International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-4, November 2019

Solving SDST Flow shop Scheduling Problems using Hybrid Artificial Bee Colony Algorithms

R Sanjeev Kumar, G Robert Singh, N Mathan Kumar

***Abstract*: *Sequence dependent setup time scheduling plays a vital role in real manufacturing industries. Setup time depends on the job sequence. Heuristic or dispatching rules are suitable for producing good initial seed sequence. For this reason, heuristic or dispatching rules based artificial bee colony algorithm is used to minimize the makespan and total flowtime in SDST shop problems. Computation result confirms that heuristic based method produced better result when compared to the dispatching based method. The recital of the algorithms is estimated by conducting Statistical test*.**

Keywords: SDST Shop, Makespan, Total Flow Time, ABC Algorithm

I. INTRODUCTION

Setup time is necessary for setting the tools, jobs cleaning and inspection work in the scheduling. SDST means sequence dependent setup time. It is considered in all real-life situations for exploiting multipurpose machine. Setup time is required between the sequences. It is an NP-hard problem to solve even two machines. SDST scheduling is one of the types in scheduling. Corwin and Esogbue [1] considered dynamic programming method to reduce makespan value in SDST flow shop. Das et al. [2] used new heuristic technique to tackled SDST problems with the target of minizingmakespan. Parthasarathy and Rajendran [3] applied simulated annealing algorithm to minimize maximum tardiness and total tardiness of a job in SDST shop. Rios-Mercado and Bard [4] applied branch and cut method to optimize makespan in flow shop. Rios-Mercado and Bard [5] developed an enhanced heuristic to reduce the makespan value in SDST shop. They ascertained the parameter value and evaluated the problem instances. Rajendran and Ziegler [6] developed a heuristic method to solve SDST flow shop with the objective of flow time and tardiness. Ruiz et al.[7] applied genetic algorithm to solved SDST scheduling shop problem. They calibrated genetic algorithm parameter and operators by using design of experiment method. Tseng et al. [8] minimized makespan in SDST shop using penalty based heuristic algorithm. They compared the penalty heuristic result and saving index heuristic result.Gajpal et al. [9]solved SDST problems with the objective of makepan using an ant colony algorithm. Eren and Guner [10] developed integer model and heuristics for solving two machine setup time problems. They minimized total flow time and tardiness. Eren [11]developed heuristic methods to solve two machine problems with the objective of makespan, total flow time, earliness and tardiness. Wang and Cheng [12] used heuristic algorithm for setup time scheduling problems with availability constraint. Allahverdi and Kovalyov [13] reported the setup time scheduling problems. Ruiz and Stutzle [14] proposed iterated greedy algorithm to minimize makespan and total weighted tardiness. They conducted statistical analysis and reported the algorithm performance. Mansouri and Hendizadeh [15] considered Pareto optimisation approach to minimize setups and makespan for two machine flow shop using genetic algorithms and simulated annealing. Dhingra and Chandna [16] developed heuristic based genetic algorithm for solving the SDST shop problem with the objective of makespan, earliness and tardiness. Dhingra and Chandna [17] developed modified heuristic genetic algorithm to minimize weighted tardiness and makespan in SDST flow shop scheduling with the objective. They reported that modified heuristic performance for large size problems.Eren [18]devolved integer programming model to reduce makespan and total flow time. Kumar et al. [19]proposed genetic algorithm with different variant (GA^1, GA^2, GA^3, GA^4) to reduce the total tardiness and makespan value in SDST shop. They reported that GA^3 produced better result than other algorithms. Hooda and Dhingra [20] proposed heuristic based simulated annealing algorithms to SDST shop with the objective of number of tardy jobs and makespan. They reported that SA (NEH) performed better than SA (NEH_EDD). Vanchipura and Sridharan [21] conducted statistical analysis with different stages of setup time for confirming the algorithm performance with the objective of makespan. Vanchipura and Sridharan [22] carried out a full factorial experiment for settings the parameters. They applied hybrid genetic algorithms to solve SDST shop. Hatami et al. [23] applied both metaheuristic and heuristic algorithm for SDST permutation flow shop to problem with the objective of makespan. Satyanarayana and Pramiladevi [24]

