International Journal of Current Engineering and Technology

E-ISSN 2277 – 4106, P-ISSN 2347 – 5161
Available at http://inpressco.com/category/ijcet

Research Article

Optimizing Dynamic Flexible Job Shop Scheduling Problem Based on Genetic Algorithm

Abd Elrahman Elgendy[#*], Mohammed Hussein[^], Abdelmoty Elhakeem[#]

[#]Department of Mechanical Engineering, Benha University, Egypt
[^]Department of Mechanical Engineering, the British University in Egypt

Accepted 03 March 2017, Available online 09 March 2017, **Vol.7, No.2 (April 2017)**

Abstract

Scheduling the flexible job shop in the dynamic environment, in which arriving new job, the breakdown of machines and processing time variation are possible, is the most complex problem in the manufacturing system until now. A genetic algorithm (GA) was developed to deal with the problem related to flexible job shop scheduling problem represented in routing and sequencing the operations, besides the problem related to dynamic environment represented in appearing new events such as new job arrival and processing time variation. The algorithm incorporated the traditional procedures of GA with a repair strategy in order to optimize the makespan of dynamic flexible job shop scheduling problem (DFJSSP). The results indicate that the proposed algorithm is effective for solving DFJSSP.

Keywords: *Flexible Job Shop Scheduling Problem, Dynamic Job Shop Scheduling Problem, Genetic Algorithm, Rescheduling strategy*

1. Introduction

The manufacturing companies work on streamlining their production processes, meeting their customers' requirements and accommodate the high variety of demand. Therefore, there was a big need for a job shop to accommodate these demands. Job shop scheduling problem is one of the most complex and combinatorial optimization problems in manufacturing systems, where the job shop problem consists of N jobs and each job has set of operations that must be processed in a predefined order on the available machines. Therefore job shop is mathematically well known as nondeterministic polynomial hard (NP-hard) combinational optimization problem Pinedo (2015). However, the ability to operate an operation on several machines in the real-world job shop turn it to be more general, which is known as flexible job shop scheduling problem (FJSSP). The FJSSP is considered the generalization of JSSP, where it is allowed for operations to be processed on any one from the specified machines. Hence, it is important for any flexible job shop to treat with two main problems sequencing problem and assignment problem. Therefore, FJSSP is strongly NP-hard too Demir and İşleyen (2013).

Most of the real-world environments in manufacturing systems work dynamically due to the unexpected disturbances that may arise either from manufacturing resources or the operated jobs, the source of these disruptions called real-time events such as (arriving urgent jobs, changes in job processing time, machines breakdown, etc.) MacCarthy and Liu (1993). Emergence any among of these events in the manufacturing environment can turn the status of the system.

These events able to change the feasible schedule to infeasible and cause deterioration in the schedule's performance. Therefore, it is necessary to adapt a rescheduling strategy in order to overcome the impact of the disruption. This part of scheduling is known as dynamic scheduling. Rescheduling is the process of updating an existing production schedule in order to respond quickly to disruptions or other changes with preserving the desired performance and schedule stability as possible. The most important issue in applying real-time scheduling is how to deal with appearing a real-time event during the execution of a given schedule on the shop floor Ouelhadj and Petrovic (2009).The remainder of this paper is organized as follows. Section 2 summarize some of the works that were presented for DFJSSP. Then, the proposed GA is introduced in Section 3. Experimental test and discussion are discussed in Section 4. Whereas, section 5 shows the conclusions and future.

2. Literature review

This review will focus on the approaches that are used for optimizing dynamic job shop scheduling problems

*Corresponding author: **Abd Elrahman Elgendy**

and flexible job shop scheduling problems. Several approaches have been presented to find an optimal solution in suitable time. Exact methods such as mixed integer programming, disjunctive graph, Lagrangian relaxation, etc. were the first approaches that were applied for FJSSPs but they are considered time-consuming and more suitable for small size problem Pinedo (2015). Therefore, researchers have turned their priority towards approximation based techniques in order to achieve good quality solutions in lesser time. However, the known approximation methods such as Priority Rules, Bottleneck Heuristic, local search methods, and artificial intelligence, do not produce solutions with a guaranteed distance from the optimal solution for FJSSP.

