Journal of Manufacturing Engineering, 2008, Vol.3, Issue.2

A SCATTER SEARCH APPROACH FOR OPTIMIZATION OF MAKESPAN AND TOTAL FLOW TIME IN FLOW SHOP SCHEDULING

M. Saravanan [1], A. Noorul Haq [2]

[1] Professor and Head, Department of Mechanical Engineering, RVS College of Engineering & Technology, Dindigul, Tamil Nadu, India-624005.

[2] Professor,Department of Production Engineering, National Institute of Technology, Tiruchirappalli, Tamil Nadu, India-620015.

ABSTRACT

In many real-world scheduling problems several criteria must be considered simultaneously for evaluating the quality of the solution or schedule. Traditionally, these problems have been tackled as single-objective optimization problems after combining the multiple criteria into a single scalar value. A number of multi objective meta-heuristics have been proposed in recent years to obtain sets of compromised solutions for multi objective optimization problems. Most of these techniques have been successfully tested in both benchmark and real-world multi objective problems. This paper proposes a new heuristic i.e, Scatter search for minimizing the make span and total flow time to a flow shop problem. Scatter Search (SS) is applied to this problem as it is able to provide a wide exploration of the search space through intensification and diversification. In addition it has a unifying principle for joining solutions and they exploit adaptive memory principle to avoid generating or incorporating duplicate solutions at various stages of the problem. This paper initially addresses the general method of finding the make span and the total flow time. Scatter search algorithm is presented and applied to the given manufacturing environment and the corresponding steps are presented. The proposed scatter search algorithm is applied to a number of multi machine and multi job combination using proper coding and the results of the experimental investigation are presented.

Keywords: *Flow shop, Scheduling, Meta- heuristic, Scatter Search, makespan, flow time, Optimization.*

1. INTRODUCTION

In a scheduling problem, using *m* machines must finish a total of *n* jobs and each job has exactly *m* operations, each of which must be proceeded in a different machine. Thus each job has to pass through each machine in a particular order. The order of machines needed to complete a job is the same for all the jobs. The objective of the scheduling problem now, is to determine a sequence on each machine that satisfies the above constraints and minimizing the objective function. In this paper, the permutation flowshop problem has been attempted with a novel evolutionary technique called Scatter Search (SS). This algorithm incorporates procedures based on different strategies, such as diversification, local search, Tabu search or path relinking. In common with other evolutionary methods, Scatter Search operates with a population of solutions, rather than with a single solution at a time, and employs procedures for combining these solutions to create new ones. Scatter search, in contrast with other evolutionary procedures such as genetic algorithms, provides an unifying principle for joining solutions based on generalized path constructions and by utilizing strategic designs where other approaches resort to randomization. Additional advantages are provided by intensification and diversification mechanisms that exploit adaptive memory, together with processes to avoid generating or incorporating duplicate solutions at various stages and drawing on foundations that link scatter search and path relinking to tabu search, Glover et. al. [1].

Flow shop Scheduling.

Classic formulation and main constraints of the flow shop problem are known as follows:

- each machine can process only one job at a time,
- the machines are continuously available,
- two operations of the same job cannot be processed at the same time,
- preemption is not allowed,
- processing times are known in advance,
- transportation time between two machines is zero,
- the sequence of operations for each job is predefined.

In the case of flow shop problem, each job undergoes the same machines sequence and the solution of the problem can be represented as a

Corresponding author: *sarandgl2k@yahoo.co.in*

permutation of all jobs to be processed. In other words, the flow shop problem is defined as a unidirectional flow of work with a variety of *n* jobs, all jobs being processed sequentially in *m* machines in the same order. The basic quality criterion, most often used for all scheduling problems, is the minimization of make span *(Cmax)*, where make span is defined as the completion time of the final job to leave the system. The minimization of this criterion combines the desire to assure high utilization of the production resources with the desire to ensure early satisfaction of customer demand. This single objective cannot, however, reflect all important aspects of the flow shop optimization. In this study we take into account the following objective: minimization of completion time (*Cmax)*, and total flow time (*Fmax)*.