Manuscript published on November 30, 2019.
*Correspondence Author
R Sanjeev Kumar*, Mechanical Department, Swarnandhra College of Engineering & Technology, Narsapur, Andhra Pradesh, India.
G Robert Singh, Mechanical Department, Swarnandhra College of Engineering & Technology, Narsapur, Andhra Pradesh, India.
N Mathan Kumar, Mechanical Department, Akshaya College of Engineering & Technology, Coimbatore, Tamil Nadu, India.

developed three new heuristic algorithms with the objective of tardiness, makespan, earliness and number of tardy jobs for SDST scheduling. They evaluated the algorithms regarding relative percentage difference (RPD). Jeong and Shim [25]proposed several heuristic algorithms to solve re-entrant shop problems. They tested the heuristic with randomly generated problems. They reported the performance of the algorithm and solution quality in real manufacturing system.From the literature, it is clear that no previous work was carried using Heuristic based Artificial Bee Colony (HABC) algorithms to optimize makespan and total flow time in SDST scheduling. Therefore, in this paper, HABC algorithms used to solve the flow shop scheduling SDST with problems. The enduring of the paper is organized as follows: it describes flow shop scheduling problem with SDST; it explains the ABC and HABC algorithm; it illustrates the numerical example for SDST flow shop problem using HABC; it explicates the experimental results; finally, it furnishes the conclusions.

II. PROBLEM STATEMENT

Setup time is considered when the jobs processed in each machine based on the sequences. Processing sequence is same for each job. The subsequent hypothesis is considered as follows; all jobs existing on time zero, setup time is considered. It is based on the job sequence, Jobs can wait for process on next machine, machine performs only one operation at a time, during the schedule machines never breakdown. The aspiration is to reduce the makespan and total flow time in SDST shop. It is represented as.

Min Z = $\alpha C_{max} + \beta$ TFT

$\alpha + \beta = 1$

Notation:

I job index

J machine index

p_{ij} job processing time

s_{ijk}job setup time

α &β weight factors

CmaxMakespan

TFT Total Flow Time

Z combined objective function.

III. METHODOLOGY

A. Artificial Bee Colony Algorithm

Artificial bee colony (ABC) algorithm is developed based on foraging activities of honey bees. Three types of bees are involved in this process. The number of food sources and number of employed bees are equal. All the employed bees are haphazardly found the food sources. Then, they move to the neighborhood food. It evaluates the nectar amount. The best food source is stored by employed bees. Employed bee shares the food information to the onlooker bees through the waggle dance. If the food is not available, then it becomes a scout. It finds a new food source. The above process is continued until a maximum termination is reached [26].ABC is applied in many fields. Kang et al. [27] solved structural problems using simplex artificial bee colony Algorithm. Karaboga and Ozturk [28] solved pattern classification problem using artificial bee colony algorithm. Zhu and Kwong [29] developed Gbest guided ABC algorithm for numerical function optimisation. Li et al. [30] solved flexible job shop problem using Pareto based ABC algorithm. Akbaria et al. [31] solved resource project problem using ABC algorithm. Tasgetiren et al. [32] proposed discrete ABC algorithm to reduce the total flow time in flow shop. Liu and Liu [33] minimized makespan in flow shop problem using hybrid ABC algorithm. Bulut and Tasgetiren [34] solved economic lot scheduling problem using ABC algorithm. Basti and Sevkli [35] solved facility location problem using ABC algorithm. Han et al. [36] solved blocking shop scheduling problem using hybrid ABC (HABC) algorithm. Caniyilmaz et al. [37] solved parallel machine scheduling using ABC algorithm. The HABC algorithm flowchart is represented in Figure 1.