In a dynamic flexible job shop scheduling problem (DFJSSP) where the jobs arrive over time, Nie, Gao *et al.* (2013)proposed a reactive approach in order to continuously respond to the disruption and improve the scheduling performance criteria. They combined some of the common dispatching rules with gene expression programming (GEP) to construct an effective reactive scheduling approach to assign jobs to suitable machines (routing) and prioritize jobs on machines (sequencing) in a dynamic environment. The approach was tested under different conditions for optimizing makespan, mean flow time and mean tardiness. The results indicated that the proposed approach was much better than human-made dispatching rules and the rules constructed by genetic programming. Whereas, Lin, Gen *et al.* (2012) adopted hybrid evaluation algorithm (HEA) in order to accommodate the disruptions that arise from machine breakdown or changing the due date. Besides, a set of dispatching rules that were used for generating initial solutions for machine assignment problem and operation sequence problem. Then, the HEA that combines some characteristics from particle swarm optimization (PSO) and genetic algorithm (GA) for discovering the solution space and guiding the search toward promising regions, adopted continuously repair strategy in order to treat machine breakdown problem. The results show the success of HEA to give a better solution in comparison to priority-based GA and random key-based PSO. Al-Hinai and ElMekkawy (2011) presented a predictive schedule based on a two-staged hybrid genetic algorithm to improve the stability and robustness of the schedule in a flexible job shop with considering the machine breakdown.

The proposed method was compared with a deterministic approach, where the results showed the success of predictive schedule to improve the stability in all instances.

Shen and Yao (2015) optimized a multi-objective DFJSSP based on an evolutionary algorithm (MOEA), where a predictive-reactive method was considered for improve resource utilization represented in makespan beside maintaining the stability of the schedule. The MOEA optimizes the DFJSSP on two stages; firstly, the algorithm work on generating initial predictive schedule in order to optimize makespan, tardiness and maximal machine workload, then at each rescheduling point, the algorithm is triggered in order to construct a new schedule to accommodate the new jobs with considering the stability and the efficiency of schedule. They applied tournament selection, crossover operator and mutation operator in reproduction. The presented results show the potential of the proposed approach to achieve better performance than some common dispatching rules and two other evolution algorithms. Kundakcı and Kulak (2016) developed a metaheuristic approach based on genetic algorithm for optimizing the efficiency of the dynamic job shop scheduling problem. The proposed algorithms depend on using a specific heuristic method for generating 25% of the initial solutions (population) and the other solutions were randomly generated, then using the usual procedure of GA for optimizing the initial solutions. This process is repeated when occurring any disruption. The performances of the different proposed algorithms were tested on a set of DJSSPs where they generated a set of different size DJSSPs where each problem considers different dynamic factors. The results indicate that the proposed algorithms deliver outstanding solutions for DJSSP on the hand of solution quality and CPU time.

Liu, Abdelrahman *et al.* (2007) proposed approach look likes an auction between jobs to win with a machine, where the machine agent announce an auction within the given time horizon for all available jobs. The approach is called complete multi-agent framework (CMAF). The CMAF adopted a completely reactive schedule where it uses dispatching rule to put forward the suitable job among available jobs for the available machine, except each machine has a local schedule with the ability to do a predictive decision-making during the time horizon based on mathematical programming. The experimental results show the success of CMAF to present more stable and adaptable. The approach succeeded to build schedules can accommodate different types of disruptions, with respect to dispatching rules in dynamic job shop scheduling. However, the approach unable to optimize the performance such as reactive approaches. Zhang, Gao *et al.* (2013) presented hybrid evaluation algorithm that aims to introduce a rescheduling approach to accommodate different types of disruptions in DJSSP. The proposed strategy is seeking to improve the performance of the schedule, besides keeping a balance between efficiency and stability of the schedule. Therefore, they adopted weighted multi-objective algorithm able to deal with two types of disruption, random machine breakdown and dynamic job arrival. They implemented hybrid periodic and event driven rescheduling policy to ensure the adaptability of the algorithm and also continuous processing in a dynamic environment. The results indicate the superiority of the proposed algorithm to present better performance in all cases but the makespan starts worse in high arrival rate.

