

# ASPECT OF SELECTIVE REROUTING IN MULTICRITERIA SCHEDULING OF FLEXIBLE MANUFACTURING

Singh, A.

Gautam Buddha University, Greater Noida, INDIA

E-Mail: amolasingh2007@rediffmail.com

## Abstract

In this paper, a Multi-Criteria Scheduling Algorithm (*MCSA*) based on swapping of dispatching rules and Selective Rerouting (*SR*) approach based on lateral entry of critical jobs in the queue of alternate machine is used. Further, the effect of simultaneous application of *MCSA* and *SR* has been investigated. The system performance has been evaluated using mean flow time, maximum flow time, mean tardiness and maximum tardiness as the performance measures. Simulation studies using ProModel<sup>®</sup> simulation software were carried out at different breakdown levels for a range of part complexity and part mix, in three types of flexible job shops with Job/Machine ratio less than, equal to and greater than unity involving a total of 799 simulation runs. Each run of simulation spans for a period of 2000 completed jobs. The results establish that simultaneous application of *MCSA* and *SR* improves the system performance between 12.17 % to 28.73 % and 30.51 % to 47.67 % over that when *SR* and *MCSA* respectively are applied individually. The improvement is more at low breakdown levels and high part complexity.

(Received in December 2009, accepted in April 2010. This paper was with the author 1 month for 1 revision.)

**Key Words:** Selective Rerouting, Part Complexity, Part Mix, Breakdown Level, Performance Index

## 1. INTRODUCTION

Increasing demand for new products creates the need for production systems that are flexible and allow shorter product life cycle, smaller production lots, and large variety and ensure 100 % on-time delivery. Nowadays manufacturing is characterized by a consumer-oriented market giving more options and choices to the consumers. Hence, the significant trends in the present day market oriented manufacturing are large product variety and frequent design changes, which translate into low production volumes. Low quantity, high variety, small batch production requires a manufacturing system with a reasonably high flexibility not only in manufacturing equipments but also in design, planning and decision making. A rapid and timely response to market demands is the key to achieve high competitiveness. These challenges set the new trends in evolution of modern manufacturing systems that are increasingly becoming more complex. In these systems, planning and scheduling functions need to be addressed with much greater rigor and precision in order to achieve their effective utilization in terms of productivity.

Scheduling is an important function during part manufacturing and may be defined as the allocation of limited number of resources (machine, tool, fixture, operator, material handling equipment) to each task of a job over time. Scheduling may be static or dynamic. The nature of job arrivals provides the distinction between static and dynamic scheduling. In static problems, a certain number of jobs arrive simultaneously in a shop that is immediately available for work. No further jobs will arrive till the complete set is processed. Hence, all

attention can be focused to schedule this completely known and available set of jobs. However, in dynamic problems, the job arrival in the shop is a continuous process. Jobs arrive intermittently, at times that are predictable only in a statistical manner, and arrivals continue indefinitely into the future. Hence, entirely different methods are required to solve the problem of scheduling in these two cases. The feasibility of schedules is evaluated for various performance criteria, which may be single or multiple. A solution that is optimal for a given criterion may not be optimal for some other criterion. In many practical situations, it would be thus desirable to achieve a solution that is best with respect to a number of different criteria simultaneously. The research on bi-criteria and multi-criteria scheduling can be categorized in four different types of models, viz. single machine-bi-criteria scheduling, single machine-multiple criteria scheduling, multiple machine-bi-criteria scheduling and multiple machine-multi criteria scheduling. Most of the research done so far on multiple criteria scheduling involves only a single machine or two-machine flow shop [1-4]. Rajendran [1] addressed the two stage bi-criteria flow shop problem with the objective of minimizing total flow time and make-span using branch and bound and two heuristics based algorithms. Lio et al. [2] addressed the same problem with the objectives of minimizing makespan and number of tardy jobs, and makespan and total tardiness using a heuristic based on Johnson algorithm and Moore algorithm. Gupta et al. [3] addressed the two identical parallel machines bi-criteria scheduling problem with the objective of minimizing makespan and mean flow time using a heuristic that includes the integration of *SPT* rule with Ho and Wong's two machine optimization algorithm. Su and Chou [4] addressed the single machine bi-criteria problem with the objective of minimizing total completion time and completion time variance using heuristic scheduling algorithm. Koksalan & Keha [5] considered two single machine bi-criteria scheduling problems with the objectives of minimizing flow time and number of tardy jobs and minimizing flow time and maximum earliness by using genetic algorithm. Teghem et al. [6] have designed a multi-objective simulated annealing algorithm (*MOSA*). However, *MOSA* is more computationally expensive than dispatching rules. Loukil et al. [7] reported a review of research on multi-criteria and bi-criteria scheduling approaches and suggested that Interactive approach and Goal programming are more suitable in real case studies. They also considered multi-criteria flow shop and single machine scheduling problems with the objective of minimizing make-span, earliness, tardiness and number of tardy jobs using multi-objective tabu search (*MOTS*) and multi-objective simulated annealing (*MOSA*) algorithms.

Hoogeveen [8] reported a survey on multi-criteria scheduling. Their review indicates that most of the research on bi-criteria and multi-criteria scheduling has been carried out on single machine, flow shop, parallel machine configurations. Very little research has been carried out on job shop configurations. Multiple machines and multi-criteria scheduling represent the most general class of scheduling problems as most shop floors employ more than one machine. Hence, multiple machine scheduling models have extensive practical applications. Reported research reveals scarcity of research in this direction, apparently due to the complex nature of the scheduling problem. Most of the research in the area of production scheduling deals with predictive scheduling problems. However, reactive scheduling or real time schedule revision is also important for successful implementation of the scheduling function in manufacturing systems with continuously changing environment and revision of pre-established plans. Scheduling research has traditionally ignored this "process view" of the problem, and focuses on optimization of the performance measures under idealized assumptions of environmental stability. However, several authors have reported that these assumptions do not hold for most of the manufacturing environments due to random machine breakdowns, rush job arrival, tool breakage [9-12] etc.