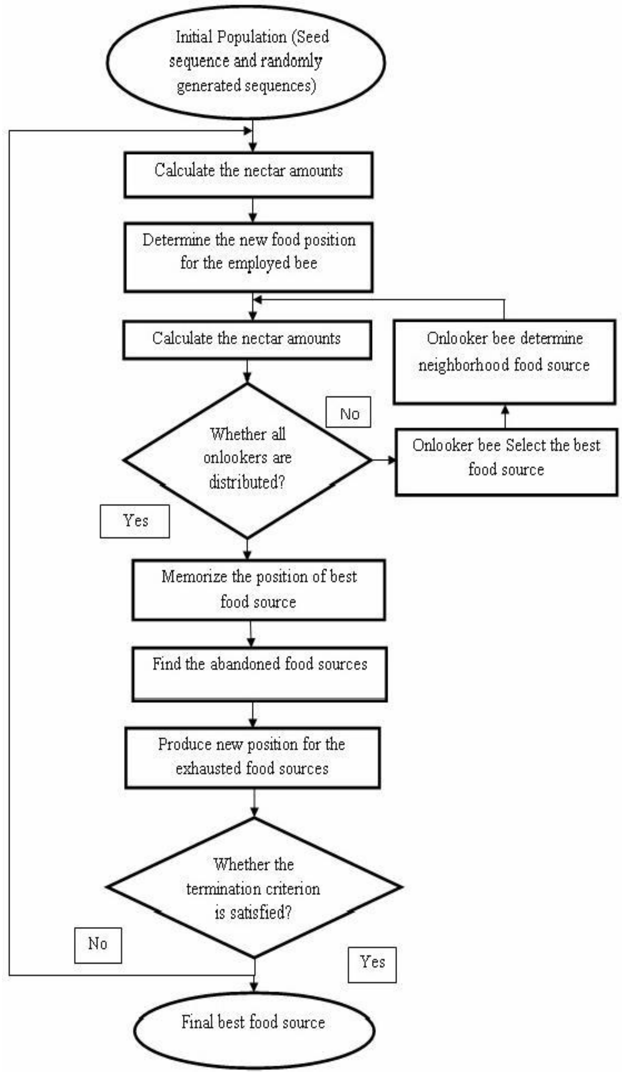

Fig. 1. ABC algorithm flowchart

B. Procedure

Step 1 Initial food sources (xi) are generated randomly.
Step 2 Evaluate each food source fitness value $f(x_i)$
Step 3 Apply a neighborhood operator on xi to determine neighborhood food and its fitness value f (x'_i).

Step 4 Using greedy selections to store the best food source in memory.
Step 5 Based on probability (Pi), the onlooker bee prefer a food source xi .[Pi = Fitness value / Sum of all fitness values]
Step 6 Apply a neighborhood operator to determine neighborhood food source.
Step 7 Use greedy selections for selecting the best food source and store in memory.
Step 8 If the food source is not exists then it acts as scout.
Step 9 Supreme food is memorized
Step 10 Cycle is continued until maximum number of iteration is reached.

C. Hybrid ABC algorithm

In the proposed method, initial seed sequence is produced by Nawaz Enscore Ham (NEH) heuristic or dispatching rule. Some important dispatching rules are Shortest Processing Time (SPT), Longest Processing Time (LPT) and Random. Creating initial population plays an important role to persuade the quality of solution. Best initial solution gives good results. HABC algorithm is developed to produce the seed sequence. The variants of ABC algorithm is HABC1 (SPT), HABC2 (LPT), HABC3 (random) and HABC4 (NEH heuristic) in Table 1.

Table- I: Variants of ABC algorithm

S. No.	Heuristics	Description
1	HABC1	Initial seed sequence is obtained by using SPT rule
2	HABC2	Initial seed sequence is obtained by using LPT rule
3	HABC3	Initial seed sequence is obtained by using random method
4	HABC4	Initial seed sequence is obtained by using NEH heuristics

The initial population consists of seed sequence and (p – 1) randomly generated sequence. The seed sequence is generated byHABC algorithm. Calculate and evaluate the fitness value of each food source. The neighborhood operators are used to find theneighborhood food source. Greedy selection is applied to memorize the supreme food source and informed to the onlooker bees.Onlooker bee selects the supreme food source. The neighborhood food source is not improved then it becomes scout bee. The procedure is continued until the large number of iteration attained.

D. Numerical illustration

Two machines five jobs (n * m) small scale problem has been considered. This problem processing time is given in Table 2.