3. Genetic algorithm for DFJSSP

The genetic algorithm is considered an effective optimization approaches for improving the performance of complex problem such as JSSP. The algorithm can be used to treat a wide range of problems under different conditions. One of the practical tests that proof the effectiveness of GA is the evolution of human being. The power of GA represented in the exploration process, where the search starts with a set of solutions that work in parallel to find the best solution. Among other approaches, GA has the highest possibility to reach the optimal solution in a certain time interval.

This section will show procedures of the algorithm for optimizing flexible job shop scheduling problems in dynamic environments. In order to effectively schedule operations in the job shop and adopt any disruption in the system an adaptable approach is developed based on GA. The followed procedure can be described in the following steps and shown in the flow chart.

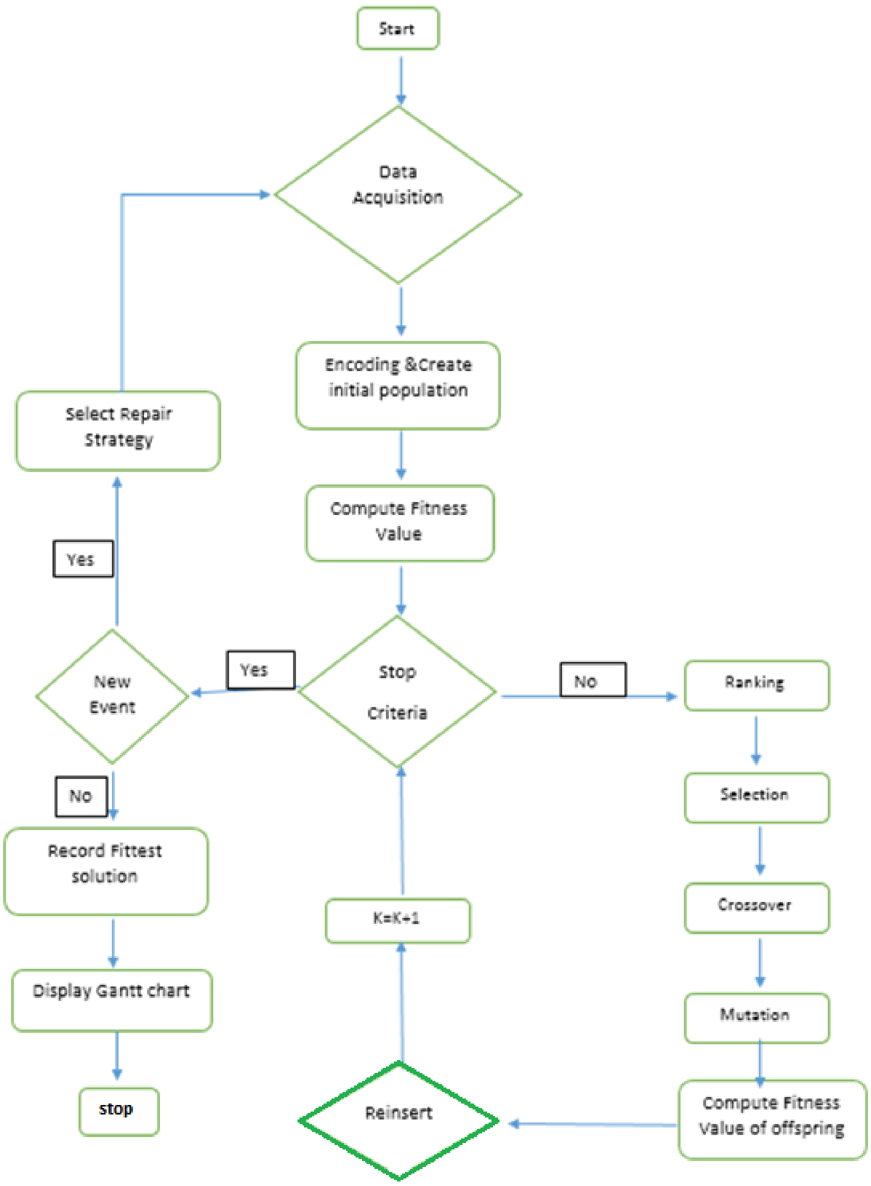

Figure 1 Genetic Algorithm Procedures

The structure of the solution algorithm can be simplified as follow:

1) Chromosomes Representation: first the operations have been assigned to machines based on shortest processing time. The algorithm adopted an operation-based representation (OBR) with considering permutation with m-repetition that was developed by Bierwirth (1995) for encoding FJSSP.
2) Initial population: The chromosomes are generated randomly based on OBR.
3) Evaluation of the population: machine utilization represented in makespan was selected measure the performance of the proposed algorithm.

4) Selection of individuals: at the first a rank-based fitness assignment was selected to arrange the population based on the fittest value. Then, stochastic universal sampling (SUS) was adapted to select the fittest chromosomes for the reproduction process Baker (1987).
5) Crossover operator: multi-point crossover based on the operation-based crossover (OBX) was used for performing permutation and maintaining the feasibility of solutions.
6) Mutation operator: random mutation was considered to escape from local optimal solution trap.
7) Stopping criteria: the algorithm stops running when getting the optimal solution or the maximum number of generation.
