A Generic Mechanism for Repairing Job Shop Schedules

Amritpal Singh Raheja, K. Rama Bhupal Reddy, and Velusamy Subramaniam

Abstract— **Reactive repair of a disrupted schedule is a better alternative to total rescheduling, as the latter is a time consuming process and also results in shop floor nervousness. The schedule repair heuristics reported in the literature generally address only machine breakdown. This paper presents a modified Affected Operations Rescheduling (mAOR) approach, which deals with many of the disruptions that are frequently encountered in a job shop. The repair of these disruptions has been decomposed into four generic repair actions that can be applied singularly or in combination. These generic repair actions are evaluated through a simulation study with the performance measures of efficiency and stability. The results indicate the effectiveness of the mAOR heuristic in dealing with typical job shop disruptions.**

Index Terms— **Schedule Repair, Job shop disruptions.**

I. Introduction

Manufacturing like other fields of engineering, has to cope with uncertainties and disruptions. Therefore, an effective scheduling and control system is becoming crucial for modern manufacturing systems. Though disruptions are very much undesirable, they are the norm in any shop floor. If some disruptions are unavoidable, their effects on the overall performance of the system should be minimized. From this perspective, it is almost inevitable for researchers to focus on the realities of the underlying environment, where unexpected events and various kinds of interruptions occur at any time. Repair of a schedule is a process of altering a given schedule partially due to the occurrence of unexpected events/disruptions. In normal practice, the shop foreman manually carries out these adjustments based on his experience and common sense. However, his ability to foresee the cascaded effects of these changes on the shop floor is very limited, and often results in inefficient and irreversible repairs. To alleviate such errors, reactive schedule repair tools are essential.

Reactive scheduling is a process of revising a given schedule in real time due to the occurrence of unexpected events during the execution of the schedule. This can be broadly defined as the continuous adaptation and improvement of pre-computed predictive schedules. Thus, reactive scheduling can be seen as offline scheduling. It is quite possible that a new schedule might differ considerably from the initial schedule. In certain situations, this is not desirable since many other decisions like: delivery of goods, assignment of operators, material-handling devices and subsequent processing of jobs in other facilities could get affected. Total rescheduling will be preferred to repairing a disrupted schedule, when repairs adversely affect the system's performance. Total rescheduling leads to a new optimized predictive schedule and will certainly be superior in quality to any repaired schedule. Fig.1 illustrates the difference between total rescheduling and schedule repair. Most of the available schedule repair heuristics reported in the literature do not address the types of disruptions that are normally seen in the shop floor, except for machine breakdown. This paper presents a generic repair mechanism in dealing with the various types of disruptions.

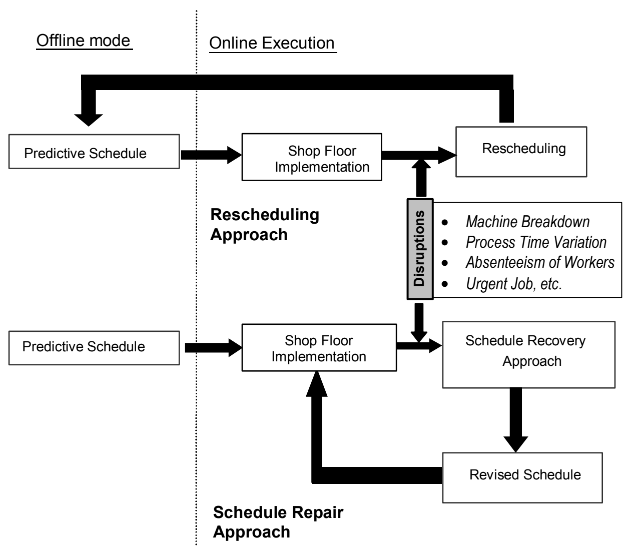

Fig. 1. Total Rescheduling vs. Schedule Repair

II. Research Motivation

The schedule repair approaches that have been reported in the recent literature are summarized in Appendix-I. Most of

Amritpal Singh Raheja is a Research Student with the Department of Mechanical Engineering, National University of Singapore. (e-mail: engp1059@nus.edu.sg).
K. Rama Bhupal Reddy is a Research Fellow with the Singapore-MIT Alliance. (e-mail: smarbrk@nus.edu.sg).
Velusamy Subramaniam is an Assistant Professor with the Department of Mechanical Engineering, National University of Singapore. He is also a Fellow with the Singapore-MIT Alliance (phone: (65) 6874-6555; fax: (65) 6779-1459; e-mail: mpesubra@nus.edu.sg).