Table- II: Jobs processing time

Machines	Jobs				
	1	2	3	4	5
Pj1	19	11	8	30	69
Pj2	5	1	2	35	17

Table-III: sequence dependent setup time on machine 1

Sj1k	1	2	3	4	5
1	-	16	12	13	19
2	20	-	15	17	18
3	22	19	-	17	14
4	22	20	17	-	20
5	5	10	2	18	-

Table- IV: sequence dependent setup time on machine 2

Sj1k	1	2	3	4	5
1	-	9	17	6	14
2	23	-	8	10	24
3	12	20	-	24	25
4	20	24	25	-	9
5	22	4	18	5	-

Step 1: The initial population consists of seed sequence and (p – 1) randomly generated sequence. The seed sequence is generated by HABC heuristic. The population size (P) is 5. Each employed bee (job sequence) has been assigned to a food source (solution) in Table 5.

Table- V: Initial food source with its fitness value

Initial food source	Objective function (Z)	Fitness value (f(xi) = 1 / Z)
IS1 → 1-2-3-4-5	368.5	0.002713
IS2 → 3-4-5-2-1	422.5	0.002366
IS3 → 5-3-4-1-2	448.5	0.002229
IS4 → 4-1-2-3-5	387.5	0.002580
IS5 → 2-5-1-4-3	429	0.002331

Step 2 Compute and Evaluate each sequence fitness value f(xi) for. i = 1 …… n. Initial food sources (parent sequence) and its fitness values represented in Table 5. Each fitness value is assumed as each job sequence value.
Step 3 Neighborhood operators are used to find neighborhood sequence and its value in Table 6.

Table- VI: Neighborhood food sources fitness values

Neighborhood food source (NS)	Objective function (Z)	Fitness value f (xi’)
NS1 → 4-3-2-1-5	402.5	0.002484
NS2 → 2-5-4-3-1	462.5	0.002162
NS3 → 1-4-3-5-2	395.5	0.002528
NS4 → 3-2-1-4-5	389	0.002570
NS5 → 4-1-5-2-3	454	0.002202

Step 4 Greedy choice has been used to pick the supreme sequence and the same has been stored in the memory. Compare the initial sequence and neighborhood sequence fitness value. Store the best fitness value and its sequence as parent sequence. It is given in Table 7.

Table- VII: Selection of best food source

Best food source	Objective function (Z)	Fitness value (f(xi))
1-2-3-4-5	368.5	0.002713
3-4-5-2-1	422.5	0.002366
1-4-3-5-2	395.8	0.002528
4-1-2-3-5	387.5	0.002580
2-5-1-4-3	429	0.002331

Step 5 The employed bees complete their process and it shares food information. Based on the probability value (Pi), the onlooker bees prefer a food source

$$Pi = \frac{f(xi)}{\sum_{i=1}^{n} f(xi)}$$

The best food source with its probability is given in Table 8.

Table- VIII: Probability for best food source

Best food source	Probability (Pi)
1-2-3-4-5	368.5
3-4-5-2-1	422.5
1-4-3-5-2	395.8
4-1-2-3-5	387.5
2-5-1-4-3	429

First onlooker bee (OB1) will select the highest probability food source (job sequence) and second onlooker bee (OB2) will select the next highest probability food source; likewise, all food sources are allotted to all onlooker bees. Select the best sequence based on the probability in Table 9.

Table- IX: Allocation of food source for onlooker bees

Onlooker bee food source (OB)	Fitness value (f(xi))	Probability (Pi)
OB1 → 1-2-3-4-5	0.002713	0.2167
OB2 → 4-1-2-3-5	0.002580	0.2061
OB3 → 1-4-3-5-2	0.002528	0.2019
OB4 → 3-4-5-2-1	0.002366	0.1890
OB5 → 2-5-1-4-3	0.002331	0.1862

Step 6 Each onlooker bee (OB1, OB2 … OBn) determines the neighborhood food source (NS1, NS2 … NSn) and compare their fitness value. If the neighborhood food source (neighborhood sequence) fitness value is better than the onlooker bee (Parent Sequence). It is represented as best. If not, it is represented as worst. It is given in Table 10.

