 [17]; Logendran and Nudtasomboon [18], 1991. The single-machine group scheduling problem within the context of minimizing the total tardiness was investigated by Nakamura et al., 1978 [19], and Ozden et al. [20], in 1985. Baker [21] in 1988 reported a dynamic programming algorithm for identifying an optimal sequence of parts that minimizes the mean completion time. Webster and Baker [22] in 1995 review the research findings in group scheduling with regard to minimizing the total weighted flow time and the maximum lateness, although their review extends the results to cover the areas of group scheduling with batch availability and batch processing. For job scheduling, Sule [23] in 1982 and Proust et al. [24] in 1991 have considered the two-machine flow shop scheduling problems with (sequence independent) setup and removal times separated. Using the encouraging performance of the new heuristic algorithm by Logendran and Nudtasomboon [18] in 1991 for solving the level 1 problem, Logendran et al. [25] in 1995 carried out an investigation to determine how well the Logendran and Nudtasomboon (LN) algorithm performed in comparison with the Campbell et al. [26] (CDS) algorithm if each is combined with Petrov's (PT) heuristic to completely solve an m-machine group scheduling problem to minimize the makespan. Several heuristics as well as a branch and bound approach to solve the sequence-dependent flow shop group scheduling problem have been proposed by Schaller et al. [27] in 2000. In 2003 Reddy and Narendran [28], and in 1989 Sriskandarajah and Sethi [29] developed heuristics

with worst-case performance guarantees for minimizing the makespan in a two-stage flexible flow line. Ding and Kittichartphayak [30] in 1994 extended the heuristics for regular flow shop scheduling problems to develop three different heuristics for minimizing the makespan in a flexible flow line. Sawik [31] in 2000 proposed mixed integer programming formulations for minimizing the makespan in a flexible flow line with limited intermediate buffers to identify an exact solution using a commercially available software package such as CPLEX. These formulations were later extended for batch scheduling by Sawik in 2002. An experimental investigation was reported by Kurz and Askin in 2003 to compare sequencing rules for minimizing the makespan in flexible flow lines with nonanticipatory sequence-dependent setup time for jobs on machines.

In 2005, Reza Hejazi and Saghafian [12] gave a complete survey about flow shop scheduling problems and the contributions from early works of Johnson in 1954 [32] to recent approaches of metaheuristics in 2004. It provides models and frameworks for the performance parameters like flow shop, makespan, dynamic programming, heuristics, metaheuristics, simulated annealing, genetic algorithm, tabu search, and ant colony optimization. It mainly considers a flow shop problem with a makespan criterion, and it surveys some exact methods (for small-size problems), constructive heuristics too [33]. In 2011, Mahdavi et al. [34], in their survey had given about NP-hard problem and makespan review [34]. The foresaid paper behaves as a complete reference for past contributions to future research needs in improving and developing better approaches to flow shop-related scheduling problems. To the best of the literature knowledge the paper by Logendran et al. in 2005 is the only one that has attempted to minimize the makespan in group scheduling flexible flow shops with sequence-independent setups. This research focuses on the development and performance characteristics of a heuristic on NC scheduling for minimizing the makespan and idle time with sequence-dependent setups as the objective.

2.1. Nagare Cell. Nagare cell is a one-piece-processing-at-a-time machine group system where machines, men, and parts work together and men move from machine to machine by carrying work piece with them in a flow shop layout. NC manufacturing is a series of techniques for identifying and eliminating inventory by continuously improving the flow of product at the pull of the customers [4]. This system satisfies the customers by on-time delivery, quality, and price. The basic goal is, (i) to get more work done with fewer resources (ii) dramatic reduction of cycle times, (iii) faster customer responsiveness, (iv) increased employee productivity (v) higher equipment utilization, (vi) reduction in scrap and rework also. It minimizes the operators walking time and walking distances. It rectifies the additional space required to accumulate the in-process inventory, unnecessary handling of materials, slow feed back of quality information, longer production lead time, and so forth. In addition, NC is also an on-demand manufacturing approach used to achieve the excellence in manufacturing based on the continuous elimination of waste and consistent improvement in productivity

[4]. It provides the cost of effective production and delivery of only the necessary quality parts in the right quantity at the right time and at right place. The most sensible organizing cell layout preferable for NC is a U-shaped layout [35, 36]. It a is product-flow-based one that enables operators to produce single piece at a time and capable of handling multiple processes easily [5]. NC features describe how the operation has to be carried out, in which direction the product has to flow, How much cycle time is required for one product, how much is the installed capacity of the layout, what machine has to be selected, before adopting NC layout some prerequisites have to be checked, that is the layout is capable of manufacturing one or a family of components in the same flow path, the components design is suitable for manual handling, machines should have short cycle time, and operators can operate all machines within the cell; medium-to-high volume of production can be achievable with less expensive machines. The machines used for NC should have autocycle and be reliable.

2.1.1. Operating Characteristics. Since NC can be proposed as an alternative to the CM, it reflects the scheduling characteristics of the CM. The notable difference between these systems is how quick they respond to outer changes [37]. When demand pattern is prone to change, NC neither adapts nor responds to the changes adequately due to the physical grouping of machines. Instead of restraining the machines into the cells, machines with similar processing abilities are located in vicinity for effective utilization [35, 36, 38]. This feature enables the system to provide the product families. This paper dealt with the scheduling phase of NC problem which underscore the creation of cells, to increase the comprehension of NC concept and the total scheduling horizon is divided into time periods. Then it is assumed that when first operation of any product starts, next product must start in the consecutive period and also the operations can be followed continuously even if the preceding operations finished early. In this concept, all machines are arranged in the shortest route. The less the makespan is, the more the system is heavily utilized. At the same time the assignment schedule of any product batch is arranged as the shortest processing time (SPT) machine pattern.

3. Problem Definition

While developing a mathematical model for the NC scheduling problem all aspects are considered since the aim is to propose a model that incorporates all the operational issues. Every product type may visit different machine assignments and its operation sequences vary. Machine types consist of more than one identical machine. So for every operation of any product more than one alternative exists. With this form, the problem reminds us of the scheduling problem. But NC inherits and has its routes from CM, an important objective to be regarded. So this problem is more complicated than the flow shop problem, which is proved to be a nondeterministic problem [39, 40]. If we are investigating a problem that looks like flow shop problem and makespan is regarded, so the finding parameters are minimizing the makespan and

TABLE 1: Processing time of n products on m machines.

Product i (P$_i$)	Machine j (M_j)				
	$m1$	$m2$	$m3$	$\ldots$	mm
Product 1	$t11$	$t12$	$t13$	$\ldots$	$tm1$
Product 2	$t21$	$t22$	$t23$	$\ldots$	$tm2$
Product 3	$t31$	$t32$	$t33$	$\ldots$	$tm3$
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
Product n	$tn1$	$tn2$	$tn3$	$\ldots$	tnm

TABLE 2: Product A and its processing times at various machines.