2. LITERATURE REVIEW

During the last 40 years, the $n/m/P/C_{max}$ problem has held the attention of many researchers [2]. Although optimal solutions of $n/m/P/C_{max}$ problems can be obtained via enumeration techniques such as exhaustive enumeration and branch and bound method [3], these methods may take a prohibitive amount of computation time even for moderate size problem. Sequencing methods in the literature can be broadly categorized into two types of approaches, namely optimization and heuristic. Optimization approaches guarantee to obtain the optimum sequence, whereas heuristic approaches mostly obtain near-optimal sequences. Among the optimization approaches, the algorithm developed by Johnson [4] is the widely cited research dealing with sequencing *n* jobs on two machines. Lomnicki [5] proposed a branch and bound technique to find the optimum permutation of jobs. Since the flow shop scheduling problem has been recognized to be NP-hard, the branch and bound method cannot be applied for large size problems. This limitation has encouraged researchers to develop efficient heuristics. For practical purposes, it is often more appropriate to look for heuristic method that generates a near-optimal solution at relatively minor computational expense. This leads to the development of many heuristic procedures.

Currently available heuristics for solving this problem in literature can be classified into two categories: constructive heuristics and improvement heuristics [6]. In a constructive heuristic, once a job sequence is determined, it is fixed and cannot be reversed. In constructive category, methods developed by Palmer [7], Campbell et al. [8], Gupta [9], Dannenbring [10], Rock and Schmidt [11] and Nawaz et al. [12] can be listed. Mostly, these methods are developed on the basis of the Johnson's algorithm. Turner and Booth [13] and Taillard [14] have verified that the method proposed by Nawaz et al. [12], namely NEH, performs well among the constructive methods tested. On the other hand, Osman and Potts [6], Widmer and Hertz [2], Ho and Chang [15], Ogbu and Smith [16], Taillard [14], Nowicki and Smutnicki [17], and Ben-Daya and Al-Fawzan [18] have developed improvement heuristics for the same problem. The improvement heuristics start with an initial solution and then provide a scheme for iteratively obtaining an improved solution. In recent years, studies with meta-heuristics have been extensively carried out on this argument. The meta-heuristic is a rather general algorithmic framework that can be applied to different optimization problems with minor modifications. Essentially, it is a type of randomized improvement heuristic[6]. Methods of this type include Genetic Algorithm [19,20], Simulated Annealing [21,22] and Tabu search [23]. Literature shows that these methods can obtain very good results for NP-hard combinatorial optimization problems. Nowicki and Smutnicki [24]

have developed new algorithm called Modified Scatter Search Algorithm (MSSA). MSSA produced 20 new beter upper bound solutions among 30 very hard,unsolved yet instances from common bench mark set. Another meta heuristic is given by Stutzle [25] called Iterated Local Search (ILS).According to the tests conducted by Stutzle, the ILS algorithm is much better than the Tabu Search of Taillard [14] and also better than the Tabu search of Nowicki and Smutnicki (TSAB) [17]. Ruben Ruiz and Concepcion Maroto [26] compared 25 methods,ranging from the classical Johnson's algorithm or dispatching rules to the most recent metaheuristics, including tabu search,simulated annealing,genetic algorithms,iterated local search and hybrid techniques for the benchmark problems[27]. In the paper[26],all the algorithms and methods are coded Delpi6.0 and run in the AthlonXP 1600+ computer with 512 Mbytes of main memory.Methods used for comparision are well known metaheuristics such as Osman and potts[6] SA algorithm (SAOP),Widmer and Hert's [2] SPIRIT,Chen et al.'s[28] GA algorihm (GAChen),Reeves [29] GA algorithm (GAReev), Hybrid GA+Local Search by Murata et al.[30](GAMIT), stutzle's ILS and GA by Ponnambalam et al.[31](GAPAC). Reza hejazi and Saghafian[32] have described complete survey of flowshop scheuling problems up to 2004. Rajendran and Ziegler (33, 344) presented bicreteria problems with heuristic and metaheuristic methods for two different objectives. Rajendran (34) proposed ant colony algorithms for the problem of scheduling in permutation flow shops with the objective of minimizing the makespan, followed by the consideration of minimization of total flow time of jobs. Varadharajan and Rajendran (35) proposed a multi-objective simulated annealing algorithm for scheduling in flow shops to minimize the makespan and total flow time of jobs. Rajendran (36,37,38,39,40) proposed various methods for scheduling problems of same kind. Tasgetiren et al (41) proposed a particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem.