8) Repair strategy: adopting a repair strategy were inevitable because of the dynamic nature of job shop environment. Therefore, the algorithm adopted the strategies that was presented by
9) Fattahi and Fallahi (2010) in order to optimize the makespan and maintain the stability of the schedule at the same time.

4. Computational Results

The improved genetic algorithm (IGA) has been implemented on personal lab with core i5, 2.5 GHz, and 4G RAM. Firstly, the algorithms were applied to a set of static benchmark problems have machines between four and ten in order to state the most suitable parameters for the GA as shown in Error! Reference source not found.. The Gantt chart and the iteration solution chart that have been generated by the IGA is shown in Figure 2and Figure 3 respectively. This research test the potential of the improved genetic algorithm to optimize flexible job shop in a dynamic environment, where arriving new jobs and processing time variation are possible. Therefore, a flexible job shop problem has 8 jobs and 9 machines was generated as shown in Table 2. The values in Table 2 represent the processing time on each machine, whereas inf means the machine cannot process the corresponding operation.

Table 1 Genetic algorithm parameters

Crossover Ratio	90 %
Mutation Ratio	1 %
Number of individuals	40
Number of generations	100
Machine assignment	Shortest processing time
Selection Operator	stochastic universal sampling (SUS)
Crossover Operator	Operation based crossover (OBX)
Mutation Operator	Swap

The Gantt chart in Figure 5 and the iterative solutions in Figure 4 represent the initial optimal solution obtained after 5 runs for DFJSSP. As the dynamic environment change continuously, it is important to accommodate the actions that may arise in the system. Therefore, this research adopts partial repair strategy to accommodate these actions. Table 3 represents a sample of new jobs that arrive at the system at different times and need to be scheduled. The algorithm succeeded to present an outstanding solution for the available problem. The IGA succeeded to accommodate the new jobs with keeping the deviation from the initial schedule very small as shown in Figure 6.