Machine	$m1$	$m2$	$m3$	$m4$	$m5$	$m6$	$m7$	$m8$
Product A	1	2	4	3	1	5	2	2

idle time. The main problems faced by the manufacturing companies are that their existing flow shop layout is used to manufacture the parts for limited types of components, which causes a congested flow and bottlenecks. So the company wants a separate NC layout for the following activities: (a) to develop programs for each type of cell formation algorithms for computing the results, (b) to compare the various cell formation algorithm results for selecting the most suitable algorithm, (c) to propose a best NC layout from the obtained algorithm result, and (d) to compare the performance of the NC layout with the existing layout.

Once the cell formation is completed, a set of parts assigned to a cell will be processed completely on a set of machines assigned to the same cell. As the parts assigned to a cell belong to one family, the sequence of operations to be followed for each of these parts on machines will be the same. If there are n parts and m machines in a flow shop, the total number of sequences that can be generated for this problem is $n!$. For example if there are 5 products and 8 machines assigned to a cell, the number of sequences generated is $5!$. The operational sequence on all the five of them could possibly be $m1$, $m2$, $m3$, $m4 \ldots mj$ where mj denotes machine j; however, it is not uncommon to have some parts skiping one or more operations on machines in a group scheduling problem. For instance, part 1 might require processing on all four machines, whereas part 5 might require processing only on machines $m1$, $m3$, $m4$. This is in contrast to a conventional flow shop problem where each part, although following the same operational sequence, will require processing on all machines.

Both optimal and heuristic scheduling algorithms have been developed for a flow shop scheduling problem by researchers in the past. Some of the optimizing algorithms are documented by the authors [26, 32, 41–43]. As the computation time and memory requirements are very high for solving even a small problem using heuristic scheduling algorithms that have received a much favorable publicity in the literature than optimizing algorithms, algorithms incorporating some of the basic ideas for solving conventional flow shop scheduling problems have been developed for solving group scheduling problems. Algorithms for minimizing total tardiness of a group scheduling problem have been

documented by Nakamura et al. [19] and Cho et al. [44]. Heuristic algorithms documented by Radharamanan and Al-Qattan have focused on minimizing the makespan of a multistage group scheduling problem [16, 17]. This paper addresses a new heuristic algorithm proposed for minimizing the makespan and idle time for an NC scheduling problem. An extensive analysis has been performed by solving a variety of test problems. For example, the processing time of n products on m machines general problem is given in Table 1.

3.1. Methodology. Formation of algorithm and machine cells is the first step towards the design of NC scheduling problem. The primary input data of customer demand per day, number of products, number of machines, and its processing time are obtained as input data and finalized. Then, the processing times are entered in rows that represent the product and the columns represent the machines. An element a_{ij} of the matrix is one if the ith product visits the jth machine for processing; otherwise it is zero. The proposed algorithm gives steps to form the cell to rearrange the machines in SPT rule by which all the machines are arranged in ascending order based on the sum of the processing times of each product with respect to machines. The methodology followed in this paper is the prepared data of product verses machine matrix table (Tables 2, 5, 7, 9, and 11) which are important as input for scheduling. Next the table values are subjected to SPT rule until the objective is reached. Gantt chart is built to study the flow of products and highlight the idle time. Then, the most near optimal schedule is selected to build the NC layout. Then, the existing conventional layout is compared with the NC layout and evaluated in terms of makespan and idle time of machine and product.

4. Proposed Work

There are some general descriptions to be followed for practicing NC scheduling: let there are n products and m machines, each product consists of m operations and each operation requires a different machine, all the n products have to be processed in the same sequence on m machines, the processing time of product i on machine j is represented by T_{ij}, that is, $(i = 1, \ldots, n; j = 1, \ldots, m)$, the makespan

TABLE 3: Product A makespan and idle time determination.

| | | Processing time | | | | | | | | | | | | | | | | | Idle time | |
| | | m1 | | m2 | | m3 | | m4 | | m5 | | m6 | | m7 | | m8 | | | | |
Prod	C_{dpd}	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_R	$t_{idm/c}$	t_{idp}
A1	1000	0	1	1	3	3	7	7	10	10	11	11	16	16	18	18	20	20		
A2		1	2	3	5	7	11	11	14	14	15	16	21	21	23	23	25	5	10	5
A3		2	3	5	7	11	15	15	18	18	19	21	26	26	28	28	30	5	10	10
A4		3	4	7	9	15	19	19	22	22	23	26	31	31	33	33	35	5	10	15
A5		4	5	9	11	19	23	23	26	26	27	31	36	36	38	38	40	5	10	20

TABLE 4: Product A optimal makespan and minimum idle time determination.

| | | Processing time | | | | | | | | | | | | | | | | | Idle time | |
| | | m1 | | m5 | | m2 | | m7 | | m8 | | m4 | | m3 | | m6 | | | | |
Prod	C_{dpd}	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_R	$t_{idm/c}$	t_{idp}
A1	1000	0	1	1	2	2	4	4	6	6	8	8	11	11	15	15	20	20		
A2		1	2	2	3	4	6	6	8	8	10	11	14	15	19	20	25	5	0	5
A3		2	3	3	4	6	8	8	10	10	12	14	17	19	23	25	30	5	0	10
A4		3	4	4	5	8	10	10	12	12	14	17	20	23	27	30	35	5	0	15
A5		4	5	5	6	10	12	12	14	14	16	20	23	27	31	35	40	5	0	20
A1000																		5015		

TABLE 5: Products A and B with their processing times.

Machine	m1	m2	m3	m4	m5	m6	m7	m8
Product A	1	2	4	3	1	5	2	2
Product B	1	2	0	3	1	0	2	2

TABLE 6: Products A and B with optimal makespan and minimum idle time determination.

| | | Processing time | | | | | | | | | | | | | | | | | Idle time | |
| | | m1 | | m5 | | m3 | | m7 | | m8 | | m2 | | m6 | | m4 | | | | |
Prod	C_{dpd}	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_R	$t_{idm/c}$	t_{idp}
A1	500	0	1	1	2	2	6	6	8	8	10	10	12	12	17	17	20	20		
B1	500	1	2	2	3	6	6	8	10	10	12	12	14	17	17	20	23	3	0	12
A2		2	3	3	4	6	10	10	12	12	14	14	16	17	22	23	26	3	0	6
B2		3	4	4	5	10	10	12	14	14	16	16	18	22	22	26	29	3	0	18
A3		4	5	5	6	10	14	14	16	16	18	18	20	22	27	29	32	3	0	12
B3		5	6	6	7	14	14	16	18	18	20	20	22	27	27	32	35	3	0	24
A4		6	7	7	8	14	18	18	20	20	22	22	24	27	32	35	38	3	0	18
B4		7	8	8	9	18	18	20	22	22	24	24	26	32	32	38	41	3	0	30
A5		8	9	9	10	18	22	22	24	24	26	26	28	32	37	41	44	3	0	24
B5		9	10	10	11	22	22	24	26	26	28	28	30	37	37	44	47	3	0	36
B_n																		3017		

TABLE 7: Products A, B, and C with their processing times.