This paper is organized as follows: Section 3 describes the formulation of sequencing problem. In Section 4, the elements of the Scatter search method based on the sequencing problems are discussed . A simple problem is solved by the proposed method,scatter search in section 5. The computational results obtained by the application of this method to the problems selected from benchmark problems [27] are discussed in Section 6. Section 7 includes discussions and conclusions

3. TERMINOLOGY AND PROBLEM FORMULATION

Let

- t_{ij} be the processing time of job i on machine j
- $t_{[i]j}$ the processing time on machine j of the job found in the ith position of a schedule
- n total number of jobs to be scheduled
- m number of machines in the flow shop
- σ the set of jobs already scheduled, out of n jobs
- n' the number of jobs in σ
- π the set of unscheduled jobs
- $q(\sigma,j)$ the completion time for the partial schedule of σ on machine j
- a an unscheduled job in π$q(\sigma_a,j)$the completion time of job a on machine j,when job a is appended to σ.
- F_σ total flow time of jobs in σ

Considering basic assumption such as non interface at machines, non simultaneous processing of jobs and no job passing, the completion time of partial schedule σa on machine j, q(σa,j), can be obtained by the following recursive equation :

$$q(\sigma a,j) = \max [q(\sigma,j) \; ; \; q(\sigma a,j-1)] + t_{aj}$$

While the makespan of the partial σa is given by

$$M_{\sigma a} = q(\sigma a,m)$$

The total flow time of jobs in σa is given by

$$F_{\sigma a} = F_\sigma + q(\sigma a,m)$$

The above equation would hold good for a typical flow shop.

4.SCATTER SEARCH

4.4 ELEMENTS OF SCATTER SEARCH

The solution approach that is developed for this permutation problem consists of an adaptation of SS. Scatter Search is an instance of the so-called evolutionary method, which is not based solely on randomization as the main mechanism for searching. It constructs solutions by combining others by means of strategic designs that exploit the knowledge on the problem at hand. The goal of these procedures is to enable a solution procedure based on the combined elements to yield better solutions than one based on the original elements.

Compared to other evolutionary methods, SS operates with a population of solutions, rather than with a single solution at a time, and employs procedures for combining these solutions to create new ones. The meaning of “combining” and the motivation for carrying it out has a rather special origin and character in the SS setting. One of the distinguishing features of this approach is its intimate association with the Tabu Search (TS) metaheuristic, and hence, its adoption of the principle that search can benefit by incorporating special forms of adaptive memory along with procedures particularly designed for exploiting that memory. More about the origin and multiple applications of SS can be found in Glover et al. [1]. The basic steps involved in the static scatter search are:

Step 1: Use the Diversification Generator to generate diverse trial solutions from the seed solution(s)

Step 2: Use the Improvement Method to create one or more enhanced trial solutions

Step 3: With these initial solutions update the Reference Set (RefSet)

Step 4: Repeat

4.1 Generate subsets of the RefSet.

4.2 Combine these subsets and obtain new solutions.

4.3 Use the Improvement Method to create a more enhanced trial solution

4.4 While continuing to maintain and update the Reference Set.

Until Refset is stable (No new solutions are included)

Step 5: if iterations (Steps 1–4) elapse without improvement stop else return to step.

1. A ***Diversification Generation Method*** to generate a collection of diverse trial solutions, using an arbitrary trial solution (or seed solution) as an input.
2. An ***Improvement Method*** to transform a trial solution into one or more enhanced trial solutions. (Neither the input nor the output solutions are required to be feasible, though the output solutions will more usually be expected to be so. If no improvement of the input trial solution results, the "enhanced" solution is considered to be the same as the input solution.)
3. A ***Reference Set Update Method*** to build and maintain a *reference set* consisting of the b "best" solutions found (where the value of b is typically small, e.g., no more than 20), organized to provide efficient accessing by other parts of the method. Solutions gain membership to the reference set according to their quality or their diversity.This problem the value of b is 10.
4. A ***Subset Generation Method*** to operate on the reference set, to produce a subset of its solutions as a basis for creating combined solutions.
5. A ***Solution Combination Method*** to transform a given subset of solutions produced by the Subset Generation Method into one or more combined solution vectors. Specific processes for carrying out these steps are described in Glover (1997).