Table- X: Allocation of food source for onlooker bees

Onlooker bee	Objective	Fitness	Neighborhood food	Objective	Fitness
OB1 → 1-2-3-4-5	368.5	0.002713	NS1 → 3-2-1-4-5	389	0.00257 (worst)
OB2 → 4-1-2-3-5	387.5	0.00258	NS2 → 2-1-4-3-5	380.5	0.002628 (best)
OB3 → 1-4-3-5-2	395.5	0.002528	NS3 → 3-4-1-5-2	557	0.001795 (worst)
OB4 → 3-4-5-2-1	422.5	0.002366	NS4 → 5-4-3-2-1	498.5	0.002 (worst)
OB5 → 2-5-1-4-3	429	0.002331	NS5 → 1-5-2-4-3	455.5	0.002195 (worst)

Step 7 Memorize the best sequence (food source) using Greedy selection to while comparing onlooker bee food source (neighborhood sequence). The memorized best food sources (best sequence) are listed in Table 11.

Table XI: Memorized best food source

Best food source	Fitness value (f(xi))
1-2-3-4-5	0.002713
2-1-4-3-5	0.002628
1-4-3-5-2	0.002528
3-4-5-2-1	0.002366
2-5-1-4-3	0.002331

Step 8 The scout bees explore the food source randomly. The best food sources (best sequence) are the input of the next cycle and the worst food source is replaced from the population.

Step 9 The above procedure is continued until the maximum number of terminations is arrived. After completion of iteration process, we can get a best sequence. It is represented in Table 12.

Table XII: Results of 5 * 2 problems

Final best sequence	TCT	Cmax	Z
1-2-3-4-5	222	515	368.5

For this 5 * 2 problem, high fitness value having low (minimum) objective function value and its sequence are stored as best. The best sequence acts as input sequence for the next cycle.

IV. COMPUTATIONAL OUTCOMES

In this paper, the HABC algorithms are executed in Java environment. HABC algorithms are tested with Taillard SDST benchmark problem instance. The problem ranges from jobs (20, 50, 100 and 200) with varying of machines (5, 10 and 20). Equal importance are given ($\alpha = 0.5$, $\beta = 0.5$) due to the conflicting nature between makespan and total flow time objectives. The Relative Error Percentage (REP) equation is used to calculate the performance value.The average REP value is represented in Table 13.

REP = {(Heuristic solution − Best solution)/Best solution} ×100

Table 13 Average relative error percentage value

Machines	Jobs	HABC1	HABC2	HABC3	HABC4
5	20	13.32	14.91	12.45	10.63
	50	11.52	9.21	10.16	11.87
	100	10.31	8.92	9.28	7.33
	Average	11.72	11.01	10.63	9.94
10	20	10.14	18.37	16.93	17.16
	50	17.16	16.71	12.49	9.92
	100	8.75	17.25	16.15	15.37
	200	13.63	14.18	15.31	11.15
	Average	12.42	16.6275	15.22	13.4
20	20	15.15	14.75	14.56	10.36
	50	13.07	12.84	12.34	13.17
	100	14.35	13.32	14.32	14.67
	200	11.35	12.15	13.65	9.84
	Average	13.48	13.265	13.7175	12.01
	AREP	12.61	13.87	13.42	11.95

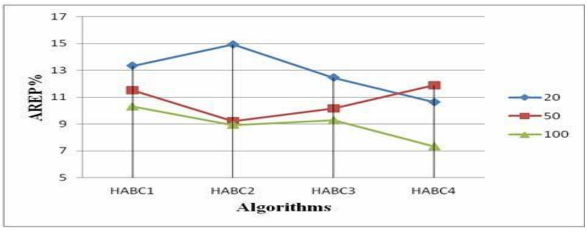

Fig. 2.Five machine problems

From Figure 2, it is clear that for five machine problems, HABC4 produced best results for 20 jobs and 100 jobs. The number of feasible solution is 2 in HABC4.HABC2 produced best results for 50 jobs. The number of feasible solution is 1 HABC2. HABC2 produced worst results for 20 jobs.