Machine	m1	m2	m3	m4	m5	m6	m7	m8
Product A	1	2	4	3	1	5	2	2
Product B	1	2	0	3	1	0	2	2
Product C	2	0	2	4	0	3	1	2

TABLE 8: Products A, B, and C with optimal makespan and minimum idle time determination.

| | | | Processing time | | | | | | | | | | | | | | | | Idle time | | |
| | | $m5$ | | $m1$ | | $m2$ | | $m7$ | | $m3$ | | $m8$ | | $m6$ | | $m4$ | | | | |
Prod	C_{dpd}	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_R	$t_{idm/c}$	t_{idp}
A1	333	0	1	1	2	2	4	4	6	6	10	10	12	12	17	17	20	20		
B1	333	1	2	2	3	4	6	6	8	10	10	12	14	17	17	20	23	3	0	12
C1	333	2	2	3	5	6	6	8	9	10	12	14	16	17	20	23	27	4	0	13
A2		2	3	5	6	6	8	9	11	12	16	16	18	20	25	27	30	3	0	10
B2		3	4	6	7	8	10	11	13	16	16	18	20	25	25	30	33	3	0	22
C2		4	4	7	9	10	10	13	14	16	18	20	22	25	28	33	37	4	0	23
A3		4	5	9	10	10	12	14	16	18	22	22	24	28	33	37	40	3	0	20
B3		5	6	10	11	12	14	16	18	22	22	24	26	33	33	40	43	3	0	32
C3		6	6	11	13	14	14	18	19	22	24	26	28	33	36	43	47	4	0	33
A4		6	7	13	14	14	16	19	21	24	28	28	30	36	41	47	50	3	0	30
B4		7	8	14	15	16	18	21	23	28	28	30	32	41	41	50	53	3	0	42
C4		8	8	15	17	18	18	23	24	28	30	32	34	41	44	53	57	4	0	43
C_n																	3350			

TABLE 9: Products A, B, C, and D with their processing times.

Machine	$m1$	$m2$	$m3$	$m4$	$m5$	$m6$	$m7$	$m8$
Product A	1	2	4	3	1	5	2	2
Product B	1	2	0	3	1	0	2	2
Product C	2	0	2	4	0	3	1	2
Product D	1	4	3	2	2	4	2	4

TABLE 10: Products A, B, C, and D with optimal makespan and minimum idle time determination.

| | | | Processing time | | | | | | | | | | | | | | | | Idle time | | |
| | | $m5$ | | $m1$ | | $m7$ | | $m2$ | | $m3$ | | $m8$ | | $m6$ | | $m4$ | | | | |
Prod	C_{dpd}	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_R	$t_{idm/c}$	t_{idp}
A1	250	0	1	1	2	2	4	4	6	6	10	10	12	12	17	17	20	20		
B1	250	1	2	2	3	4	6	6	8	10	10	12	14	17	17	20	23	3	0	12
C1	250	2	2	3	5	6	7	8	8	10	12	14	16	17	20	23	27	4	0	13
D1	250	2	4	5	6	7	9	9	13	13	16	16	20	20	24	27	29	2	2	7
A2		4	5	6	7	9	11	13	15	16	20	20	22	24	29	29	32	3	0	12
B2		5	6	7	8	11	13	15	17	20	20	22	24	29	29	32	35	3	0	24
C2		6	6	8	10	13	14	17	17	20	22	24	26	29	32	35	39	4	0	25
D2		6	8	10	11	14	16	17	21	22	25	26	30	32	36	39	41	2	0	19
A3		8	9	11	12	16	18	21	23	25	29	30	32	36	41	41	44	3	0	24
B3		9	10	12	13	18	20	23	25	29	29	32	34	41	41	44	47	3	0	36
C3		10	10	13	15	20	21	25	25	29	31	34	36	41	44	47	51	4	0	37
D3		10	12	15	16	21	23	25	29	31	34	36	40	44	48	51	53	2	0	31
D_n																	3017			

TABLE 11: Products A, B, C, D, and E with their processing times.

Machine	$m1$	$m2$	$m3$	$m4$	$m5$	$m6$	$m7$	$m8$
Product A	1	2	4	3	1	5	2	2
Product B	1	2	0	3	1	0	2	2
Product C	2	0	2	4	0	3	1	2
Product D	1	4	3	2	2	4	2	4
Product E	0	3	2	1	0	2	4	3

indicates the sequence of products minimizing the maximum flow time, the customer demand per day is C_{dpd} and takt time is ratio of total cycle time (t_{ct}) to time available per day (t_a) (i.e., tact = t_{ct}/t_a), the idle time of machine and product is t_{idm} and t_{idp}, and the product ready time is T_R. In addition to the descriptions some assumptions are to be followed while design scheduling: every product has to be processed on all machines in the machine assignment order ($j = 1, 2, \ldots, m$), every machine processes only one product at a time, operations are not preemptive, setup times for the operations are sequence independent and are included in the processing times, the operating sequences of the products are the same on every machine, the common sequence has to be determined, some spare machines should be maintained, and each operator should have skill to operate all machines.

4.1. Algorithm. In essence, the algorithm provides solution for assigning operations of each product to machines and sequencing operations on each machine. The significant side of the proposed heuristic is the collocation and reformation of the structure of the model under study, which is the key to adapt it [45]. The following is a step-by-step explanation of the algorithm.

Step 1. To start the procedure get the customer demand per day as input data, that is, C_{dpd}, the number of products, that is, "i" (1, 2, 3,..., n), and the number of machines, that is, "j" (1, 2, 3, ..., m).

Step 2. Form the flow shop for the range of product, that is, I = A, AB, ABC, and so forth by each machine of j = 1 to m.

Step 3. For each machine of j, compute the total processing time T_j, given by

$$Tj = \sum t_{ij}, \quad \text{overall } i = 1, \ldots, n, \ t_{ij} > 0, \quad (1)$$

where t_{ij} is the processing time of product i on machine j on which $t_{ij} \geq 0$.

Step 4. For each machine of j, the total processing time T_j is calculated:

$$T_{j=1} = \{t11 + t21 + t31 + \cdots + tn1\},$$

$$T_{j=2} = \{t12 + t22 + t32 + \cdots + tn2\}, \text{ like up to}$$

$$T_{j=m} = \{t1m + t2m + t3m + \cdots + tnm\} \text{ for all products } i. \quad (2)$$

Step 5. Sort the machines in an ascending order of T_j's; if two or more machines are having the same value of T_j, sort them in an arbitrary order.