4.5 IMPLEMENTATION ON TEST PROBLEM

The main aim of this step is to find the make span and the total flow time using Scatter search approach for the following job and machine combination as shown in the table

MACHINES

JOBS		1	2	3
J	1	20	10	30
O	2	10	10	30
B	3	40	50	10
S	4	10	20	10
	5	10	10	10

Step 1: Seed Solution
Obtai seed sequence by random generation.
1-4-3- Make span=170, TFT=590
Step Diversification Step
Takin =2, we get
P(2:2) ,2) P(2:1)=(1,3,5) So, p(2)=(4-2-1-3-5)
Step 3 nprovement Step
For i<
Delete from jth place in p and insert between current pi-1 and pi
P'=(p1,….,pi-1,pj,pi,….,pj-1,pj+1,….,pn)
Case 1: i=2,j=3
New Sequence = (4-1-2-3-5)(MS=150, TFT=200)
Case 2: i=3,j=4
New Sequence =(4-2-3-1-5)(MS=160, TFT=560)
Case 3: i=4, j=5
New Sequence = (4-2-1-5-3)(MS=150, TFT=470)
For i >j:
Delete pj from jth place in p and insert between current pi and pi+1
P'=(p1,….,pj-1,pj+1,….,pi,pj,pj+1….,pn)
Case 1: i=2,j=1
New Sequence = (2-4-1-3-5)(MS=150, TFT=490)
Case 2: i=3,j=2
New Sequence = (4-1-2-3-5)(MS=150, TFT=500)
Case 3: i=4,j=1
New Sequence = (2-1-3-4-5)(MS=160, TFT=470)
Step 4: Order solutions in the order of best objective function
4-1-2-3-5
4-2-3-1-5 } = b1
4-2-1-5-3
2-4-1-3-5
4-1-2-3-5 } = b2

Step 5: Building Reference Set
Reference Set: 4-1-2-3-5
4-2-3-1-5
4-2-1-5-3
P-Reference Set: 2-4-1-3-5
4-1-2-3-5
2-1-3-4-5
Step 6: Scatter Search Implementation
Let p= (p1, p2, p3,……pn) and
q= (q1,q2,q3,……pn)

$$d(p,q) = \sum_{i=1}^{n} |p_i - q_i|$$

d(p,q)= number of times pi+1 does not immediately follow pi in q, for i=1,..,n-1
Step 7: Finding d (RefSet) where x is a solution in P-RefSet.
For e.g.

For (2-4-1-3-5),
d1 is taken between 2-4-1-3-5 and 4-1-2-3-5
$p_1=2$, $p_2=4$, $p_3=1$, $p_4=3$, $p_5=5$. $q_1=4$, $q_2=1$, $q_3=2$, $q_4=3$, $q_5=5$
d1= |2-4| + |4-1| + |2-1| + |3-3| + |5+5| =2+3+1+0+0
d1=6
Similarly,
For (2-4-1-3-5), d1=6, d2=12, d3=8 $D_{1max}=12$
For (4-1-2-3-5), d1=4, d2=10, d3=8 $D_{2max}=10$
For (2-1-3-4-5), d1=4, d2=12, d3=6 $D_{3max}=12$
So, $D_{1max}=D_{3max}>D_{2max}$
Step 8: Reference Set Update
New RefSet: 4-1-2-3-5
4-2-3-1-5
4-2-1-5-3
2-4-1-3-5
2-1-3-4-5
4-1-2-3-5
Step9: Solution Combination
Take two sequence at a time from the updated reference set for combination and find the best solution from them
For ex: Taking the sequences
2-1-3-4-5 and 4-1-2-3-5
Gives 2-4-1-3-5, by combination.
By repeating the steps2-9 until the termination condition is reached. The Best sequence using Scatter Search Algorithm is:
(1-4-2-3-5) (MS=150, TFT=495)
2-1-3-4-5

5. RESULTS

The coding for the multi-objective flow shop scheduling using the Scatter search Algorithm is developed and 9 problem instances with the number of jobs ranging from 5 to 20 and number of machines also ranging from 5 to 20 have been generated and the final sequences for each combination instances are presented in the subsequent tables.