In recent years, the above issues have been addressed in terms of development of several efficient dispatching rules [13-18], development of multi criteria scheduling algorithm by swapping of dispatching rules (*MCSA*) that minimizes/maximizes several performance measures simultaneously [19] and rerouting algorithms [20, 21] to counter the machine breakdowns in the presence of routing flexibility.

The focus of the present work is to address the multi criteria dynamic scheduling problem with machine breakdowns in a flexible job shop. Since, the traditional optimization techniques normally fail to yield a global solution to such a complicated problem, the problem is attempted using the dynamic scheduling approach in which jobs arrive on the shop floor randomly over time, so that the shop floor itself behaves like a network of queues, i. e. as soon as a machine becomes free, a dispatching rule is applied to decide what the machine will do next. The main aim of the present work is to enhance the performance of dynamic scheduling with respect to multiple performance measures and reduce the effect of shop floor disruptions. In this paper, *MCSA* and selective rerouting (*SR*) strategies are applied simultaneously. System performance is evaluated by multi criteria approach using mean and maximum flow time as the flow time based performance measures and mean and maximum tardiness as the tardiness based performance measures. The results of this study using 17 dispatching rules are compared with the performance measure obtained by selective rerouting and *MCSA* separately. Though already reported, [19, 22] the main steps of *MCSA* and selective rerouting that form the basis of the present paper are briefly described below for ease of comprehension.

## **2. MULTI-CRITERIA SCHEDULING ALGORITHM AND SELECTIVE REROUTING**

Consideration of more than one performance measure during scheduling is an important aspect in scheduling problems. In dynamic scheduling problems generally dispatching rules or priority heuristics are used to find the solution of the problems. However, a rule that is best for a given performance measure may give poor performance for others. Hence, it is necessary to select the dispatching rules in such a way that the system gives improved performance with respect to multiple performance measures. This issue is considered in the present work by using an algorithm to minimize / maximize several performance measures simultaneously. The salient features of the algorithm are as follows:

- the algorithm considers several dispatching rules simultaneously;
- it continuously monitors the attained value of the performance measures;
- the selection of dispatching rule is made by identifying the worst performance measure;
- a dispatching rule which improves system performance for the worst performance measure is selected;
- this algorithm improves one performance measure at the cost of others such that the gain from improvement in one measure is more than the deterioration in the performance of the other measure;
- again, the worst performance measure is identified and the steps are iteratively repeated till the end of the simulation period.

Further, the performance of the system with respect to multiple performance measures can be improved by reducing the effect of machine breakdowns. Machine breakdown is the most common problem faced by the shop management during scheduling. In this regard, several rescheduling or rerouting approaches have been proposed in literature to reduce the impact of machine breakdowns. Selective rerouting is one of the rerouting approach in which researchers have considered off line data for making the rerouting decisions. However, the

actual data essential for making the rerouting decision often do not match with the off line data, thereby affecting the performance of rerouting decisions. This is due to the fact that some time a part not needing rerouting may be rerouted. During rerouting, the rerouted part is placed at the end of the queue of the alternative machine. However, in some cases the rerouted part may carry a higher priority than some of the parts at the alternative machine. In view of these issues, in the present work a selective rerouting algorithm is used for making rerouting decisions by using online data on machine breakdowns with the provision of lateral entry of parts in queue of the alternative machines. Some of the important features of selective rerouting algorithm are given below:

- both the newly arrived jobs and the jobs in the queue of the failed machine are considered for rerouting;
- the waiting queue of the failed machine is rearranged on the basis of earliest due date or number of operations remaining dispatching rule;
- the parts are selectively rerouted by comparing the completion time of each job at the failed machine as well as the alternative machines;
- parts are entered laterally in the queue of the alternative machine;
- the parts are rerouted only if the benefit of rerouting is positive.

Architecture for the simultaneous use of *MCSA* and selective rerouting of parts is shown in Fig. 1 and the details of *MCSA* and selective rerouting algorithm can be seen in [19, 22].

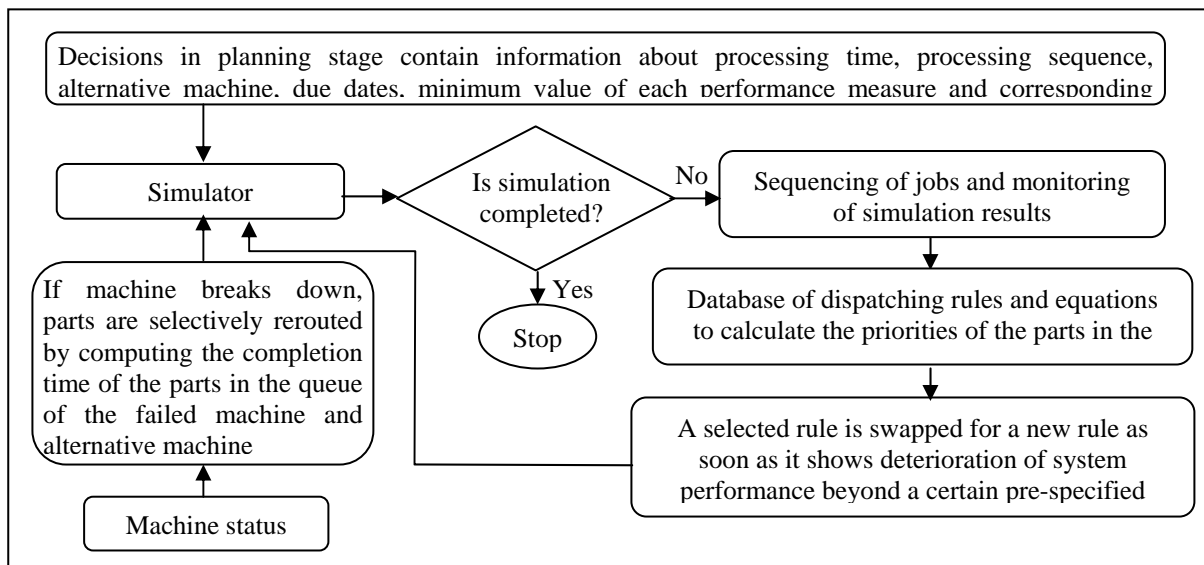


Figure 1: Architecture for simultaneous use of *MCSA* and selective rerouting of part.