the researchers have focused on heuristic based schedule repair techniques, as these techniques are easy to implement. Among these, Right Shift Rescheduling (RSR) and Affected Operation Rescheduling (AOR) are popular. In RSR, repair is performed by globally shifting the schedule of the remaining operations forward in time (to the right) by the amount of the disruption time [2, 3]. When a disruption occurs, the operations on all the machines after the point of disruption are incremented (right shifted). The RSR process is quite similar to manual schedule repair and is very simple to model and implement. In this repair process, a high deviation from the original schedule is reported [3]. On the other hand, AOR reschedules or repairs only those operations that are directly or indirectly affected by the disruptions [2]. The AOR heuristic minimizes both the increase in the makespan and the deviation from the initial schedule, thus making the repaired schedule both efficient and stable.

RSR and AOR schedule repair heuristics have been developed for repairing machine breakdowns. In addition to machine breakdowns, the job shop experiences a wide variety of disruptions as listed in Table I. For example, if an urgent job has to be incorporated in the schedule, a different repair strategy other than that for repairing simple machine breakdown will be required. These disruptions are complex and needs a systematic repair. Though AOR has been reported in the literature to exhibit excellent schedule repair characteristics, it is not capable of repairing disruptions other than a machine breakdown [2]. In this context, a modification to the AOR heuristic is proposed such that it would be capable of repairing most of the typical disruptions seen in a job shop.

TABLE I
DISRUPTIONS ON THE SHOP FLOOR

S/No	Disruption	Ref.
1.	Machine Breakdown	2, 5
2.	Maintenance of Machine	2
3.	Absenteeism	6, 7
4.	Tool Breakdown	7
5.	Process Time variation	6, 16
6.	Delay in transport using material handling system	4
7.	Variation in performance of machine	6, 16
8.	Tool Wear	7
9.	Variation of Set-up times	11
10.	Arrival of a new Job order	6, 16
11.	Rework	6
12.	Rejection	11, 16
13.	Unavailability of raw material	11
14.	Urgent Job	7, 16
15.	Change of priority	6, 16
16.	Cancellation of Order	6, 16
17.	Outsourcing	6

Schedule repair affects the assignment of operators, delivery of raw material, material-handling devices, subsequent processing of jobs, etc. This reassignment of operators and jobs to a different resource requires the cancellation of expensive setups and orders, and leads to *Shop floor nervousness* [1]. Alternatively, the existing schedule can be modified to adapt to the changes in the production environment by repairing the schedule. A repaired schedule will lead to a slight degradation in the quality of the schedule. It is less deviated from the old schedule, and results in minimal shop floor nervousness.

III. THE MAOR HEURISTIC

In order to handle the complex disruptions that are spread over the span of the schedule, it is necessary to analyze them carefully. Depending on the requirements, the repair heuristics can be modified to solve most of these typical job shop disruptions. A careful analysis of the disruptions reveals that the seemingly complicated repair of these disruptions can be broken down into a few simple basic steps. In Appendix-II, the general repair steps (for each of the disruption listed in Table I) are detailed. The basic repair mechanism needed to repair each of these disruptions is also listed.

A. The mAOR Mechanism

The mAOR heuristic adopts the AOR heuristic as its basis to perform the generic repair actions, which may be used singularly or in combination to repair a disruption. The procedure used by mAOR to perform each of the generic repair actions is summarized in Table II.

TABLE II
THE MAOR BASED GENERIC REPAIR MECHANISM

Repair Action	Repair Procedure
Insert idle time	Record the duration of idle time. Insert idle time in the schedule at desired location. Repair the schedule using AOR heuristic.
Insert adjustment time	Record the new End time of the deviated operation. Calculate the difference of new and original End Times. Insert an adjustment time equivalent to this time difference on the affected machine. Repair the schedule using the AOR heuristic.
Insert Operation	Record the time when operation has to be inserted and note the operation's process time. Insert a time period equal to operation's process time at required position in the schedule, on assigned machine. Repair the schedule using AOR heuristic.
Delete Operation	It involves simple deletion of the operation or removing extra time from the schedule. The original schedule is not disturbed. Delete operation does not involve the use of the AOR heuristic.