6. CONCLUSION

The main aim of this research is to explore the potential of Scatter Search for scheduling problems of a flow-shop. The inherent. weakness of many search procedures is that they often get trapped in a region around some local minima. Their ability to breakout of such entrapments and achieve better, ideally global minima, is based on their capacity to provide a suitable mixture of intensification and diversification. Scatter search also provides unifying principles for joining solutions based on generalized path constructions and by utilizing strategic designs where other approaches resort to randomization. Additional advantages are provided by intensification and diversification mechanisms that exploit adaptive memory, together with processes to avoid generating duplicate solutions at various stages. In fact the Scatter search metaheuristic is able to achieve best results compared to other metaheuristics (42-48). This paper will help the researchers to use Scatter Search method as comparative tool for their approaches for the research work

7. FUTURE SCOPE OF THE PROJECT

This paper addresses the problem of scheduling in flow shop with the objective of minimizing makespan and total flow time. A correct formulation of makespan and flow time in a flow shop is first presented. Then scatter search approache is highlighted under the multi objective criteria with the aid of a numerical illustration. Finally, the proposed scatter search algorithm is applied to a number of multi machine and multi job combinations using proper coding.

Further other evolutionary heuristics like Tabu Search , Simulated Annealing,Ant colony algorithm, Particle Swarm optimisation, Mmentic Algorithm,etc. can be applied to the same multi objective criteria environment with appropriate coding for different types of multi job and multi machines combinations and the results can be compared suitably and to find any percentage deviations among the metaheuristics.Furthermore, more number of objectives can be included during the study like total idle times on the machines etc. to make the study more accurate and practical

Table 6.10 Performance of scatter search for multi objective criteria

No. of Jobs	No.of Machines	Initial Sequence(seed)	Optimal Job sequence	Makespan	Total Flow Time
5	5	4-2-1-5-3	4-1-2-3-5	45	183
	10	1-2-3-5-4	4-5-3-2-1	73	321
	20	4-5-3-2-1	3-4-2-5-1	121	543
10	5	3-7-5-4-10-9-1-2-6-8	8-9-6-5-7-3-4-2-1-10	769	5179
	10	3-7-8-6-5-10-4-2-1-9	9-3-5-10-7-1-8-6-4-2	1200	8570
	20	4-9-2-5-7-1-10-3-6-8	2-9-3-5-10-4-7-6-8-1	1758	13648
20	5	13-9-10-16-14-2-19-8-12-20-11-3-4-5-6-7-15-1-18-17	15-8-3-17-14-9-4-13-19-1-11-5-7-18-16-12-2-6-10-20	1297	15344
	10	4-2-10-19-3-12-6-14-20-1-17-11-16-5-15-7-9-8-18-13	18-5-9-3-4-14-12-17-19-6-20-2-10-13-8-15-7-11-1-16	1643	22697
	20	7-12-8-16-9-6-4-11-17-1-2-14-13-20-3-19-5-10-15-18	8-7-9-15-16-17-12-13-1-20-10-11-5-6-14-18-2-3-4-19	2352	38430