Step 6. Rank the machines in the sorted order of T_j's, and this will be scheduled for the product flow.

Step 7. Assign the product range for processing to find out input time T_i and output time T_o of each machine.

Step 8. As per the assignment order, the first product of A will be completed in k minutes, which is denoted as product ready time T_R, then the second one has to refer to the T_o of next machine. If the machine is free, it will proceed the processing and complete in $k + r$ minutes. Likewise the cycle will repeat.

Step 9. Here if the machine is waiting for the product, this is accounted as the machine idle time $t_{idm/c}$, and if the product is waiting this is accounted as the product idle time t_{idp}. Similarly the products will be processed as A1, A2, A3... An for product A, A1, B1, A2, B2….An, Bn for product AB, and so forth up to n number of products to meet C_{dpd}. If product range is more than two, permutations will be followed and the minimal value is selected.

Step 10. The summation values of T_R are accounted as the makespan.

Step 11. Enumerate Step 2 to Step 10 towards the end of schedule as minimum makespan and zero idle time of machine and product will reach.

Step 12. If $i = n$, stop and pick up the minimum makespan value and assignment sequence.

5. Experimental Results

To better comprehend the model some illustrative examples are given in this section. There are 5 types of products A, AB, ABC, ABCD, and ABCDE of an NC problem to be processed in 8 machines $m1, m2, \ldots, m8$ as shown in Tables 2, 5, 7, 9 and 11. In Table 2 machine numbers are seen in the first row and the product processing times are given in the second row. In Table 3 the processing schedule of product A up to five products is explained. The details of C_{dpd} are shown in the second column, and further machines are arranged in normal order that is, $m1$, $m2$, $m3$, $m4$, $m5$, $m6$, $m7$, and $m8$ for processing. First product of A starts processing in machine "$m1$" for 1 minute, then $m2$ for 2 minutes, then $m3$ for 4 minutes, then $m4$ for 3 minutes, then $m5$ for 1 minute, then $m6$ for 5 minutes, then m7 for 2 minutes, and then $m8$ for 2 minutes. It is noted that the first product of A will get ready by 20 minutes and the second product of A will get ready in another 5 minutes; similarly, for every 5 minutes, the next product of A will be ready. And 10 minutes of machine idle time and 5 minutes of product idle time are obtained for every product of A. The foresaid results are obtained without the application of the SPT rule. But by applying the SPT rule the machine assignment order has changed to $m1$, $m5$, $m2$, $m7$, $m8$, $m4$, $m3$, and $m6$; through this order the first product of A will be ready by 20 minutes, and the second product of A will be ready in another 5 minutes; similarly, for every 5 minutes, the next product of A will be ready. Here a notable change is observed, that is, zero idle time for machine and 5 minutes of idle time for each product of A are shown in Table 4. Hence, by the near optimal solution of 5015 minutes of makespan and zero idle time of machine, the paper objective is achieved.
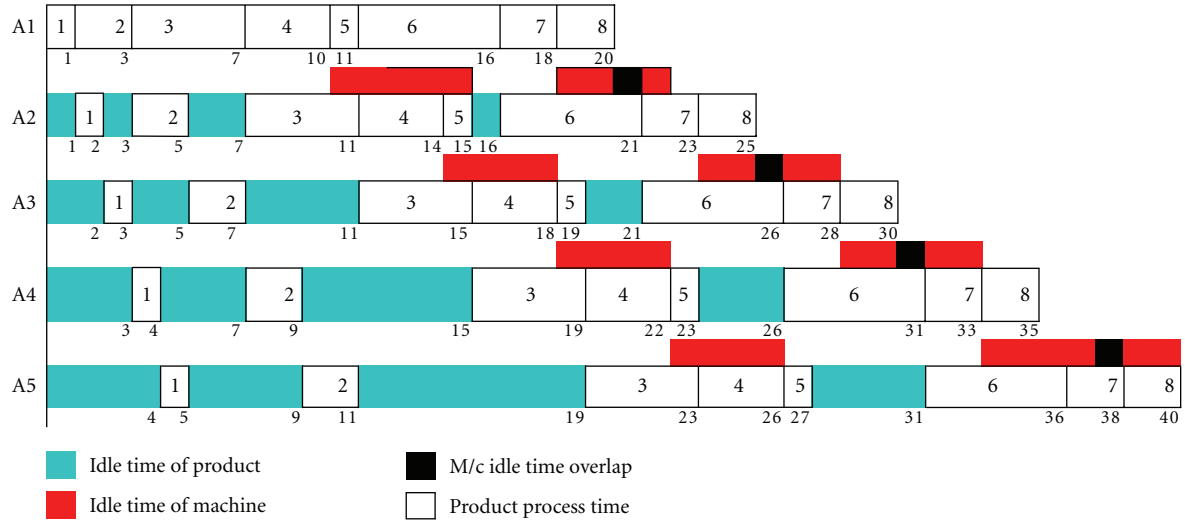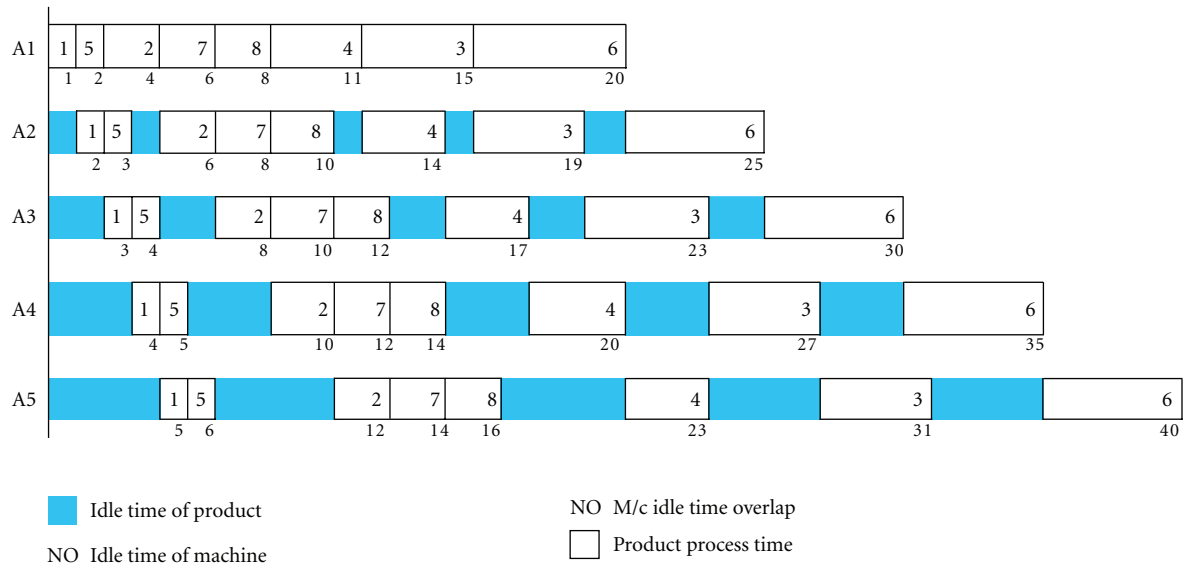
FIGURE 1: Gantt chart of Product A without SPT rule.