### **3. SIMULATION MODEL**

The  $n \times m$  classical job shop scheduling problem involves  $n$  jobs and  $m$  machines, where, each job has its own processing sequence and each machine processes only one job at a time. In practice, the shop-floor setup usually consists of multiple copies of the most critical machines so that bottlenecks due to long operations or busy machines can be reduced. Hence, an operation may be carried out on more than one machine with the same function. This leads to a more complex problem known as the flexible job shop scheduling problem [10, 21]. In addition, for complex manufacturing systems, a job can usually visit a machine more than once (known as backtracking). These features of the Job shop significantly increase the complexity of finding the efficient solution of scheduling problem in these systems. In the

present investigation, the flexible job shop with backtracking in which each part type has alternate routes is considered. The simulation models of flexible job shop have been developed in ProModel<sup>®</sup> simulation software by considering ten machines and ten part types to investigate the performance of selective rerouting algorithm with MCSA. The number of operations for each job type is assumed to be uniformly distributed in the range of (2-9) with uniformly distributed processing times in the range of (1-9). Routing flexibility has been incorporated with the provision of backtracking in the operation sequences. The inter-arrival times for jobs are generated from the exponential distribution with mean calculated from the equation,  $\lambda = p/M \times U$  [13]. Where,  $p$  represents the mean processing time of a job type,  $M$  is the number of machines and  $U$  the shop utilization.

In the present work, a manufacturing system is defined by two sets of parameters: system operating parameters and system configuration parameters. The system operating parameters are breakdown level ( $BL$ ) and mean time to repair ( $MTTR$ ), while the system configuration parameters are number of job types to number of machine types ( $J/M$ ) ratio, part complexity ( $PC$ ) and part mix ( $PM$ ). These parameters along with their levels are discussed below:

### 3.1 Break down level

Machine breakdown level ( $BL$ ) is defined by the ratio of  $MTTR$  over the sum of  $MTTR$  and  $MTBF$ .

$$BL = MTTR / (MTTR + MTBF) \quad (1)$$

Five values of breakdown levels i.e. 0, 2.5, 5, 7.5 and 10 % have been considered.

### 3.2 Mean Time to Repair

This is the average time to repair a machine and bring it back to acceptable operating condition. It includes the actual time spent on arranging spares and resources and then restoring the machine to make it operation worthy. Five levels of mean time to repair i. e.  $p$ ,  $2.5p$ ,  $5p$ ,  $7.5p$  and  $10p$  have been considered, where  $p$  is the average processing time of a job.

Here it is pertinent to note that the frequency of breakdown is inversely proportional to the mean time between failures ( $MTBF$ ). Hence, for a given  $BL$ , it will be inversely proportional to the  $MTTR$ . For example the number of breakdowns for  $MTTR = 2.5p$  will be twice the number of breakdowns for  $MTTR = 5p$  as described below:

$$BL = \frac{MTTR}{MTTR + MTBF} \quad (2)$$

$$MTBF = \frac{MTTR - (BL \times MTTR)}{BL} \quad (3)$$

$$\text{IF } MTTR = 2.5p, \text{ then } MTBF = \frac{2.5p - BL \times 2.5p}{BL} = \alpha$$

$$\text{IF } MTTR = 5p, \text{ then } MTBF = \frac{5p - BL \times 5p}{BL} = 2\alpha$$

### 3.3 Part complexity

Part complexity ( $PC$ ) is defined as the number of operations needed to complete the processing of a job [23]. In the present work, part complexity is defined by the average number of operations needed to complete the processing of a job and four; five and six operations have been selected for three levels of complexity. A job requiring more operations to complete its processing is said to be more complex than a job that requires fewer operations.

### 3.4 Shop configuration

In the present work, ten machines and eight, ten and sixteen job types have been selected for study with three levels of shop configurations. In the first configuration, number of job types is less than the number of machine types  $J < M$  (i. e. 8 job types with 10 machine types). In the second configuration, number of job types is equal to the number of machine types  $J = M$  (i.e. 10 job types with 10 machine types) and in the third configuration, number of the job types is more than number of machine types  $J > M$  (i.e. 16 job types with 10 machine types).

Table I: Routing with alternative machines and operation processing times for  $J = 8$  and  $M = 10$ .