The mAOR heuristic repairs the complicated disruptions by successively performing the sequence of generic repair actions. In order to illustrate the generic nature of the repair process and to demonstrate the capability of the mAOR heuristic, it has been applied to the following types of disruptions:

- Machine Breakdown
- Process time variation
- Urgent Job

1) Machine Breakdown: Fig.2 illustrates the repair of a schedule, when a machine breakdown is encountered. The repair involves a generic repair action, namely, *Insert Idle Time*. In order to carry out the repair, the time of breakdown and the estimated duration of the breakdown are recorded. At the point of breakdown, an idle time equivalent to the time of breakdown is thereby inserted into the schedule and the operation on the machine is put on hold until the machine repair is completed.

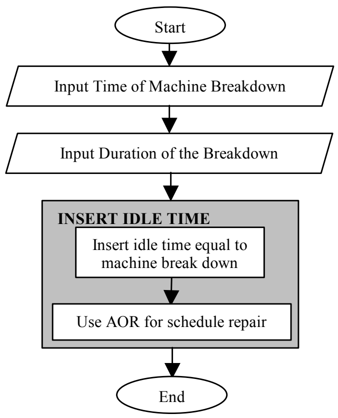

Fig. 2. Machine Breakdown

2) Process time variation: Process time variation is defined in Appendix-II as a disruption that involves a change in the End time of operations. If the processing time of an operation is reduced, the operation is deleted from the schedule and reinserted at the same starting time, but with a new end time using the *Insert operation* generic repair action. Alternatively, if there is an increase in the process time, an adjustment time equivalent to the extra time needed to complete the operation is inserted in the schedule, using *Insert Adjustment Time,* as shown in Fig.3.

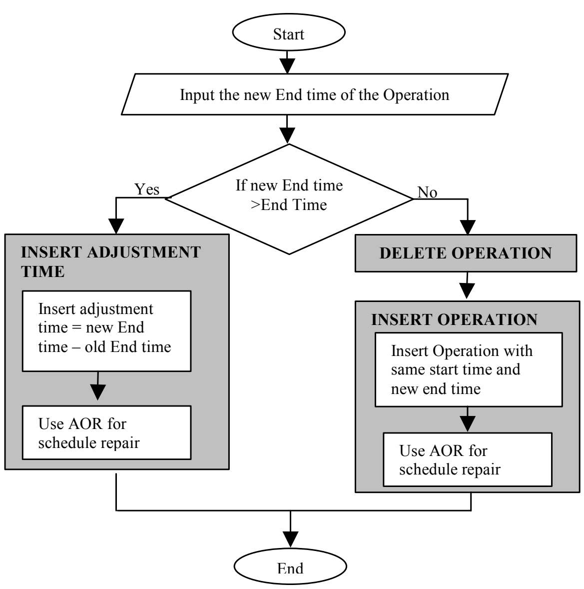

Fig. 3. Process Time Variation

3) Urgent Job: An urgent job is defined as one that is already scheduled, but it needs to be completed earlier than planned due to changes in priority or revision of due dates. Accommodating an urgent job in a schedule involves deleting the original job operations and reinserting them at the new time locations in the schedule. The repair consists of a combination of generic repair actions, namely the *Deletion* of operations followed by a series of *Insert Operations*, as illustrated in Fig.4.

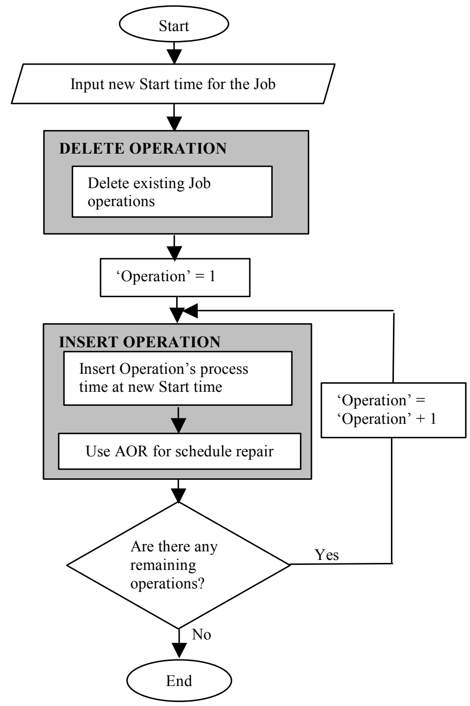

Fig. 4. Urgent Job

B. *Performance Measures*

The objective of any repair heuristic is to minimize the deviations in the schedule. In addition, an important feature of the repair is to preserve the technological and precedence constraints of the operations. Two performance measures are considered, namely Efficiency and Stability. These measures are adapted from those reported in the literature [2].