FIGURE 2: Gantt chart of Product A with SPT rule.

TABLE 12: Products A, B, C, D, and E with optimal makespan and minimum Idle time determination.

		Processing time																Idle time		
	Machine	$m5$		$m1$		$m7$		$m3$		$m2$		$m8$		$m4$		$m6$				
Prod	C_{dpd}	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_i	T_o	T_R	$t_{\text{idm/c}}$	t_{idp}
A1	200	0	1	1	2	2	4	4	8	8	10	10	12	12	15	15	20	20		
B1	200	1	2	2	3	4	6	8	8	10	12	12	14	15	18	20	20	0	0	9
C1	200	2	2	3	5	6	7	8	10	12	12	14	16	18	22	22	25	5	2	11
D1	200	2	4	5	6	7	9	10	13	13	17	17	21	22	24	25	29	4	2	7
E1	200	4	4	6	6	9	13	13	15	17	20	21	24	24	25	29	31	2	0	16
A2		4	5	6	7	13	15	15	19	20	22	24	26	26	29	31	36	5	1	16
B2		5	6	7	8	15	17	19	19	22	24	26	28	29	32	36	36	0	0	25
C2		6	6	8	10	17	18	19	21	24	24	28	30	32	36	36	39	3	0	25
D2		6	8	10	11	18	20	21	24	24	28	30	34	36	38	39	43	4	0	21
E2		8	8	11	11	20	24	24	26	28	31	34	37	38	39	43	45	2	0	30
E_n																	2817			
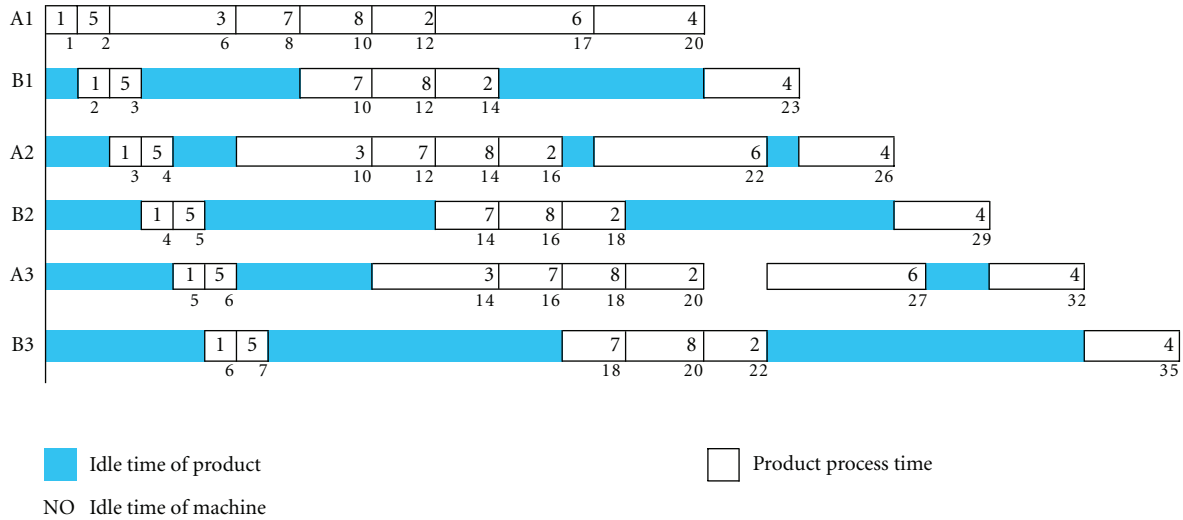
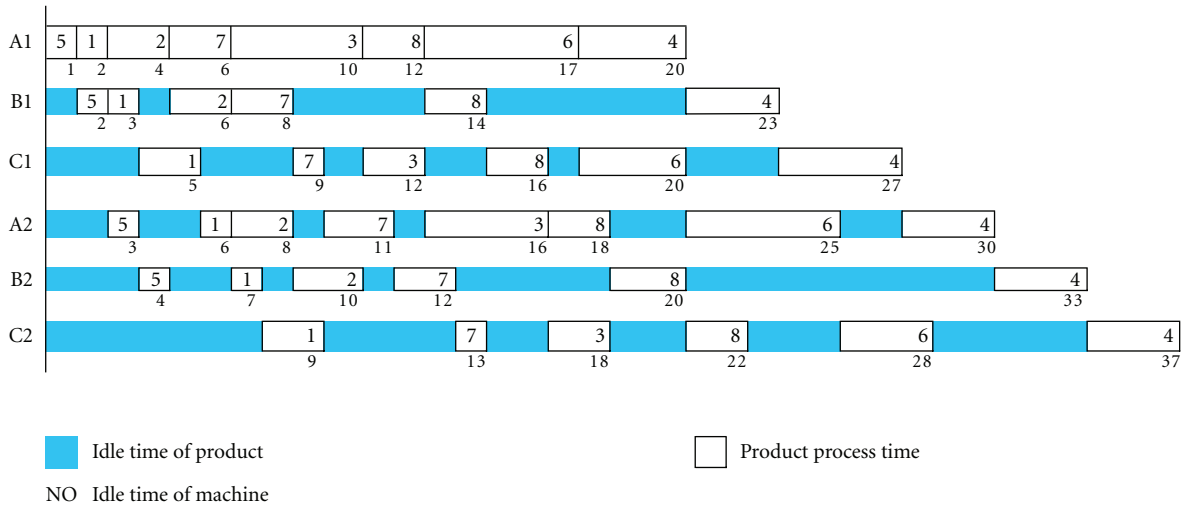FIGURE 3: Gantt chart of Products A and B with SPT rule.

FIGURE 4: Gantt chart of Products A, B and C with SPT rule.

In addition to processing tables, Gantt charts are also prepared to clearly explain about the processing time and idle time. In Figure 1, the sky blue colour represents the product idle time, the red colour represents the machine idle time, and the black colour represents the overlapping of machine idle time for every product. But in Figure 2, the sky blue represents the minimum product idle time and the red and black colours are not noticed. It justifies that machines are effectively used without any idle running. The same procedure is repeated for the so forth type of the problems. The Tables 5, 7, 9 and 11 gives the details of the processing time of products AB, ABC so forth up to ABCDE. Then the Tables 6, 8, 10 and 12 gives the details of optimal makespan, minimum idle time of products AB, ABC so forth up to ABCDE. Also the Figures 3, 4, 5, and 6 explain the Gantt chart for products AB, ABC so forth up to ABCDE with SPT rule. The experimental results are completely shown in Figures 3–6 and Tables 5–12.