		Processing Sequence									
		<b>8 Job types and 10 Machine types with PC = 6</b>									
		<i>M1</i>	<i>M2</i>	<i>M3</i>	<i>M4</i>	<i>M5</i>	<i>M6</i>	<i>M7</i>	<i>M8</i>	<i>M9</i>	<i>M10</i>
Job Types	1	O <sub>1</sub> (6)	<b>O<sub>2</sub>(6)</b>	<b>O<sub>1</sub>(7)</b>		O <sub>2</sub> (5)			O <sub>3</sub> (8)		<b>O<sub>3</sub>(9)</b>
	2	<b>O<sub>1</sub>(9)</b>	O <sub>1</sub> (8)	<b>O<sub>2</sub>(6)</b>	O <sub>2</sub> (5)		O <sub>3</sub> (7)	<b>O<sub>3</sub>(8)</b>		<b>O<sub>4</sub>(7)</b>	O <sub>4</sub> (6)
	3	O <sub>1</sub> (3)	<b>O<sub>1</sub>(4)</b>	O <sub>2</sub> (8)	<b>O<sub>2</sub>(9)</b>	O <sub>3</sub> (2)	O <sub>4</sub> (6)	O <sub>8</sub> (7)	O <sub>6</sub> (4)	O <sub>5</sub> (1)	O <sub>7</sub> (8)
		<b>O<sub>3</sub>(3)</b>				<b>O<sub>4</sub>(7)</b>	<b>O<sub>5</sub>(2)</b>	<b>O<sub>6</sub>(5)</b>	<b>O<sub>7</sub>(8)</b>		
	4	O <sub>2</sub> (4)	<b>O<sub>1</sub>(9)</b>	O <sub>4</sub> (2)	O <sub>1</sub> (9)	<b>O<sub>4</sub>(3)</b>	<b>O<sub>3</sub>(4)</b>	O <sub>5</sub> (5)		<b>O<sub>5</sub>(6)</b>	O <sub>3</sub> (3)
					<b>O<sub>2</sub>(5)</b>						
	5	<b>O<sub>1</sub>(9)</b>	O <sub>6</sub> (3)	O <sub>1</sub> (9)	<b>O<sub>3</sub>(7)</b>	O <sub>5</sub> (7)	O <sub>4</sub> (5)	O <sub>2</sub> (4)	<b>O<sub>4</sub>(6)</b>	O <sub>3</sub> (6)	<b>O<sub>5</sub>(8)</b>
	6	O <sub>3</sub> (4)	O <sub>4</sub> (9)	O <sub>2</sub> (1)	O <sub>1</sub> (6)	O <sub>6</sub> (7)	O <sub>5</sub> (3)	O <sub>7</sub> (5)	<b>O<sub>4</sub>(9)</b>	<b>O<sub>5</sub>(4)</b>	<b>O<sub>7</sub>(6)</b>
<b>O<sub>1</sub>(7)</b>		<b>O<sub>2</sub>(2)</b>		<b>O<sub>3</sub>(5)</b>		<b>O<sub>6</sub>(8)</b>					
7	<b>O<sub>4</sub>(7)</b>	O <sub>4</sub> (6)	<b>O<sub>1</sub>(6)</b>	O <sub>1</sub> (5)	<b>O<sub>2</sub>(5)</b>	O <sub>2</sub> (4)	<b>O<sub>3</sub>(8)</b>	O <sub>5</sub> (5)	O <sub>3</sub> (7)	O <sub>6</sub> (5)	
						<b>O<sub>5</sub>(6)</b>		<b>O<sub>6</sub>(6)</b>			
8	O <sub>3</sub> (5)	O <sub>4</sub> (4)	O <sub>1</sub> (8)	O <sub>5</sub> (3)	O <sub>6</sub> (2)	<b>O<sub>3</sub>(6)</b>	O <sub>7</sub> (6)	O <sub>9</sub> (1)	O <sub>2</sub> (4)	O <sub>8</sub> (3)	
	<b>O<sub>7</sub>(7)</b>	<b>O<sub>1</sub>(8)</b>	<b>O<sub>4</sub>(5)</b>	<b>O<sub>9</sub>(2)</b>	<b>O<sub>2</sub>(5)</b>		<b>O<sub>5</sub>(4)</b>	<b>O<sub>6</sub>(3)</b>	<b>O<sub>8</sub>(4)</b>		
		<b>8 Job types and 10 Machine types with PC = 5</b>									
		<i>M1</i>	<i>M2</i>	<i>M3</i>	<i>M4</i>	<i>M5</i>	<i>M6</i>	<i>M7</i>	<i>M8</i>	<i>M9</i>	<i>M10</i>
Job Types	1	<b>O<sub>1</sub>(6)</b>	O <sub>2</sub> (4)		O <sub>1</sub> (5)	<b>O<sub>2</sub>(5)</b>			O <sub>3</sub> (9)		<b>O<sub>3</sub>(9)</b>
	2		<b>O<sub>1</sub>(9)</b>	<b>O<sub>2</sub>(8)</b>		O <sub>1</sub> (8)	O <sub>2</sub> (7)	<b>O<sub>3</sub>(6)</b>	O <sub>3</sub> (5)	<b>O<sub>4</sub>(7)</b>	O <sub>4</sub> (6)
	3	O <sub>3</sub> (4)	O <sub>1</sub> (9)	O <sub>2</sub> (1)	O <sub>4</sub> (6)	O <sub>6</sub> (7)	O <sub>5</sub> (3)	O <sub>7</sub> (5)	<b>O<sub>4</sub>(7)</b>	<b>O<sub>6</sub>(8)</b>	<b>O<sub>7</sub>(6)</b>
		<b>O<sub>2</sub>(2)</b>			<b>O<sub>1</sub>(9)</b>	<b>O<sub>5</sub>(4)</b>	<b>O<sub>3</sub>(5)</b>				
	4	O <sub>1</sub> (3)	<b>O<sub>1</sub>(4)</b>	O <sub>2</sub> (8)	<b>O<sub>2</sub>(9)</b>	O <sub>3</sub> (2)	O <sub>4</sub> (6)	<b>O<sub>3</sub>(3)</b>	<b>O<sub>4</sub>(7)</b>	O <sub>5</sub> (7)	<b>O<sub>5</sub>(8)</b>
	5	<b>O<sub>1</sub>(6)</b>	O <sub>1</sub> (5)	<b>O<sub>2</sub>(6)</b>		<b>O<sub>3</sub>(5)</b>	<b>O<sub>4</sub>(7)</b>	O <sub>4</sub> (6)	O <sub>3</sub> (4)	O <sub>2</sub> (5)	
	6	O <sub>1</sub> (4)	<b>O<sub>1</sub>(5)</b>	O <sub>4</sub> (2)	O <sub>2</sub> (9)	<b>O<sub>4</sub>(3)</b>	<b>O<sub>2</sub>(9)</b>	O <sub>5</sub> (5)	<b>O<sub>3</sub>(4)</b>	<b>O<sub>5</sub>(6)</b>	O <sub>3</sub> (3)
	7	O <sub>3</sub> (5)		<b>O<sub>1</sub>(5)</b>	O <sub>1</sub> (4)		<b>O<sub>4</sub>(8)</b>	<b>O<sub>2</sub>(4)</b>	O <sub>5</sub> (8)	O <sub>2</sub> (3)	O <sub>4</sub> (7)
				<b>O<sub>3</sub>(6)</b>					<b>O<sub>5</sub>(9)</b>		
8	<b>O<sub>1</sub>(9)</b>	O <sub>6</sub> (3)	O <sub>1</sub> (9)	<b>O<sub>3</sub>(7)</b>	O <sub>5</sub> (7)	O <sub>4</sub> (5)	O <sub>2</sub> (4)	<b>O<sub>4</sub>(6)</b>	O <sub>3</sub> (6)	<b>O<sub>5</sub>(8)</b>	
		<b>O<sub>2</sub>(5)</b>					<b>O<sub>6</sub>(4)</b>				
		<b>8 Job types and 10 Machine types with PC = 4</b>									
		<i>M1</i>	<i>M2</i>	<i>M3</i>	<i>M4</i>	<i>M5</i>	<i>M6</i>	<i>M7</i>	<i>M8</i>	<i>M9</i>	<i>M10</i>
Job Types	1	<b>O<sub>1</sub>(7)</b>		<b>O<sub>2</sub>(5)</b>	O <sub>1</sub> (6)	<b>O<sub>3</sub>(6)</b>	<b>O<sub>4</sub>(6)</b>		O <sub>4</sub> (5)	O <sub>3</sub> (5)	O <sub>2</sub> (4)
	2		<b>O<sub>1</sub>(9)</b>	O <sub>1</sub> (8)				<b>O<sub>2</sub>(7)</b>	O <sub>2</sub> (7)		
	3	O <sub>1</sub> (3)	<b>O<sub>2</sub>(9)</b>	O <sub>2</sub> (8)	<b>O<sub>1</sub>(4)</b>	O <sub>3</sub> (2)	O <sub>4</sub> (6)	<b>O<sub>4</sub>(7)</b>	<b>O<sub>3</sub>(3)</b>	O <sub>5</sub> (7)	<b>O<sub>5</sub>(8)</b>
	4	O <sub>1</sub> (8)		<b>O<sub>1</sub>(9)</b>		O <sub>2</sub> (5)	<b>O<sub>3</sub>(8)</b>		O <sub>3</sub> (7)	<b>O<sub>2</sub>(6)</b>	
	5	<b>O<sub>1</sub>(7)</b>	O <sub>1</sub> (6)		O <sub>2</sub> (5)	<b>O<sub>2</sub>(6)</b>	O <sub>3</sub> (4)		<b>O<sub>3</sub>(5)</b>	<b>O<sub>4</sub>(9)</b>	O <sub>4</sub> (8)
	6	<b>O<sub>2</sub>(9)</b>	O <sub>2</sub> (9)	<b>O<sub>1</sub>(5)</b>	O <sub>1</sub> (4)			O <sub>3</sub> (6)			<b>O<sub>3</sub>(6)</b>
	7	O <sub>2</sub> (2)	<b>O<sub>2</sub>(3)</b>	O <sub>4</sub> (5)	<b>O<sub>1</sub>(5)</b>	<b>O<sub>3</sub>(9)</b>		O <sub>1</sub> (4)		<b>O<sub>4</sub>(6)</b>	O <sub>3</sub> (9)
	8	<b>O<sub>1</sub>(9)</b>	O <sub>6</sub> (3)	O <sub>1</sub> (9)	<b>O<sub>2</sub>(5)</b>	O <sub>5</sub> (7)	O <sub>4</sub> (5)	O <sub>2</sub> (4)	<b>O<sub>3</sub>(7)</b>	O <sub>3</sub> (6)	<b>O<sub>5</sub>(8)</b>
						<b>O<sub>6</sub>(4)</b>	<b>O<sub>4</sub>(6)</b>				