1) Efficiency: The measure of efficiency indicates the effectiveness of the repair in the schedule. It is defined as the percentage change in makespan of the repaired schedule as compared to the original schedule. Efficiency is expressed as:

$$\eta = \left\{ 1 - \frac{(M_{new} - M_o)}{M_o} \right\} \times 100 \qquad (1)$$

where,

η = Efficiency

M_{new} = Makespan of the repaired schedule

M_o = Makespan of the original schedule.

The makespan of the repaired schedule is always more than or equal to that of the original schedule. The better repair process is one in which there is a minimum or no increase in the makespan after the disruption is incorporated in the repaired schedule.

2) Stability: The stability of a schedule is measured in terms of deviations of the starting times of the job operations from the original schedule. A schedule will be stable if it deviates minimally from the original schedule. The deviation in the starting times is computed as the absolute sum of difference in starting times of the job operations between the initial and repaired schedules. It is then normalized as a ratio of total number of operations in the schedule. Stability is a cost function and is considered high if the normalized deviation is low. In other words, a better repair heuristic minimizes the normalized deviation, which is expressed as:

$$\xi = \frac{\sum_{j=1}^{k} \sum_{i=1}^{p_j} \left| \left(S_{ji}^{*} - S_{ji} \right) \right|}{\sum_{j=1}^{k} p_j} \quad (2)$$

where,
ξ = Normalized deviation.
p_j = number of operations of job j.
k = number of jobs.
S_{ji}^{*} = Starting time of i^{th} operation of job j in repaired schedule.
S_{ji} = Starting time of i^{th} operation of job j in original schedule.

IV. Experimentation

In this paper, the effects of single disruption(s) are studied. Initial schedules are generated using a discrete event simulator that handles *m* machines and *n* jobs. The values of the initial schedule parameters pertaining to the experimental study are specified in Table III. The static schedules generated from the scheduler are then subjected to disruptions of various dimensions chosen for the study. The original schedule is then repaired using the RSR and mAOR repair heuristics. The resulting repaired schedule is then evaluated using the performance measures of stability and efficiency.

TABLE III
Initial Schedule Parameters

Parameters	Values
Processing Time of a Operation	Uniformly distributed [1,50]
Number of operations per job	Uniformly distributed [2,10]
Number of Job Types	20
Number of Machines in Job Shop	6

The following dimensions have been undertaken for the study:

Size of Disruption : This refers to the average duration for which the schedule is subjected to disruptions, such as machine breakdown. In the simulation study, this disruption is expressed as a percentage of the initial schedule's makespan.

Incidence of the disruption : This refers to the time of occurrence of the disruption, which can occur either early or late in the schedule.

Size of Schedule : This refers to the size of the scheduling problem, and is expressed as the approximate number of job operations present in the initial schedule.

The possible combinations of these chosen dimensions are prepared for experimentation. These combinations are aimed to characterize the size (small and big) and occurrence (early and late) of different disruption events over the length of schedule. Details are presented in Tables IV and V.

TABLE IV
Levels of Experiment

Basic repair		Level 1	Level 2
Size of Disruption	Machine Breakdown	Small 1-3% of Makespan	Big 6-8% of Makespan
	Process Time Variation	Small 1-3% of Makespan	Big 6-8% of Makespan
	Urgent Job	Proc Time about 1-3% of Make Span	Proc Time about 6-8% of Make Span
Incidence of Disruptions		Early 5-40% of Makespan	Late 60-90% of Makespan
Schedule Size		Small 100 Operations (about 20 jobs)	Big 400-500 Operations (about 80 jobs)

TABLE V
The Experimental Combinations

Expt. No.	Size of Schedule	Incidence of Disruption	Size of Disruption
1.	Level 1 (Small)	Level 1 (Early)	Level 1 (Small)
2.	Level 1 (Small)	Level 1 (Early)	Level 2 (Big)
3.	Level 1 (Small)	Level 2 (Late)	Level 1 (Small)
4.	Level 1 (Small)	Level 2 (Late)	Level 2 (Big)
5.	Level 2 (Big)	Level 1 (Early)	Level 1 (Small)
6.	Level 2 (Big)	Level 1 (Early)	Level 2 (Big)
7.	Level 2 (Big)	Level 2 (Late)	Level 1 (Small)
8.	Level 2 (Big)	Level 2 (Late)	Level 2 (Big)

V. Results and Discussions

The effectiveness of the RSR and mAOR heuristics is studied in terms of efficiency and stability. The results are presented graphically in Figs. 5 to 7, in which the average of the performance measures and the corresponding standard deviation of simulation runs are plotted.