6. Discussions

The results of the above illustrations are evaluated, and their relationships are shown in Tables 13 and 14 and Figures 7, 8, and 9. From Figure 7 the chart shows the relationship between makespan values to the total number of parts. On observing the path of curves it is shown that while increasing the product variety the makespan values are decreased drastically, that is, the makespan value of single product type A is 5015 minutes for 1000 products. While increasing the product type from A to A, B then A, B, C then A, B, C, D, and then A, B, C, D, E, the makespan values are decreased, that is, 3017 minutes for A, B then 3350 minutes for A, B, C then 3017 minutes for A, B, C, D, E, and lastly 2817 minutes for A, B, C, D, E as shown in Table 13. The bar chart as shown in Figures 8 and 9 gives the comparison of the idle time of machine and the idle time of product before evaluation and after evaluation. So it is clearly understood
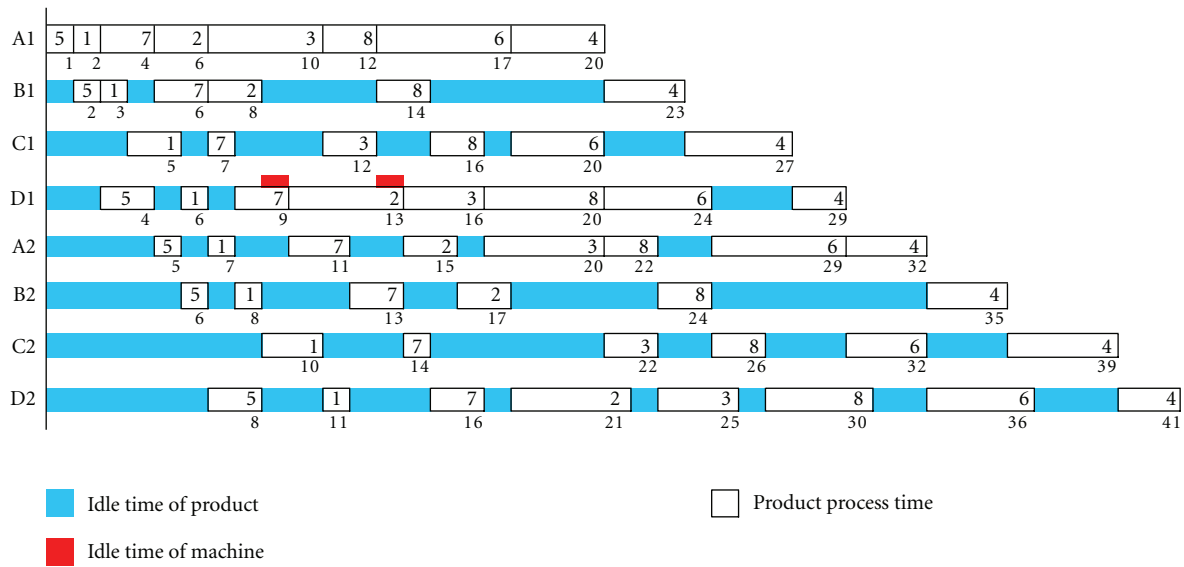
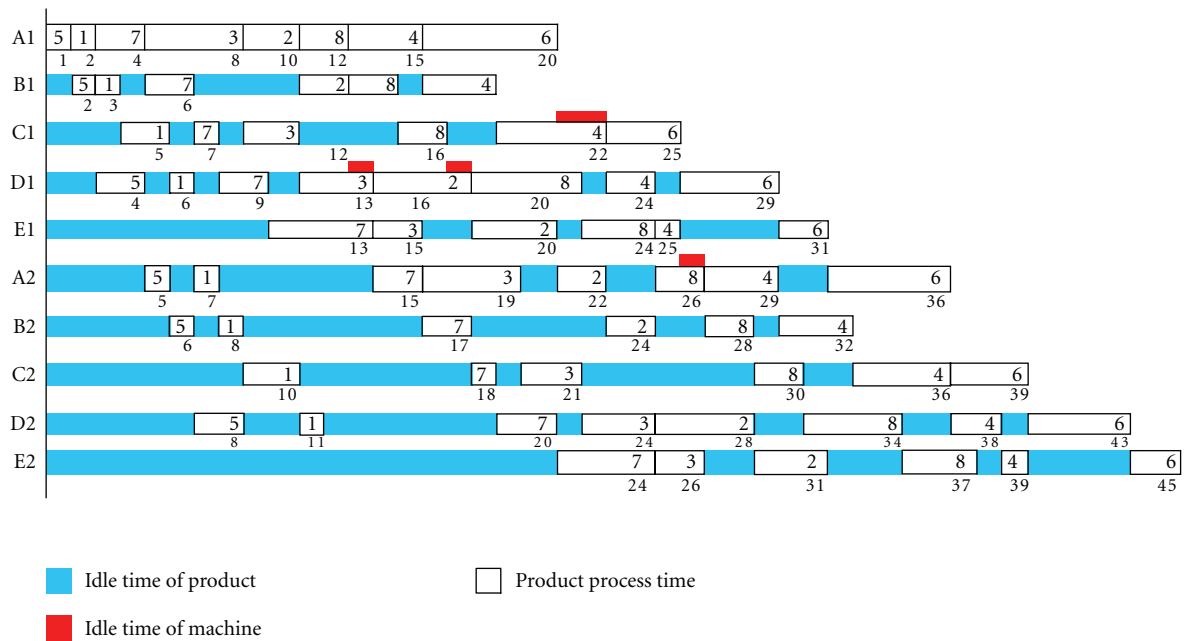FIGURE 5: Gantt chart of Products A, B, C, and D with SPT rule.

FIGURE 6: Gantt chart of Products A, B, C, D, and E with SPT rule.

TABLE 13: Results of makespan, idle time, and total products produced.

Sl. no.	Product	Customer demand per day	Machine sequence	Makespan	Idle time	
					Machine	Product
1	A	1000	1-5-2-7-8-4-3-6	5015	0	5
2	AB	1000	1-5-3-7-8-2-6-4	3017	0	12
3	ABC	1000	5-1-2-7-3-8-6-4	3350	0	25
4	ABCD	1000	5-1-7-2-3-8-6-4	3017	2	32
5	ABCDE	1000	5-1-7-3-2-8-4-6	2817	5	43

TABLE 14: Results of output/cell, Takt time, No of operators required.