**Legend:**  $M1, M2 \dots M10$  in Table I denote the machine types, numbers 1, 2, 3 ...8 in the second column denote the job types,  $O_1, O_2, O_3 \dots O_9$  denote  $i^{\text{th}}$  operation of a job, bold  $O_1, O_2, O_3 \dots O_9$  are the alternate operations, figures in parentheses are the processing time, figures in bold parentheses are the processing time at the alternative machines due to its different processing capabilities.

### 3.5 Part mix

Three levels of part mix ( $PM$ ) have been considered for evaluating the performance of multi-criterion methodology. Part mix is based on the number of processing operations required in a part type. In the first part mix ( $PM1$ ), it is assumed that all part types have equal volume of production. The second ( $PM2$ ) and third ( $PM3$ ) part mix have been generated by considering the industrial data given by Muhlemann [24]. The data of their survey is given below, from which a cumulative probability distribution has been generated for obtaining the second and third part mix.

Number of operations	:	1	2	3	4	5	6	7	8	9	10	11	12
Percentage of jobs	:	0.1	27.5	27.1	14.2	14.1	6.5	5.3	3.0	0.8	0.7	0.5	0.2

The processing times data are generated in the range of 1-10 time units and the processing sequence data for job types with the above discussed shop configuration parameters are randomly generated. A randomly generated flexible job shop of 8 job types and 10 machines with three levels of part complexities is shown in Table I. Similar type of flexible job shop of '16 job types and 10 machines' and '10 job types and 10 machines' are used for investigation.

As the performance of selective rerouting algorithm with  $MCSA$  methodology is evaluated in the presence of machine breakdowns, down time of individual machines has been incorporated irrespective of whether it occurs due to tool breakage, tool adjustment, machine breakdown etc. However, breakdowns, such as those due to power failure, that affect the whole system are not considered. Mean time between failure ( $MTBF$ ) and mean time to repair ( $MTTR$ ) are assumed to follow Gamma distribution. In the simulation model, busy time approach has been chosen for generating the breakdown times (i. e. a machine can fail only while performing an operation on a job). Law and Kelton [25] suggested that in the absence of real time data busy time distribution is most likely to be a Gamma distribution with a shape parameter of 0.7. They also suggested that Gamma distribution with a shape parameter of 1.4 is appropriate for generating the repair time. Thus the busy time between two successive failures (i.e. inter-breakdown time) is assumed to follow a Gamma distribution with  $\alpha = 0.7$  and  $\beta = MTTR \times e / (1-e) \times 0.7$  and the duration of each breakdown (which is also known as repair time) is also assumed to follow a Gamma distribution with  $\alpha = 1.4$  and  $\beta = MTTR / 1.4$ . Where,  $e = MTBF / (MTBF + MTTR)$ . Machine breakdown level is inversely proportional to the availability and mean time to repair ( $MTTR$ ) is the average time to repair a machine and bring it back to acceptable operating condition. In the present research, simulation model has been run for 2000 completed jobs after attaining a steady state.