A. Machine Breakdown

The results for machine breakdown indicate that the mAOR approach yields significantly better performance than the RSR heuristic (Fig. 5), in terms of efficiency and stability. The duration of machine breakdown (size of disruption) has a significant effect on the repair. If the duration is small (irrespective of occurring early or late in the schedule), it is accommodated with relative ease within the schedule and the efficiency of repair is about 99 ± 1% (Expts. 1, 3, 5 and 7) versus 97.4 ± 0.8% for the efficiency achieved using the RSR heuristic. The stability of the repair is also significantly better (normalized deviation is lower) by using mAOR compared to RSR. If the machine breakdown occurs for a longer duration and occurs early in the schedule (as in Expts. 2 and 6), higher efficiency (96 ± 1.9%) is achieved using mAOR as compared to RSR (92 ± 0.9%). The stability of the mAOR-based repair is also better by approximately three times (Expts. 2 and 6). If a bigger disruption is encountered in the later half of the schedule (Expts. 4 and 8), the efficiency is lower using mAOR (95.6 ± 1.7%) as it is more difficult to accommodate a bigger disruption at a later stage and therefore an increase in the makespan is inevitable. For the same conditions using RSR, the efficiency is 92.8±0.3% and is inferior to mAOR. In addition, the stability of the schedule is also better with mAOR, compared to RSR. It is also interesting to note that the size of schedule has no major impact on the performance of the schedule repair heuristics.

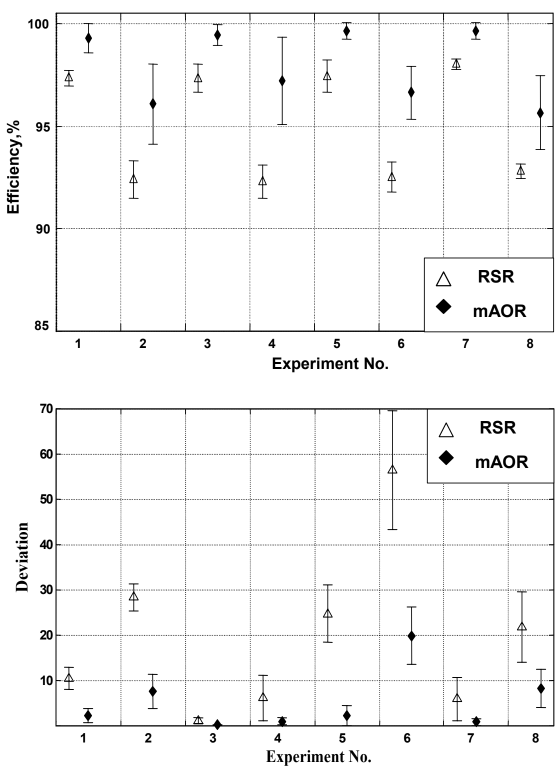

Fig. 5. Efficiency and Deviation of repair (Machine Breakdown)

B. Process Time Variation

The results (Fig. 6) show that the mAOR heuristic gives better results over the RSR heuristic in all the experiments for process time variation. Both the performance measures namely, efficiency and stability confirm the mAOR heuristic's advantage in repairing this type of disruption. The duration of process time variation corresponds to the size of the disruption. If the process time variation is small, it is easily accommodated in the schedule, irrespective of the disruption occurring early or late in the schedule (Expts. 1, 3, 5 and 7). In case of a bigger process time variation at an earlier time in the schedule (Expts. 2 and 6), higher efficiency is achieved using mAOR and the stability is also significantly better. In addition, if a similar process time variation is introduced at the later half of the schedule (Expts. 4 and 8), the efficiency is relatively lower using mAOR as it is more difficult to accommodate a bigger disruption at a later stage.