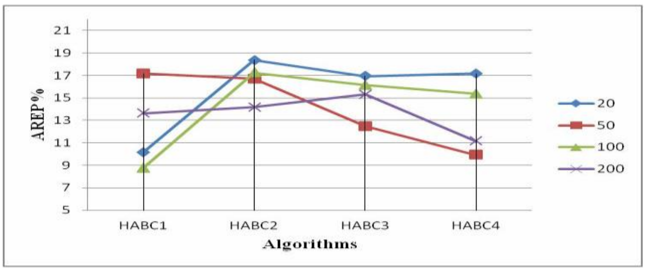

Fig. 3. 10 machine problems

Figure 3 represented ten machine problems average relative error percentage (AREP) value. HABC1 produced best results for 20 jobs and 100 jobs. The number of feasible solution is 2 in HABC1. HABC2 produced worst value for 20 jobs. HABC4 produced best results for 50 jobs and 200 jobs. The number of feasible solution is 2 in HABC4.

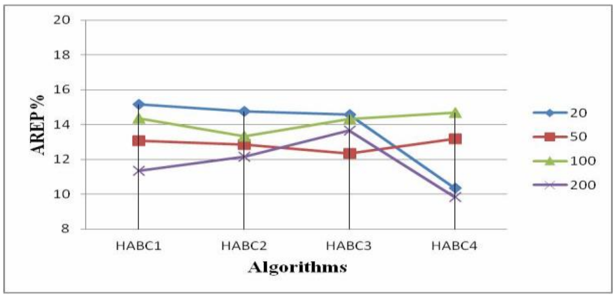

Fig. 4. 20 machine problems

From Figure 4 for 20 machine problems, HABC1 produced worst results for 20 jobs. HABC2 produce best results for 100 jobs and produce one best solution. HABC3 produce best results for 50 jobs and produce one best solution. HABC4 produced best results when compared to HABC1, HABC2 and HABC3 for 20 jobs and 200 jobs. The number of feasible solution is two in HABC4. Totally, two feasible solutions yield from HABC1 and HABC2, one feasible solutions yield from HABC3 and six feasible solutions yield from HABC4. The average REP for hybrid algorithm is shown in Figure 5.

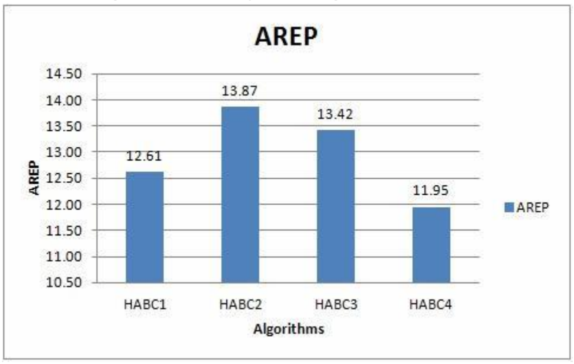

Fig.5. AREP for hybrid algorithms

Statistical test is conducted for validating the results of the HABC algorithms. HABC4 algorithm produces good results compare to other approaches. Using ANOVA method to find out whether the sample is accepted or rejected. The ANOVA table is represented in Table 14.

Table 14 ANOVA Table

Source of variations	Sum of squares	Degrees of freedom	Mean squares	F-ratio
Between samples	SSC = 5.859	V1 = 4 – 1 = 3	MSC = 1.953	FC = 2.20
Within samples	SSE = 34.53	V2 = 12 – 4 = 8	MSE = 4.3	

Sum of squares within the solution is high when compared between the solutions.Calculated F-value (F_C) = 2.20, v_1 = 3, v_2 = 9 and table value (F_T) = 4.07. Therefore,$F_C < F_T$. There is a significant difference while increasing the number of machines and average REP value between the algorithms.

V. CONCLUSIONS

In this work, equal importance is considered in the objectives for SDST flow shop problems. The algorithm performance is represented in terms of average REP. HABC2 produced maximum value of average REP whereas HABC4 produced minimum error value of average relative percentage.

The performance of HABC4 is better when compared to other approaches. Statistical results show that the significant difference is high by increasing the number of machines and average relative error between the algorithms.

REFERENCES

1. BD. Corwin and AO. Esogbue, "Two-machine flow shop scheduling problem with sequence dependent setup times: a dynamic programming approach," Naval Research Logistics, Quarterly, 1974; vol. 2(1),pp. 515–524.
2. S.R. Das, J.N.D. Gupta and B.M. Khumawala, "A savings indexheuristic algorithm for flow shop scheduling with sequence dependent setup times," Journal of Operation Research Society, 1995, vol. 46,pp. 1365–1373.
3. S. Parthasarathy and C. Rajendran,"An experimental evaluation of heuristics for scheduling in a real-life flow shop with sequence-dependent setup times of jobs," International Journal of Production Economics, 1997, vol. 49, pp. 255–263.