## 4. EVALUATION OF PROPOSED METHODOLOGY

The objective function ( $OF$ ) is defined to minimize the flow time based and tardiness based performance measures (mean flow time, maximum flow time, mean tardiness and maximum tardiness) simultaneously. The performance of selective rerouting and  $MCSA$  simultaneously with multi objectives has been investigated at different breakdown parameters and configuration parameters (i.e. part complexity, part mix and job types to number of machine ratio). Here, part complexity is defined by the average number of operations needed to

complete the processing of a job. As tardiness based performance measures are also taken into consideration, in this analysis the due date of an arriving job is found from the following total work content relation [26, 27].

$$D_i = A_i + (C \times \sum_{k=1}^{k=n_i} P_{ik}) \quad (4)$$

Where  $C$  = due date allowance factor,  $A_i$  = arrival time of job  $i$  at the shop floor,  $p_{ik}$  = processing times of job  $i$  for its  $k^{th}$  operation and  $n_i$  = total number of operations in a job. In this research, seventeen dispatching rules viz. *SPT*, *EDD*, *FIFO*, *ODD*, *AT*, *AT-RPT*, *FDD*, *PT+PW*, *PT+PW+FDD*, *PT+PW+ODD*, *WTIS/CPT+COPT*, *(WTIS/PT)*, *(WTIS/PT)max*, *PT+WTIS*, *FDD+WTIS/CPT+COPT*, *ODD+WTIS/CPT+COPT* and *(WTIS)max* have been considered and simulation has been carried out by taking one rule at a time. Where, *ODD* = operational due date, *AT* = arrival time, *RPT* = remaining processing time, *FDD* = flow due date, *PT* = process time, *PW* = process wait, *WTIS* = waiting time in system, *CPT* = completed processing time and *COPT* = current operation processing time. The description of these dispatching rules can be found in [13-18]. Here, performance index (*PI*) of each dispatching rule and *MCSA* is found by summing up the  $(\sum f_j / f_{min,j})$  values, where  $f_j$  is the performance value of  $j^{th}$  performance measure for individual dispatching rules and  $f_{min,j}$  is the minimum value of  $j^{th}$  performance measure among all the dispatching rules. After each simulation run the values of the four performance measures considered in (*OF*) are singled out and the best rule for each performance measures is identified as the one having the least value. The performance index (*PI*) of each dispatching rule is then calculated by adding the four considered performance measures. After determining these parameters, the simulation is continued by applying the *MCSA* and selective rerouting simultaneously for minimizing the considered performance measures.

The improvement in the system performance by using the threshold based swapping of rules and selective rerouting and *MCSA* is calculated by using the following relationship.

$$\text{Percentage improvement} = (PI_i - PI_{minsr}) / (PI_{minsr} \times 100) \quad (5)$$

$$\text{Percentage improvement} = (PI_i - PI_{mcsa}) / (PI_{mcsa} \times 100) \quad (6)$$

Where  $PI_i$  is the performance index of the system by swapping of dispatching rules and selective rerouting while  $PI_{minsr}$  is the minimum performance index among all the individual dispatching rules with selective rerouting and  $PI_{mcsa}$  is the performance index of *MCSA*. The results of this analysis are presented in the following section.

## **5. RESULTS AND DISCUSSION**

The performance of the multi criteria scheduling algorithm (*MCSA*) with selective rerouting is evaluated by varying the breakdown level and mean time to repair values with part complexity (average number of operations to complete the job)  $PC = 6$ , job type to machine ratio = 1 (i.e. 10 job types with 10 machine) and with a part mix in which, it is assumed that all part types have equal volume of production for objective function *OF*. For determining the minimum value of each performance measure and the performance index of individual dispatching rules with *MCSA* and selective rerouting, four breakdown levels (2.5 %, 5 %, 7.5 %, 10 %), five *MTTR* levels ( $p$ ,  $2.5p$ ,  $5p$ ,  $7.5p$ ,  $10p$ , where  $p$  is the mean processing time of the jobs) and 17 dispatching rules are considered as variables and 340 ( $4 \times 5 \times 17$ ) simulation runs were carried out.

The values of objective function (*OF*) in terms of percentage improvement obtained by applying the *MCSA* and selective rerouting simultaneously in comparison with the best performing individual dispatching rule with selective rerouting as defined in section 4 at

different values of  $MTTR$  and breakdown level is summarized in Table II. The percentage improvement is 27.85 % at 2.5 % breakdown level and 16.11 % at 10 % breakdown level, when the repair time is  $2.5p$ . Similarly, the percentage improvement is 20.86 % at 2.5 % breakdown level and 16.56 % at 10 % breakdown level, when the repair time is  $10p$ . This trend indicates that as the breakdown level increases, improvement in the performance index decreases. This is because, if a machine has a failure with a long breakdown time, this machine becomes a large bottleneck, increases the waiting time and correspondingly the flow time and tardiness of each of the waiting jobs. Similarly, at low breakdown levels, higher percentage improvement has been observed when there are few breakdowns of long duration compared with the case when there are relatively more breakdowns of short duration under identical breakdown level. This may be attributed to enhanced instability of the system when subjected to large number of breakdowns.

Table II: % improvement in  $PI$  of  $OF$  by using with  $MCSA$  and  $SR$  simultaneously at varying breakdown parameters in comparison with the best individual dispatching rule with  $SR$ .

$MTTR$	Breakdown level ( $BL$ )			
	2.5 %	5 %	7.5 %	10 %
p	17.23	21.47	16.99	14.23
$2.5p$	27.85	26.77	26.19	16.11
$5p$	26.19	23.43	18.77	14.52
$7.5p$	21.04	23.71	20.21	16.32
$10p$	20.86	24.59	21.03	16.56

Table III: % improvement in  $PI$  of  $OF$  by using with  $MCSA$  and  $SR$  simultaneously at varying breakdown parameters in comparison with the  $MCSA$ .