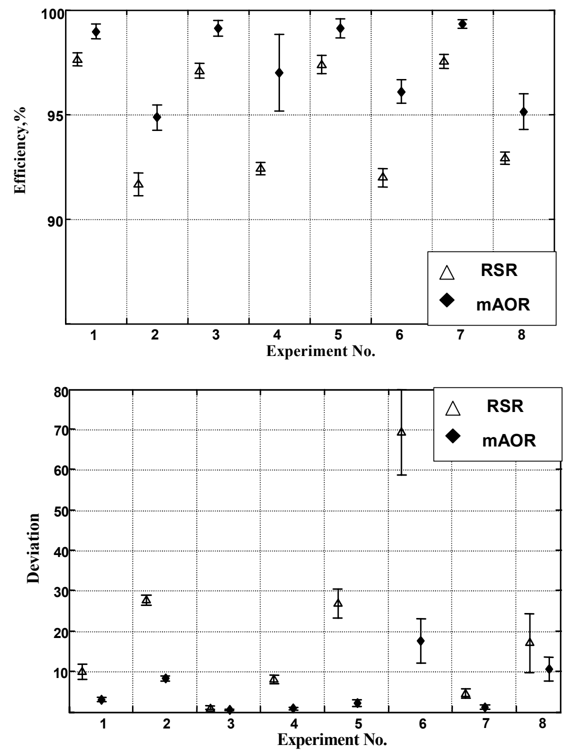

Fig. 6. Efficiency and Deviation of repair (Process Time Variation)

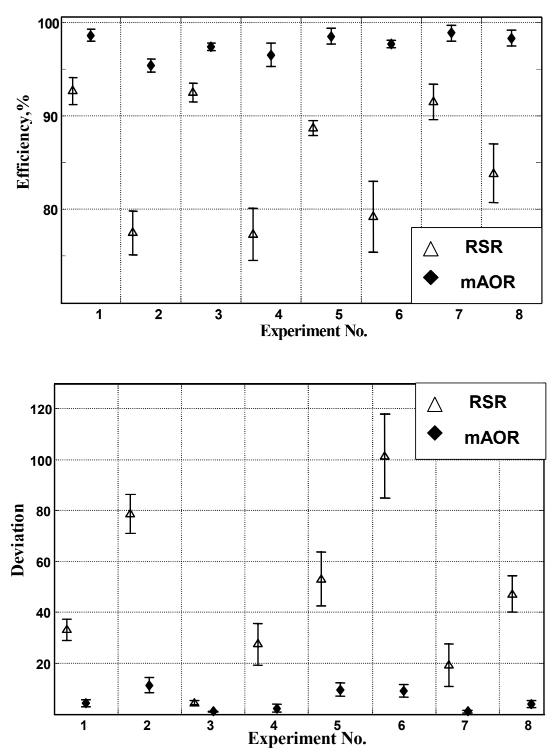

Fig. 7. Efficiency and Deviation of repair (Urgent Job)

C. *Urgent Job*

The repair performances of mAOR and RSR, in dealing with an urgent job are presented in Fig. 7. An urgent job is accommodated in the schedule with high efficiency and stability using mAOR as compared to RSR. The performance of RSR as a repair heuristic is poor, because multiple operations have to be accommodated. The RSR heuristic right- shifts the entire schedule to insert each of the job operations resulting in lower efficiencies (as low as 77.3 ± 2% in Expts. 2 and 4). The higher deviation values (101.4± 16 in Expt. 6) also indicate that the stability of the repaired schedule is poor

VI. CONCLUSIONS

The results of the simulation study show that mAOR is successful in repairing the typical job shop disruptions. In addition, the simulation study clearly indicates that mAOR has a significant edge over RSR in repairing these disruptions under varied shop floor conditions. In this paper, complicated repair processes for various disruptions have been shown to comprise of only four basic repair actions. This decomposition in the repair process allows one to address most of the complicated schedule disruptions. The framework reported in this paper can be easily extended to other disruptions or can be used as a testbed for future heuristics.

REFERENCES

[1] J. Dorn, "Case based reactive scheduling", Chapter 4 in *Artificial Intelligence in Reactive scheduling*, Chapman and Hall (UK), 1994, pp. 32-50.

[2] R. J. Abumaizar and J.A Svestka, "Rescheduling job shops under random disruptions", *International Journal of Production Research*, 35 (7), 1997, pp. 2065-2082.

[3] P. Brandimarte, M. Rigodanza and L. Roero, "Conceptual modeling of an object oriented scheduling architecture based on the shifting bottleneck procedure", *IIE Transactions*, 32 (10), 2000, pp. 921-929.

[4] G. Hasle and S. F. Smith, "Directing an opportunist scheduler: an empirical investigation on reactive scenario." Chapter 1 in *Artificial Intelligence in Reactive scheduling*, Chapman and Hall (UK), 1994, pp.1-11.

[5] H. Henseler, "From reactive to active scheduling by using multi agents." Chapter 2 in *Artificial Intelligence in Reactive scheduling*, Chapman and Hall (UK), 1994, pp. 12-18.