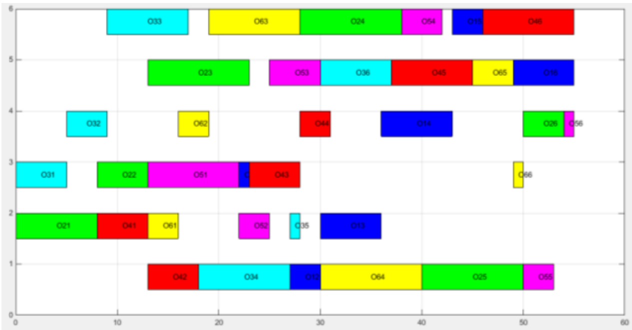

Figure 2 Gantt chart of FT06. Cmax=55

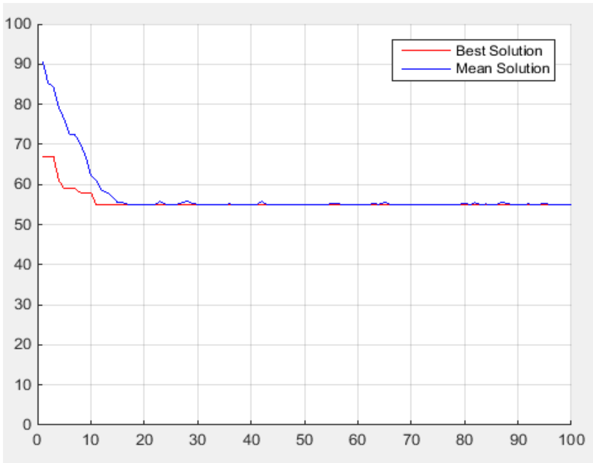

Figure 3 Iterative solutions of FT06

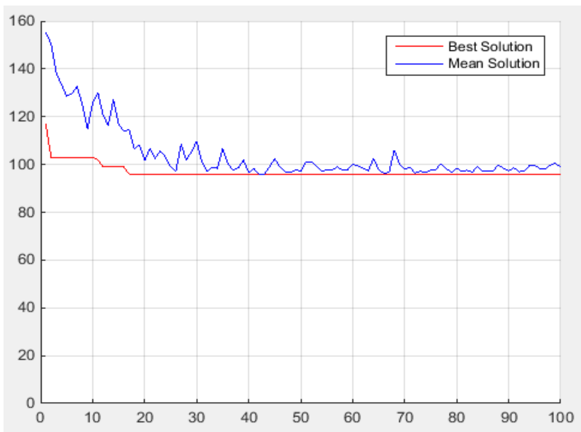

Figure 4 Iterative solutions of DFJSSP

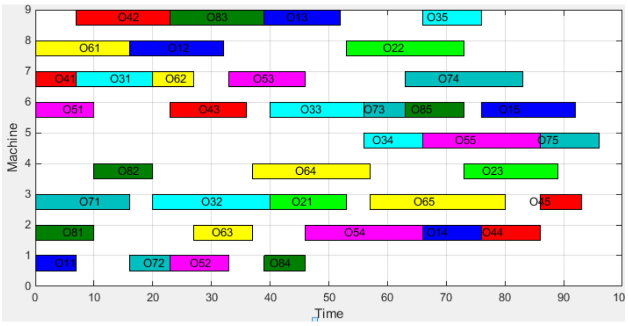

Figure 5 Gantt chart of DFJSSP, Cmax=96

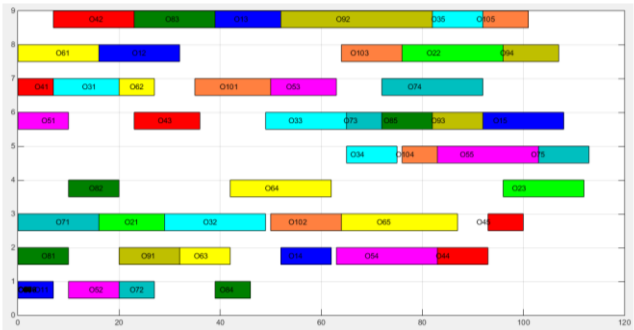

Figure 6 Repaired Gantt chart, Cmax=113

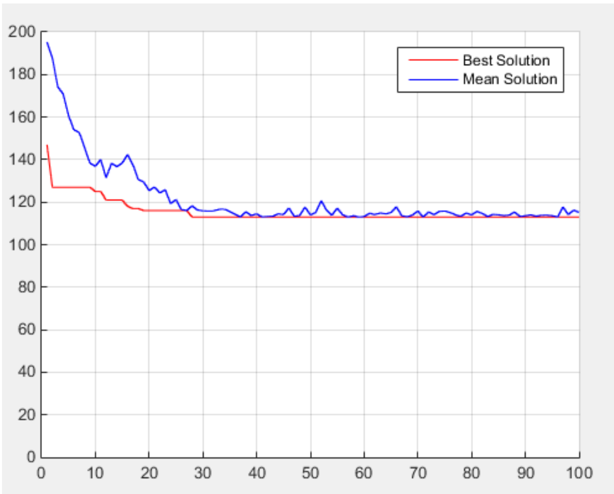

Figure 7 Iterative solutions of repaired schedule

Table 2 Problem information

Job	Operation	Machines								
		M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9
1	1-1	7	inf	13	inf	32	10	inf	26	inf
	1-2	inf	23	inf	26	inf	29	inf	16	inf
	1-3	inf	20	16	inf	32	29	inf	inf	13
	1-4	inf	10	inf	30	inf	13	inf	inf	inf
	1-5	inf	inf	inf	23	inf	16	inf	26	inf
2	2-1	32	inf	13	inf	32	inf	inf	26	inf
	2-2	29	inf	26	inf	inf	23	inf	20	inf
	2-3	inf	20	inf	16	inf	26	23	inf	inf
3	3-1	inf	inf	29	inf	20	inf	13	inf	inf
	3-2	inf	inf	20	inf	26	inf	inf	23	inf
	3-3	inf	29	inf	inf	inf	16	inf	20	inf