$MTTR$	Breakdown level ( $BL$ )			
	2.5 %	5 %	7.5 %	10 %
p	31.69	37.41	33.97	30.69
$2.5p$	43.55	38.31	34.57	32.33
$5p$	39.53	37.29	36.31	30.56
$7.5p$	31.76	35.13	31.63	31.96
$10p$	34.58	40.77	37.09	32.68

The values of objective function ( $OF$ ) in terms of percentage improvement obtained by applying the  $MCSA$  and selective rerouting simultaneously in comparison with the  $MCSA$  at different values of  $MTTR$  and breakdown level is summarized in Table III. The percentage improvement is 43.55 % at 2.5 % breakdown level and 32.33 % at 10 % breakdown level, when the repair time is  $2.5p$ . Similarly, the percentage improvement is 34.58 % at 2.5 % breakdown level and 32.68 % at 10 % breakdown level, when the repair time is  $10p$ . In this case, higher percentage improvement is observed as compared with the previous case. This is because by using  $MCSA$  separately the parts rest in the queue of the failed machine when it breaks down, thereby increasing the flow time as well as tardiness of the parts.

The performance of the multi criteria scheduling algorithm ( $MCSA$ ) with selective rerouting is evaluated by varying the part complexity (i.e.  $PC=4$ ,  $PC=5$  and  $PC=6$ ), part mix (i.e.  $PM1$ ,  $PM2$  and  $PM3$ ) and Job types to number of machine ratio (i.e. 16 job types with 10 machines ( $J/M>1$ ), 10 job types with 10 machines ( $J/M=1$ ) and 8 job types with 10 machines ( $J/M<1$ )) at breakdown level = 5 % and  $MTTR = 5p$  for objective function  $OF$ . Here, first part mix is generated as defined above. However, the industrial data given by Muhlemann et al. 1982 is considered in defining the second ( $PM2$ ) and third part mix ( $PM3$ ).

The values of objective function (*OF*) in terms of percentage improvement obtained by applying the *MCSA* and selective rerouting simultaneously in comparison with the best performing individual dispatching rule with selective rerouting as defined in section 4 at different values of part complexity, part mix and job types to number of machine ratio is summarized in Table IV. Percentage improvement of 28.73 % and 25.45 % is observed when  $J/M > 1$  and  $J/M < 1$  for the first part mix, 28.53 % and 19.11 % for the second part mix and 19.26 % and 14.33 % for the third part mix. This trend indicates that an increase in  $J/M$  ratio increases the percentage improvement in the performance index. Results presented in Table IV indicate that the proposed methodology works more efficiently in a system that has high part complexity in comparison to low part complexity. In all the taken cases, the proposed methodology improves the performance index in the range of 12.17 % to 28.73 %.

The objective function (*OF*) in terms of percentage improvement obtained by applying the *MCSA* and *SR* simultaneously in comparison with the *MCSA* at different values of part complexity, part mix and job types to number of machine ratio is summarized in Table V. Percentage improvement of 46.19 % and 45.35 % is observed when  $J/M > 1$  and  $J/M < 1$  for the first part mix, 41.59 % and 40.37 % for the second part mix and 46.23 % and 37.59 % for the third part mix. The range of percentage improvement lies in the range of 46.19 % to 33.65 % which is significantly higher than in the previous case because, by using *MCSA* separately the parts rest in the queue of the failed machine when machine breaks down thereby increasing flow time and tardiness both.

Table IV: % improvement in *PI* of *OF* by using with *MCSA* and *SR* simultaneously at varying configuration parameters in comparison with the best individual dispatching rule with *SR*.

Part mix	Part complexity	Percentage improvement		
		$J/M > 1$	$J/M = 1$	$J/M < 1$
PM1	4	20.93	17.26	19.18
	5	27.54	20.16	25.00
	6	28.73	23.43	25.45
PM2	4	20.17	12.17	16.92
	5	27.30	13.16	17.51
	6	28.53	14.17	19.11
PM3	4	16.19	14.36	14.31
	5	18.54	14.59	14.52
	6	19.26	17.05	14.33

Table V: % improvement in *PI* of *OF* by using with *MCSA* and selective rerouting simultaneously at varying configuration parameters in comparison with the *MCSA*.

Part mix	Part complexity	Percentage improvement		
		$J/M > 1$	$J/M = 1$	$J/M < 1$
PM1	4	32.79	34.57	36.62
	5	36.62	40.48	41.23
	6	46.19	37.29	45.35
PM2	4	30.51	39.41	39.76
	5	38.90	35.51	37.53
	6	41.59	47.67	40.37
PM3	4	33.71	40.49	36.39
	5	37.71	41.13	33.65
	6	46.23	45.15	37.59

## **6. CONCLUSIONS**

In the present research work, an attempt has been made to improve the performance of a flexible job shop of dynamic nature by applying the multi-criteria scheduling algorithm (*MCSA*) and selective rerouting simultaneously for flow time based and tardiness based performance measures in the form of *OF*. Several values of breakdown parameters with constant part complexity and part mix and several values of part mix, part complexity and job type to number of machine ratio with constant breakdown parameters have been taken to investigate the performance of the proposed approach. It has been noted that system performance is enhanced by the simultaneous use of *MCSA* and selective rerouting in comparison with the system performance by using *MCSA* and selective rerouting separately. This is because, by using *MCSA* and selective rerouting simultaneously the swapping of dispatching rules is allowed when the deterioration in the worst performing measure reaches a pre-defined limit and selective rerouting of parts is done during machine breakdowns. It has been demonstrated that at different breakdown parameters with constant part mix, part complexity and job type to number of machine ratio, the simultaneous use of *MCSA* and selective rerouting improves the system performance between 14.23 % to 27.85 % and 30.56 % to 43.55 % in comparison with selective rerouting and *MCSA* respectively. Similarly, at different configuration parameters with constant breakdown parameters, the simultaneous use of *MCSA* and selective rerouting improves the system performance between 12.17 % to 28.73 % and 30.51 % to 47.67 % in comparison with selective rerouting and *MCSA* respectively. Hence, finally it may be concluded that the integrated application of *MCSA* and selective rerouting significantly improves the system performance.