[6] E. Szelke and G. Markus, "A black board based perspective of reactive scheduling," Chapter 6 in *Artificial Intelligence in Reactive scheduling*, Chapman and Hall (UK), 1994, pp. 60-77.

[7] J. Dorn, R. Kerr and G. Thalhammer, "Reactive Scheduling in a fuzzy-temporal framework. Knowledge based reactive scheduling", *IFIP Transactions B (Applications in Technology)*, B-15, 1994, pp. 39-54.

[8] G. Schmidt, "How to apply fuzzy logic to reactive production scheduling. Knowledge based reactive scheduling", *IFIP Transactions B (Applications in Technology)*, B-15, 1994, pp. 57-66.

[9] E. Szelke and G. Markus, "A learning reactive scheduler using CBR/L." *Computers in Industry*, 33 (1), 1997, pp. 31-46.

[10] K. Miyashita, "Case based knowledge acquisition for schedule optimization." *Artificial Intelligence in Engineering*, 9 (4), 1995, pp.277-287.

[11] J.E. Spargg, G. Fozzard and D. J. Tyler, "Constraint based reactive rescheduling in a stochastic environment". *Proceedings, Recent Advances in AI Planning. 4th ECP'97*, 1997, pp. 403-413.

[12] B. J. Garner and G. J. Ridley, "Application of neural network process models in reactive scheduling. Knowledge based reactive scheduling", *IFIP Transactions B (Applications in Technology)*, B-15, 1994, pp.19-28.

[13] G. A. Rovithakis, S. E. Perrakis and M. A. Christodoulou, "Application of a neural network scheduler on a real manufacturing system," *IEEE Transactions on Control Systems Technology*, 9 (2), 2001, pp. 261-270.

[14] J. G. Qi, G. R. Burns and D. K. Harrison, "The application of the parallel multi population genetic algorithms to dynamic job shop scheduling," *The International Journal of Advanced Manufacturing Technology*, 16, 2000, pp. 609-615.

[15] S. B. Gershwin, "Manufacturing Systems Engineering". Englewood Cliffs, New Jersey: PTR Prentice Hall, 1994.

[16] H. Henseler, "Reaktion: a system for event independent reactive scheduling", Chapter 3 in *Artificial Intelligence in Reactive scheduling*, Chapman and Hall (UK), pp. 19-31, 1994.

[17] V. Subramaniam, G. K. Lee, T. Ramesh, G.S. Hong and Y. S. Wong, "Machine selection rules in a Dynamic Job Shop", *The International Journal of Advanced Manufacturing Technology*, 16, pp. 902-908, 2000.

APPENDIX – I

Summary of Schedule Repair

Schedule Recovery Method		Advantages	Disadvantages	Performance Measures	Application area	References
Heuristics based approaches (RSR and AOR)		Simple to implement. AOR has better schedule quality than RSR.	Limited disruptions can be handled. Schedule quality is not recuperated after repair.	Makespan and deviation from the original schedule.	Job shop that is usually stable with minor disruptions at prolonged intervals. The technological constraints and processing times are predetermined and fixed.	2, 3
Multi agents in Distributed Artificial Intelligence		Complete automated approach. Module for repair, refinement and rescheduling. Responsiveness of system is good. Multithreaded operations are possible.	Coordination between the agents is difficult to achieve. Better integration between human and automated agents is difficult to achieve.	Computation Time Reactive and Quality measures.	Dynamic job shop with uncertainties and random disruptions.	4-6
Knowledge Based Scheduling and Artificial Intelligence approaches	**Case Based Reasoning (CBR)**	Well suited to domain specific problems. Continuous learning from past cases. Multiple disruptions can be modeled and addressed.	Extensive search through the database consumes time. An extensive experience database is essential.	Schedule quality and Reactive efficiency in terms of deviation from the original schedule.	Job shops where scheduling experience is available in form of expert's advice or case database.	1, 9
	Constraint based Scheduling	Human interaction and supervision is better. Timely response is possible in stipulated time frame. Performs better than CBR as it includes both knowledge base and constraint satisfaction modules.	Real time approach needs further refinement. Multiple agent architecture is needed for multi-threaded operations for random disruptions.	CPU time (execution responsiveness), Schedule Quality and Repaired weightage tardiness.	Dynamic job shops with multiple disruptions.	10, 11
	Fuzzy Logic	Complete scan of the schedule for constraint violation after every repair. Random processing times can be used for disruptions. Response is fast as the same module is used for schedule generation and repair.	The knowledge of the domain has to be built into the algorithm. Learning and growth of the algorithm is not possible.	CPU time (execution responsiveness), Schedule Quality and Repaired weightage tardiness.	Job shops with variability in processing time and large number of constraints to be adhered to either fully or partially.	7, 8
	Neural Networks	Response time is very fast for trained neural net. Predictions are extrapolated from the past experience and are reliable.	Carefully prepared training sets are needed for accurate prediction. Extensive knowledge base and expert advice has to be formulated in the form of a knowledge base.	CPU time (execution responsiveness), Schedule Quality and Repaired weightage tardiness.	More applicable in job shops with a continuous flow and repetitiveness in the type of disturbances.	12-14