REFERENCES

1. Glover F, Laguna M, Marti R. (2000), "Fundamentals of scatter search and path relinking". Control and Cybernetics. 29(3): 653–684.
2. Widmer M, Hertz A. (1989), "A new heuristic method for the flow shop sequencing problem". European Journal of Operational Research 41:186–93.
3. Pinedo M. (1995), "Scheduling: theory, algorithm, and system". Englewood Cliffs, NJ: Prentice-Hall,
4. Johnson SM. (1954), "Optimal two- and three-stage production schedules with setup times included". Naval Research Logistics Quarterly; 1(1):61–8.
5. Lomnicki ZA. (1965), "A branch and bound algorithm for the exact solution of the three-machine scheduling problem". Operational Research Quarterly; 16(1):89–100.
6. Osman IH, Potts CN (1989). "Simulated annealing for permutation flow-shop scheduling". OMEGA, International Journal of Management Science; 17:551–7.
7. Palmer DS (1965). "Sequencing jobs through a multi-stage process in the minimum total time-a quick method of obtaining a near optimum". Operational Research Quarterly; 16(1):101–7
8. Campbell HG, Dudek RA, Smith ML. (1970) "A heuristic algorithm for the n-job, m-machine sequencing problem". Management Science 16(B): 630–7.
9. Gupta JND. (1971) "A functional heuristic algorithm for the flow shop scheduling problem". Operational Research Quarterly; 22(1):39–47.
10. Dannenbring DG. (1977), "An evaluation of flow shop sequencing heuristics". Management Science 23(11): 1174–82.
11. Rock H, Schmidt G. (1982), "Machine aggregation heuristics in shop scheduling". Methods of Operations Research; 45: 303–14.
12. Nawaz, M., Enscore, J.E.E., Ham, I. (1983), "A heuristic algorithm for the m-machine, n-job flow shop sequencing problem". OMEGA: International Journal of Management Science 11 (1), 91–95.
13. Turner S, Booth D. (1987), "Comparison of heuristics for flow shop sequencing".OMEGA : International Journal of Management Science; 15:75–8.
14. Taillard E. (1990), "Some efficient heuristic methods for the flow shop sequencing problem". European Journal of Operational Research; 47:65–74.
15. Ho J C, Chang Y L. (1990), "A new heuristic for the n-job, m-machine flow shop problem". European Journal of Operational Research; 52:194–206.
16. Ogbu FA, Smith DK. (1990), "The application of the simulated annealing algorithm to the solution of the n/m/Cmax flow shop problem". Computers and Operations Research; 17:243–53.
17. Nowicki E, Smutnicki C. (1996), "A fast tabu search algorithm for the permutation flow shop problem". European Journal of Operational Research; 91:160–175.
18. Ben-Daya M, Al-Fawzan M. (1998), "A tabu search approach for the flow shop scheduling problem". European Journal of Operational Research 109:88–95.
19. Goldberg DE. (1989.), "Genetic algorithms in search, optimization and machine learning. Reading". MA: Addison-Wesley.
20. Holland JH. (1975), "Adaptation in natural and artificial systems". Ann Arbor: The University of Michigan Press.
21. Van Laarhoven PJM, Aarts EHL. (1987), "Simulated annealing: theory and applications". Dordrecht: D. Reidel Publ. Co.,
22. Metropolis N, Rosenbluth A,Rosenbluth M,Teller A,Teller E(1953),"Equation of state calculation by fast computing machine". Journal of Chemical Physics; 21:1087–92.
23. Glover, F. and M. Laguna. (1997), "Tabu Search". Kluwer Academic Publishers.
24. Eugeniusz Nowicki, Czeslaw Smutnicki (2006),"Some aspects of scatter search in the flow-shop problem",European Journal of Operational Research 169 ,654–666
25. Stutzle, T., (1998). "Applying iterated local search to the permutation flow shop problem". Technical Report, AIDA-98-04,FG Intellektik,TU Darmstadt.
26. Ruben Ruiz, Concepcion Maroto (2005), "A comprehensive review and evaluation of permutation flowshop heuristics",European Journal of Operational Research, 165, 479–494
27. Taillard E. (1993), "Benchmarks for basic scheduling problems". European Journal of Operational Research. 64,278–285.
28. Chen, C.-L., Vempati, V.S., Aljaber, N., 1995. "An application of genetic algorithms for flow shop problems". European Journal of Operational Research 80, 389–396.
29. Reeves, C., Yamada, T., 1998. "Genetic algorithms, path relinking, and the flowshop

sequencing problem". Evolutionary Computation 6 (1), 45–60.