## **REFERENCES**

- [1] Rajendran, C. (1992). Two-stage flow shop scheduling problems with bi-criteria, *Journal of Operational Research Society*, Vol. 43, No. 9, 871-884
- [2] Lio, C. J.; Yu, W. C.; Joe, C. B. (1997). Bi-criteria scheduling in the two machine flow shop, *International Journal of Production Research*, Vol. 53, No. 9, 1004-1015
- [3] Gupta, J. N. D.; Ho, J. C.; Webster, S. (2000). Bi criteria optimization of the make span and mean flow time on two identical parallel machine, *Journal of Operational Research Society*, Vol. 51, No. 11, 1330-1339
- [4] Su, L. H.; Chou, F. D. (2001). Heuristic for scheduling in a bi-criteria single machine problem, *Journal of Chinese Institute of Industrial Engineers*, Vol. 18, No. 5, 39-46
- [5] Koksalan, M.; Keha, A. B. (2003). Using genetic algorithms for single-machine bicriteria scheduling problems, *European Journal of Operational Research*, Vol. 145, No. 3, 543-556
- [6] Teghem, J.; Tuyttens, D.; Ulungu, E. L. (2000). An interactive heuristic method for multi objective combinatorial optimization, *Computer and Operations research*, Vol. 27, 621-634
- [7] Loukil, T.; Teghem, J.; Tuyttens, D. (2005). Solving multi-objective production scheduling problems using metaheuristics, *European Journal of Operational Research*, Vol. 161, No. 1, 42-61
- [8] Hoogeveen, H. (2005). Multicriteria scheduling, *European Journal of Operational Research*, Vol. 161, No. 3, 592-623
- [9] Jain, A. K.; Elmaraghy, H. A. (1997). Production scheduling/rescheduling in flexible manufacturing, *International Journal of Production Research*, Vol. 35, No. 1, 281-309
- [10] Kutanoglu, E.; Sabuncuoglu, I. (2001). Routing based reactive scheduling policies for machine failures in dynamic job shop, *International journal of production research*, Vol. 39, No. 14, 3141-3158
- [11] Shafaei, R.; Brunn, P. (1999). Workshop scheduling using practical (inaccurate) data - Part 2: An investigation of the robustness of scheduling rules in a dynamic and stochastic environment, *International Journal of Production Research*, Vol. 37, No. 18, 4105-4117

- [12] Sabuncuoglu, I.; Bayiz, M. (2000). Analysis of reactive scheduling problems in a job shop environment, *European Journal of Operational Research*, Vol. 126, No. 3, 567-586
- [13] Holthaus, O.; Rajendran, C. (1997). Efficient dispatching rules for scheduling in a job shop, *International Journal of Production Economics*, Vol. 48, No. 1, 87-105
- [14] Jayamohan, M. S.; Rajendran, C. (2000). New dispatching rules for shop scheduling: a step forward, *International Journal of Production Research*, Vol. 38, No. 3, 563-586
- [15] Singh, A.; Mehta, N. K.; Jain, P. K. (2004). Performance analysis of flow time based dispatching rules with unreliable machines, *International Journal of Simulation Modelling*, Vol. 3, No. 4, 109-120
- [16] Singh, A.; Mehta, N. K.; Jain, P. K. (2005). Tardiness based new dispatching rules for shop scheduling with unreliable machines, *International Journal of Simulation Modelling*, Vol. 4, No. 1, 5-16
- [17] Jain, A.; Jain, P. K.; Singh, I. P. (2004). An investigation on the performance of dispatching rules in FMS scheduling, *International Journal of Simulation Modelling*, Vol. 3, No. 2-3, 49-60
- [18] Buchmeister, B.; Kremljak, Z.; Pandza, K.; Polajnar, A. (2004). Simulation study on the performance analysis of various sequencing rules, *International Journal of Simulation Modelling*, Vol. 3, No. 2-3, 80-89
- [19] Singh, A.; Mehta, N. K.; Jain, P. K. (2007). Multi-criterion dynamic scheduling with the swapping of dispatching rules, *International Journal of Advanced Manufacturing Technology*, Vol. 34, No. 9-10, 988-1007
- [20] Holthaus, O. (1999). Scheduling in job-shop with machine breakdowns: an experimental study, *Computer & Industrial Engineering*, Vol. 36, No. 1, 137-162
- [21] Mason, S. J.; Oey, K. (2003). Scheduling complex job shops using disjunctive graphs: a cycle elimination procedure, *International Journal of Production Research*, Vol. 41, No. 5, 981-994
- [22] Singh, A.; Mehta, N. K.; Jain, P. K. (2003). Selective approach for rerouting in the presence of machine breakdowns, *Annals of DAAAM and Proc. 14<sup>th</sup> International DAAAM Symposium*, DAAAM International Vienna, Austria, 417-418
- [23] Liu, J.; MacCarthy, B. L. (1996). The classification of FMS scheduling problems. *International Journal of Production Research*, Vol. 34, No. 3, 647-656
- [24] Muhlemann, A. P.; Lockett, A. G.; Farn, C. K. (1982). Job shop scheduling heuristics and frequency of scheduling, *International Journal of Production Research*, Vol. 20, No. 2, 227-241
- [25] Law, A. M.; Kelton, W. D. (2000). *Simulation Modeling and Analysis*, Third Edition, McGraw-Hill, New-York
- [26] Blackstone, J. H.; Phillips, D. T.; Hogg, G. L. (1982). A state of the art survey of dispatching rules for manufacturing job shop operations, *International Journal of Production Research*, Vol. 20, No. 1, 27-45
- [27] Haupt, R. (1989). A survey of priority rule based dispatching. *OR Spektrum*, Vol. 11, No. 1, 3-16