4. R.Z. Rios-Mercado and J.F. Bard,"Computational experience with a branch-and cut algorithm for flow shop scheduling with setups," Computers and Operations Research, 1998, vol. 25, pp. 351–366.
5. R.Z. Rios-Mercado and J.F. Bard,"An enhanced TSP-based heuristic for makespan minimization in a flow shop with setup times,"Journal of Heuristics, 1999, vol. 5, pp. 53–70.
6. C. Rajendran and H. Ziegler, "Scheduling to minimize the sum of weighted flow time and weighted tardiness of jobs in a flow shop with sequence-dependent setup times," Journal of Operational Research, 2003, vol. 149, pp. 513–522.
7. R. Ruiz, C. Marotoand J. Alcaraz, "Solving the flowshop scheduling problem with sequence dependent setup times using advanced metaheuristics," European Journal of Operational Research, 2005, vol.165, pp. 34–54.
8. F.T. Tseng, J.N.D. Gupta and E.F. Stafford, "A penalty-based heuristic algorithm for the permutation flow shop scheduling problem with sequence-dependent set-up times," Journal of Operation Research Society, 2005, vol. 57, pp. 541–551.
9. Y. Gajpal, C. Rajendran and H. Ziegler, "An ant colony algorithm for scheduling in flow shops with sequence dependent setup of jobs," International Journal of Advanced Manufacturing Technology, 2006, vol. 30, pp. 416–424.
10. T. Eren and E. Güner, "A bicriteria flow shop scheduling problem with setup times," Applied Mathematics and Computation, 2006, vol. 183, pp. 1292–1300.
11. T. Eren,"A multicriteria flow shop scheduling problem with setup times," Journal of Materials Processing Technology, 2007, vol.186, pp. 60–65.
12. X. Wang and T.C.E. Cheng, "Heuristics for two-machine flow shop scheduling with setup times and an availability constraint," Computer Operation Research, 2007, vol. 34,pp. 152– 162.
13. A. Allahverdi and M.Y. Kovalyov, "A survey of scheduling problems with setup times or costs," European Journal of Operation Research, 2008, vol.187, pp. 985–1032.
14. R. Ruiz and T. Stutzle, "An iterated greedy heuristic for the sequence dependent setup times flow shop problem with makespan and weighted tardiness objectives," European Journal of Operational Research, 2008, vol.187, pp. 1143–1159.
15. S.A. Mansouri and S. Hendizadeh, "Bicriteria scheduling of a two-machine flow shop with sequence-dependent setup times," International Journal of Advanced Manufacturing Technology, 2009, vol. 40, pp. 1216–1226.
16. A. Dhingra and P. Chandna, "Hybrid genetic algorithm for multicriteria scheduling with sequence dependent set up time," International Journal of Engineering, 2009, vol.3, pp. 10–520.
17. A. Dhingra and P. Chandna, "A bi-criteria M-machine SDST flow shop scheduling using modified heuristic genetic algorithm," International Journal of Engineering, Science and Technology, 2010, vol. 2,pp. 216–225.
18. T. Eren, "A bicriteria m-machine flowshop scheduling with sequence-dependent setup times," Applied Mathematical Modelling, 2010, vol. 34, pp. 284–293.
19. A. Kumar, N. Hooda and A.K. Dhingra, "Minimization of total weighted tardiness and makespan for SDST flow shop scheduling using genetic algorithm," International Journal of Engineering Sciences, 2012, vol.1, pp. 2277–9698.
20. N. Hooda and A.K. Dhingra, "Bi-criteria SDST flow shop scheduling using simulated annealing," International Journal of Engineering Sciences, 2012, vol.1, pp. 2277–9698.
21. R. Vanchipuraand R. Sridharan, "Impact of varying levels of setup time on the performance of algorithms for scheduling a flow shop with sequence dependent setup time," International Journal of Services and Operations Management, 2013, vol.14: pp.421–444.
22. R. Vanchipura and R. Sridharan, "Development and analysis of hybrid genetic algorithms for flow shop scheduling with sequence dependent setup time," International Journal Services and Operations Management, 2014, vol.17, pp. 168–193.