Sl. no.	Product	Customer demand per day	Machine sequence	Product cycle time	Output/ cell/shift	Takt time	No of operators /cell	Total parts produced
1	A	1000	1-5-2-7-8-4-3-6	300.9	89.73	81	4	1000
2	AB	1000	1-5-3-7-8-2-6-4	181.02	149.15	81	2	1000
3	ABC	1000	5-1-2-7-3-8-6-4	201.00	134.32	81	2	1000
4	ABCD	1000	5-1-7-2-3-8-6-4	181.02	149.15	81	2	1000
5	ABCDE	1000	5-1-7-3-2-8-4-6	169.02	159.74	81	2	1000
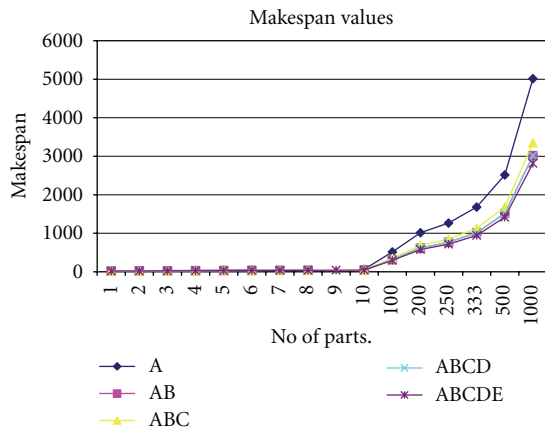
FIGURE 7: Makespan values of five different experiments.
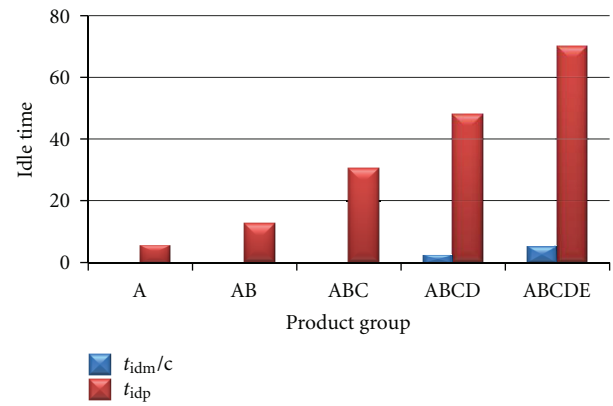
FIGURE 9: Results of idle time after evaluation.
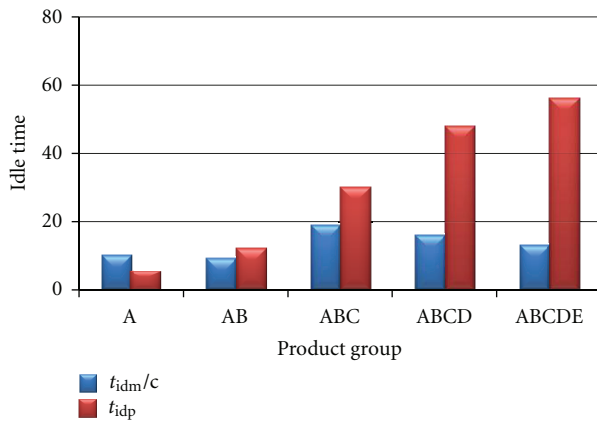
FIGURE 8: Results of idle time before evaluation.

that after the SPT rule evaluation the idle time is reduced to near zero value. The results of arithmetic calculations obtained during iteration are summarized in Table 14. It provides details of the machine sequence, product cycle time, output per cell per shift, takt time required, and the number of operators required to finish the work. From the foresaid discussion it is noted that this paper clearly discussed and provides solution for NC scheduling problems with (i) higher difference in makespan minimization and machine idle time and (ii) smaller difference in product idle time. Hence, the NC layout model is built based on the selected

scheduling. Then, the existing conventional manufacturing system layout is compared for performance with the NC manufacturing layout in terms of productivity and customer satisfaction.

7. Conclusion and Future Research

In this paper, the solution adopted creates a dedicated group of conventional machines and provides flexible output with one and more operators working simultaneously. The developed heuristic step-by-step scheduling algorithm suggested the shortest assignment schedule. Based on the assignment schedule the NC layout has arranged for processing the product and representing the potential issues. Hence, it is concluded that the NC manufacturing system has the ability against the demand changes, thereby providing effective utilization of machine and workers. Since the NC manufacturing looks like flow shop system, it can be developed to some extent. Following the characteristics of the model a lot of researches are going, a future research can also be carried out with scheduling software, and other scheduling criteria may be added to the problem in parallel. Since the problem discussed is among the NP hard problems, the same can be subjected for big-size problems, that is, more than 10×15, 15×20, 20×25 jobs and machines, and so forth. An optimizing tool can be applied for finding the optimal solutions. Metaheuristic techniques may be applied for getting solutions.

References

[1] M. Murugan and V. Selladurai, "Optimization and implementation of cellular manufacturing system in a pump industry using three cell formation algorithms," *International Journal of Advanced Manufacturing Technology*, vol. 35, no. 1-2, pp. 135–149, 2007.

[2] S. E. Kesen, S. K. Das, and Z. Gungor, "A mixed integer programming formulation for scheduling of virtual manufacturing cells (VMCs)," *The International Journal of Advanced Manufacturing Technology*, vol. 47, no. 5–8, pp. 665–678, 2009.

[3] M. Muthukumaran and S. Muthu, "A heuristic approach of Nagare Cell manufacturing in flow product system," in *Proceedings of the National Conference (RTDMS '10)*, pp. 19–20, Sri Ramakrishna Engineering College, Coimbatore, Tamil Nadu, India, 2010.

[4] D. G. M. Brooks, "The competitive edge in developing a traditional manufacturing business," *Technovation*, vol. 14, no. 6, pp. 357–362, 1994.

[5] D. G. M. Brooks and R. Duffill, "Towards world class at Leslie & Co," in *Proceedings of the 8th International Conference on Manufacturing Research*, University of Central England, London, UK, 1992.

[6] S. Benjaafar, S. S. Heragu, and S. A. Irani, "Next generation factory layouts: research challenges and recent progress," *Interfaces*, vol. 32, no. 6, pp. 58–77, 2002.

[7] A. Kusiak, *Intelligent Manufacturing Systems*, Prentice Hall, Englewood Cliffs, NJ, USA, 1990.

[8] U. Wemmerlov and N. L. Hyer, "Research issues in cellular manufacturing," *International Journal of Production Research*, vol. 25, no. 3, pp. 413–431, 1987.

[9] U. Wemmerlov and N. L. Hyer, "Cellular manufacturing in the US industry: a survey of users," *International Journal of Production Research*, vol. 27, no. 9, pp. 1511–1530, 1989.

[10] R. M. Gupta and J. A. Tompkins, "An examination of the dynamic behavior of part families in group technology," *International Journal of Production Research*, vol. 20, no. 1, pp. 73–86, 1982.

[11] J. B. Jensen, M. K. Malhotra, and P. R. Philipoom, "Machine dedication and process flexibility in a group technology environment," *Journal of Operations Management*, vol. 14, no. 1, pp. 19–39, 1996.