	3-4	inf	15	inf	inf	10	inf	30	12	inf
	3-5	inf	inf	13	inf	inf	20	inf	inf	10

4	4-1	inf	26	inf	16	inf	inf	7	13	inf
	4-2	inf	inf	20	inf	inf	inf	26	inf	16
	4-3	16	inf	inf	26	inf	13	inf	inf	inf
	4-4	inf	10	inf	inf	36	20	inf	26	inf
	4-5	inf	inf	7	inf	16	26	inf	inf	26
5	5-1	20	inf	inf	13	inf	10	inf	inf	26
	5-2	10	inf	inf	23	inf	20	inf	32	inf
	5-3	inf	inf	29	inf	inf	inf	13	inf	inf
	5-4	22	20	32	inf	inf	inf	inf	inf	inf
	5-5	inf	23	inf	inf	20	inf	inf	23	inf
6	6-1	inf	inf	inf	inf	32	26	inf	16	inf
	6-2	inf	inf	inf	10	inf	inf	7	23	inf
	6-3	inf	10	inf	inf	32	inf	inf	inf	20
	6-4	32	inf	inf	20	inf	32	inf	inf	29
	6-5	inf	inf	23	inf	26	29	inf	32	inf
7	7-1	29	inf	16	inf	23	29	inf	inf	inf
	7-2	7	inf	20	inf	26	inf	10	23	inf
	7-3	inf	inf	inf	16	20	7	inf	29	10
	7-4	inf	inf	inf	inf	inf	inf	20	30	24
	7-5	29	inf	48	16	10	inf	inf	20	inf
8	8-1	inf	10	inf	20	inf	26	16	inf	inf
	8-2	13	inf	20	10	inf	inf	inf	26	32
	8-3	inf	16	26	inf	inf	20	inf	inf	16
	8-4	7	inf	20	inf	inf	inf	20	inf	10
	8-5	15	inf	inf	16	inf	10	inf	16	29