Appendix – II

Analysis of Disruptions

S/No	Disruption	Effect	General Repair Procedure	Basic Repair Mechanism
1	Machine Breakdown	Machine unavailable for a period.	Record the time of disruption and the estimated time of repair. Introduce an idle time period equal to the breakdown time and Repair the schedule.	Insert an idle time
2	Maintenance of Machine	Machine unavailable for a period.	Process is similar as above; only the maintenance time replaces machine breakdown time.	Same as (1).
3	Absenteeism	If operator is unavailable temporarily, this disruption may be treated as an idle machine. If the operator is unavailable for a longer period, a substitute operator or total rescheduling is required as the disruption to the schedule is critical and repair will not be sufficient.	The machine is idle for the duration of time the operator is absent. Introduce an idle time period equal to the idle time in order to keep the machine free and repair the schedule.	Same as (1).
4	Tool Breakdown	Machine unavailable for a period.	The machine is idle for the time needed for tool change. Introduce a time slot equal to the tool change time in order to halt the machine and repair the schedule.	Same as (1).
5	Process Time variation	Change in End Time.	Identify the operation. Record the increment in the end time. Insert a time period equal to increment in process time and repair the schedule. In case, the process time is decreased, delete the operation from the schedule and insert it again with same start time and new end time.	Insert an adjustment time (in case of increase of process time) and use Deletion and Insert Operation if the process time is reduced.
6	Delay in transport using material handling system	Failure to deliver the parts to the machine in time leads to increase in end times.	Identify the operation affected due to delay. Record the increment in its end time. Insert a time period equal to the additional time needed to process the operation and repair the schedule.	Insert an adjustment time.
7	Variation in performance of machine	This may lead to changes in the process times and subsequent changes in the end times.	Same as (5).	Same as (5).
8	Tool Wear	Requires longer process times and subsequently leads to changes in end times	Same as (6).	Same as (6).
9	Set-up times variation	This will lead to change in start/end time of Jobs.	If start time is late, it will lead to delay in end time of job operation and if start time is earlier, job operation will end earlier, therefore repair process is similar as (5)	Same as (5).
10	Rework	Some operation of the job is required to be redone.	Record the time of rework. Insert a time period equal to the operation time on the machine. Repair the schedule using a repair heuristic.	Insert operation.
11	Arrival of a new Job order	A new job arrives and has to be immediately inserted in the schedule.	Record the arrival time Insert a time period equivalent to the operation on assigned machine and repair the schedule using a repair heuristic. Repeat the process for other operations.	Insert operations iteratively.
12	Rejection	The entire job has to be redone.	Record the time of re-arrival Insert a time period equivalent to the operation on assigned machine and repair the schedule using a repair heuristic. Repeat the process for other operations.	Same as (11).
13	Unavailability of raw material	Job operation cannot be processed as planned	Record the estimated time of raw material availability. Delete the operation from current time. Insert a time period equal to the operation time on the machine. Repair the schedule using a repair heuristic.	Delete the scheduled operation. Insert new operation.
14	Urgent Job	A job needs to be done urgently because of due date revision or sudden demand. It is shifted up the time ladder.	Record the estimated new time for the job. Delete the job from the existing position in the schedule. Insert a time period equivalent to the new operation's processing time and repair schedule. Repeat the process for other operations.	Delete the Job. Insert operations iteratively.
15	Change of priority	A job is required earlier or later than its scheduled time.	Same as (14).	Same as (14).
16	Cancellation of Order	A job is no longer required to be produced.	Identify the job that is cancelled. Remove it from the Schedule.	Delete the Job and its operations.
17	Outsourcing	Job is outsourced and production is no longer needed.	Identify the job that is outsourced Remove it from the Schedule.	Same as (16).