[12] S. Reza Hejazi and S. Saghafian, "Flowshop-scheduling problems with makespan criterion: a review," *International Journal of Production Research*, vol. 43, no. 14, pp. 2895–2929, 2005.

[13] R. Logendran and V. Sirikrai, "Machine duplication and part subcontracting in the presence of alternative cell locations in manufacturing cell design," *Journal of the Operational Research Society*, vol. 51, no. 5, pp. 609–624, 2000.

[14] N. C. Suresh and J. M. Kay, *Group Technology and Cellular Manufacturing: A State-of-the-Art Synthesis of Research and Practice*, Springer, 1998, Technology & Engineering.

[15] M. M. Liaee and H. Emmons, "Scheduling families of jobs with setup times," *International Journal of Production Economics*, vol. 51, no. 3, pp. 165–176, 1997.

[16] R. Radharamanan, "A heuristic algorithm for group scheduling," in *Proceedings of the International Industrial Engineering Conference*, pp. 229–236, 1986.

[17] I. AI-Qattan, "Designing GT cells enhanced by group scheduling: key to flexible manufacturing system," in *Proceedings of the IIE Integrated Systems Conference*, pp. 25–30, 1988.

[18] R. Logendran and N. Nudtasomboon, "Minimizing the makespan of a group scheduling problem: a new heuristic," *International Journal of Production Economics*, vol. 22, no. 3, pp. 217–230, 1991.

[19] N. Nakamura, T. Yoshida, and K. Hitomi, "Group production scheduling for minimum total tardiness: part I," *AIIE Transactions*, vol. 10, no. 2, pp. 157–162, 1978.

[20] M. Ozden, P. J. Egbelu, and A. V. Iyer, "Job scheduling in a group technology environment for a single facility," *Computers and Industrial Engineering*, vol. 9, no. 1, pp. 67–72, 1985.

[21] K. R. Baker, "Scheduling the production of components at a common facility," *IIE Transactions*, vol. 20, pp. 32–35, 1988.

[22] S. Webster and K. R. Baker, "Scheduling groups of jobs on a single machine," *Operations Research*, vol. 43, no. 4, pp. 692–703, 1995.

[23] D. R. Sule, "Sequencing *n* jobs on two machines with setup, processing and removal times separated," *Naval Research Logistics Quarterly*, vol. 29, no. 3, pp. 517–519, 1982.

[24] C. Proust, J. N. D. Gupta, and V. Deschamps, "Flowshop scheduling with set-up, processing and removal times separated," *International Journal of Production Research*, vol. 29, no. 3, pp. 479–493, 1991.

[25] R. Logendran, L. Mai, and D. Talkington, "Combined heuristics for bi-level group scheduling problems," *International Journal of Production Economics*, vol. 38, no. 2-3, pp. 133–145, 1995.

[26] H. G. Campbell, R. A. Dudek, and M. L. Smith, "A heuristic algorithm for the n job, m machine sequencing problem," *Management Science*, vol. 16, no. 10, pp. B630–B637, 1970.

[27] J. E. Schaller, J. N. D. Gupta, and A. J. Vakharia, "Scheduling a flowline manufacturing cell with sequence dependent family setup times," *European Journal of Operational Research*, vol. 125, no. 2, pp. 324–339, 2000.

[28] V. Reddy and T. T. Narendran, "Heuristics for scheduling sequence-dependent set-up jobs in flow line cells," *International Journal of Production Research*, vol. 41, no. 1, pp. 193–206, 2003.

[29] C. Sriskandarajah and S. P. Sethi, "Scheduling algorithms for flexible flowshops: worst and average case performance," *European Journal of Operational Research*, vol. 43, no. 2, pp. 143–160, 1989.

[30] F. Y. Ding and D. Kittichartphayak, "Heuristics for scheduling flexible flow lines," *Computers and Industrial Engineering*, vol. 26, no. 1, pp. 27–34, 1994.

[31] T. Sawik, "Mixed integer programming for scheduling flexible flow lines with limited intermediate buffers," *Mathematical and Computer Modelling*, vol. 31, no. 13, pp. 39–52, 2000.

[32] S. M. Johnson, "Optimal two-and three-stage production schedules with setuptimes included," *Naval Research Logistics Q*, vol. 1, no. 1, pp. 61–68, 1954.

[33] S. R. Hejazi and S. Saghafian, "Flowshop-scheduling problems with makespan criterion: a review," *International Journal of Production Research*, vol. 43, no. 14, pp. 2895–2929, 2005.

[34] I. Mahdavi, V. Zarezadeh, and P. Shahnazari-Shahrezaei, "Flexible flow shop scheduling with equal number of unrelated parallel machines," *International Journal of Industrial Engineering*, vol. 7, no. 13, pp. 74–83, 2011.

[35] K. Ohno and K. Nakade, "Analysis and optimization of a U-shaped production line," *Journal of the Operations Research Society of Japan*, vol. 40, no. 1, pp. 103–104, 1997.

[36] S. Henn, S. Koch, Gerking H. et al., "A U-shaped layout for a manual order picking," Tech. Rep. o4, Faculty of Economics and Management, 2011.

[37] M. S. Akturk and A. Turkcan, "A cellular manufacturing system design using a holonistic approach," *International Journal of Production Research*, vol. 38, no. 10, pp. 2327–2347, 2000.

[38] http://www.ateneonline.it/chase2e/studenti/tn/6184-7_tn05.pdf.

[39] E. M. Wicks and R. J. Reasor, "Designing cellular manufacturing systems with dynamic part populations," *IIE Transactions*, vol. 31, no. 1, pp. 11–20, 1999.

[40] K. L. Mak, P. Peng, X. X. Wang, and T. L. Lau, "An ant colony optimization algorithm for scheduling virtual cellular manufacturing systems," *International Journal of Computer Integrated Manufacturing*, vol. 20, no. 6, pp. 524–537, 2007.

[41] E. Ignall and L. E. Schrage, "Application of the branch and bound technique to some flow shop scheduling problems," *Journal of Operations Research*, vol. 13, no. 3, pp. 400–412, 1965.

[42] W. L. Maxwell and L. W. Miller, *Theory of Scheduling*, Addison-Wesley, Boston, Mass, USA, 1967.

[43] D. S. Palmer, "Sequencing jobs through a multistage process in the minimum total time—a quick method of obtaining a near optimum," *Journal of the Operational Research Society*, vol. 16, pp. 101–107, 1965.

[44] K. K. Cho, E. E. Enscore Jr., and I. Ham, "A heuristic algorithm for multistage group scheduling problem to minimize total tardiness," in *Proceedings of the Society of Manufacturing Engineers*, 1982.

[45] Y. F. Peng, Z. L. Guan, L. Ma, C. Y. Zhang, and P. G. Li, "A two-stage mathematical approach for the design of virtual manufacturing cells," *International Journal of Mechatronics and Manufacturing Systems*, vol. 2, no. 1-2, pp. 80–96, 2009.