Table 3 New jobs arrival

Arrival time	Job	Operation	Machines								
			M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9
20	9	9-1	inf	12	30	inf	inf	23	13	inf	inf
		9-2	inf	inf	inf	inf	inf	inf	inf	inf	30
		9-3	inf	11	16	inf	inf	10	inf	inf	inf
		9-4	13	inf	15	inf	23	inf	33	11	inf
35	10	10-1	inf	inf	inf	inf	inf	inf	15	inf	inf
		10-2	16	16	14	18	15	22	inf	20	21
		10-3	44	19	inf	33	24	15	25	12	14
		10-4	inf	inf	inf	inf	7	inf	15	inf	inf
		10-5	22	28	inf	inf	23	15	inf	22	9

Conclusions

This research presented an improved Genetic algorithm for optimizing resources utilization of flexible job shop scheduling problem in the dynamic environment. Experimental results indicate the ability of the improved genetic algorithm to efficiently schedule static job shop, flexible job shop problems. Besides, the potential to accommodate the new events that disturb the system effectively. The future work will concern with optimizing multi-objective flexible job shop scheduling problems in a dynamic environment and compare the results with other approaches.

References

Al-Hinai, N. and T. ElMekkawy (2011). Robust and stable flexible job shop scheduling with random machine breakdowns using a hybrid genetic algorithm. International Journal of Production Economics 132(2): 279-291.

Baker, J. E. (1987). Reducing Bias and Inefficiency in the Selection Algorithm. Proceedings of the Second International Conference on Genetic Algorithms and their Application. Hillsdale, New Jersey: L. Erlbaum Associates: 14–21.

Bierwirth, C. (1995). A generalized permutation approach to job shop scheduling with genetic algorithms. Operations-Research-Spektrum 17(2-3): 87-92.

Demir, Y. and S. K. İşleyen (2013). Evaluation of mathematical models for flexible job-shop scheduling problems. Applied Mathematical Modelling 37(3): 977-988.

Fattahi, P. and A. Fallahi (2010). Dynamic scheduling in flexible job shop systems by considering simultaneously efficiency and stability. CIRP Journal of Manufacturing Science and Technology 2(2): 114-123.

Kundakcı, N. and O. Kulak (2016). Hybrid genetic algorithms for minimizing makespan in dynamic job shop scheduling problem. Computers & Industrial Engineering 96: 31-51.

Lin, L., *et al.* (2012). A hybrid EA for reactive flexible job-shop scheduling. Procedia Computer Science 12: 110-115.

Liu, N., *et al.* (2007). A complete multiagent framework for robust and adaptable dynamic job shop scheduling. Ieee Transactions On Systems Man And Cybernetics Part C Applications And Reviews 37(5): 904.

MacCarthy, B. and J. Liu (1993). Addressing the gap in scheduling research: a review of optimization and heuristic methods in production scheduling. The International Journal of Production Research 31(1): 59-79.

Nie, L., *et al.* (2013). A GEP-based reactive scheduling policies constructing approach for dynamic flexible job shop scheduling problem with job release dates. Journal of Intelligent Manufacturing 24(4): 763-774.

Ouelhadj, D. and S. Petrovic (2009). A survey of dynamic scheduling in manufacturing systems. Journal of Scheduling 12(4): 417-431.

Pinedo, M. (2015). Scheduling, Springer.

Shen, X.-N. and X. Yao (2015). Mathematical modeling and multi-objective evolutionary algorithms applied to dynamic flexible job shop scheduling problems. Information Sciences 298: 198-224.

Zhang, L., *et al.* (2013). A hybrid genetic algorithm and tabu search for a multi-objective dynamic job shop scheduling problem. International journal of production research 51(12): 3516-3531.