30. *Murata, T., Ishibuchi, H., Tanaka, H., 1996. "Genetic algorithms for flowshop scheduling problems". Computers and Industrial Engineering 30 (4), 1061–1071.*
31. *Ponnambalam, S.G.,Aravindan, P., Chandrasekaran, S., 2001."Constructive and improvement flow shop scheduling heuristics:An extensive evaluation". Production Planning and Control, 12 (4), 335–344.*
32. *Reza hejazi.S and. Saghafian. S (15 July 2005),"Flowshop-scheduling problems with makespan criterion: a review", International Journal of Production Research,Vol. 43, No. 14, , 2895–2929.*
33. *Rajendran C. and Ziegler H. (2004) 'Ant-colony algorithms for permutation flow shop scheduling to minimize makespan/total flow time of jobs', European Journal of Operational Research, Vol. 155,pp.426-438.*
34. *Rajendran C. and Ziegler H. (1997) 'An efficient heuristic for scheduling in a flow shop to minimize total weighted flow time of jobs', European Journal of Operational Research, 103,pp.129–138..*
35. *Varadharajan T.K. and Chandrasekharan Rajendran (2005) "A multi-objective simulated-annealing algorithm for scheduling in flow shops to minimize the makespan and total flowtime of jobs." European journal of operational research, Vol. 167, no. 3 pp. 772-795.*
36. *Rajendran C.and Ziegler H. (1997), "A heuristic for scheduling to minimize the sum of weighted flow time of jobs in a flow shop with sequence-dependent setup times of jobs", Computers & Industrial Engineering,Vol. 33, Issues 1-2, Pp. 281-284.*
37. *Chandrasekharan Rajendran (1991), "A Flow shop Scheduling Algorithm to Minimize Total Flow time", Journal of Operations Research Society of Japan, Vol. 34, No. 1, pp 28 - 46.*
38. *Chandrasekharan Rajendran and Hans Ziegler,(2004), "Ant-colony algorithms for permutation flowshop scheduling to minimize makespan/total flowtime of jobs". European Journal of Operational Research,Vol. 155, Issue 2, 2004, Pp. 426-438.*
39. *Rajendran C. (1992) Two-stage flow shop scheduling problem with bi criteria. Journal of the Operational Research Society, 43, pp.871–884.*
40. *Rajendran C. (1995) 'Heuristics for scheduling in flow shop with multiple objectives.' European Journal of Operational Research 82, pp.540–555.*
41. *Tasgetiren, M. Fatih & Liang, Yun-Chia & Sevkli, Mehmet & Gencyilmaz, Gunes, (2007),"A particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem," European Journal of Operational Research, Elsevier, vol. 127(3), pp.1930-1947.*
42. *Noorul haq A., and Saravanan M., Vivekraj A.R. and Prasad T. (2007) 'A scatter search algorithm for general flow shop scheduling problem', International journal of Advanced manufacturing Technology, 31,731-736.*
43. *Noorul haq A. and Saravanan M. (2007) 'A Scatter Search Approach for minimizing Earliness and Tardiness Penalties in single machine scheduling around large restrictive Common Due Windows', International journal of Applied Management and Technology, vol.5,No.2,pp. 76-91.*
44. *Saravanan M and Noorul haq A. (2007) 'Evaluation of Scatter Search Approach for Scheduling Optimization of Flexible Manufacturing Systems', International journal of Advanced manufacturing Technology. - Published in online (August 3, 2007).*
45. *Saravanan M and Noorul haq A. Vivekraj A.R. and Prasad T. (2007) 'Performance Evaluation of Scatter Search Method for Permutation Flow-Shop Sequencing Problems', International journal of Advanced manufacturing Technology. - Published in online (June15, 2007).*
46. *Noorul haq A. and Saravanan M. (2007) 'Evaluation of Scatter Search Approach for Minimizing Earliness and Tardiness Penalties in Single Machine Scheduling', Accepted for Publication in International journal of manufacturing Science and Technology.*
47. *Saravanan M and Noorul haq A, 'A Scatter Search Method to Minimize Makespan of Cell Scheduling Problem', Accepted for publication in International Journal of Agile Systems and Management.*
48. *Saravanan M and Noorul haq A, 'A Scatter Search Algorithm for Scheduling Optimisation of Job Shop Problems', Accepted for publication in International Journal of Product Development.*