23. S. Hatami, R. Ruiz and C.A. Romano, "Heuristic and metaheuristic for the distributed assembly permutation flowshop scheduling problem with sequence dependent setup times," International Journal of Production Economics, 2015, vol.169, pp. 76–88.
24. D. Satyanarayana and M. Pramiladevi, "Multi-criteria M-machine SDST flow shop scheduling using modified heuristic genetic algorithm," International Journal of Industrial and Systems Engineering, 2016, vol. 22, pp. 409–422.
25. B.J. Jeong and B.J.S. Shim,"Heuristic algorithms for two-machine re-entrant flowshop scheduling problem with jobs of two classes,". Journal of Advanced Mechanical Design, Systems and Manufacturing, 2017, vol.11, pp. 1-14.
26. D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numeric function optimization: artificial bee colony (ABC) algorithm," International Journal of Global Optimization, 2007, vol. 39: pp. 459–471.
27. F. Kang, J. Li and Q. Xu, "Structural inverse analysis by hybrid simplex artificial bee colony algorithms," Computers and Structures, 2009, vol. 87, pp. 861–870.
28. D. Karaboga and C. Ozturk, "Neural networks training by artificial bee colony algorithm on pattern classification," Neural Network World, 2009, vol.19, pp. 279–292.
29. G. Zhu and S. Kwong, "Gbest-guided artificial bee colony algorithm for numerical function optimization," Applied Mathematics and Computation, 2010, vol.217, pp. 3166–3173.
30. J. Li, Q. Pan and K. Gao, "Pareto-based discrete artificial bee colony algorithm for multi-objective flexible job shop scheduling problems," International Journal of Advanced Manufacturing Technology, 2011, vol. l55, pp. 159–1169.
31. R. Akbaria, V. Zeighamib and K. Ziaratia, "Artificial bee colony for resource constrained project scheduling problem," International Journal of Industrial Engineering Computations, 2011, vol. 2(1), pp. 45–60.
32. M.F. Tasgetiren, Q. Pan, P.N. Suganthan and A. Chen, "A discrete artificial bee colony algorithm for the total flowtime minimization in permutation flow shops," Information Sciences, 2011, vol. 181, pp. 3459–3475.
33. Y.F. Liu and S.Y. Liu, "A hybrid discrete artificial bee colony algorithm for permutation flow shop scheduling problem," Applied Soft Computing, 2013, vol.13, pp. 1459–1463.
34. O. Bulut and M.F. Tasgetiren, "An artificial bee colony algorithm for the economic lot scheduling problem," International Journal of Production Research, 2014, vol.52, pp. 1150–1170.
35. M. Basti and M. Sevkli, "An artificial bee colony algorithm for the p-median facility location problem," International Journal of Metaheuristics, 2015, vol. 4, pp. 91–113.
36. Y.Y. Han, D. Gong and X. Sun, "A discrete artificial bee colony algorithm incorporating differential evolution for the flow-shop scheduling problem with blocking," Engineering Optimization, 2015, vol. 47, pp. 927–946.
37. E. Caniyilmaz, B. Benli and M.S. Ilkay, "An artificial bee colony algorithm approach for unrelated parallel machine scheduling with processing set restrictions, job sequence-dependent setup times, and due date," International Journal of Advanced Manufacturing Technology, 2015, vol. 77, pp. 2105–2115.

AUTHORS PROFILE

Dr. R. SANJEEV KUMAR is currently working as an Associate Professor in Swarnandhra College of Engineering & Technology, Andhra Pradesh. He has completed BE(Mechanical), ME(CAD/CAM) & Ph.D in Anna University Chennai. He has 9 Years Teaching Experience. He is the Recognized Research Supervisor for guiding Ph.D scholar in Anna University Chennai. He published 4 International journal papers and published 3 International Conference papers. Life membership in ISTE & IAENGG. Reviewer in Journal of Industrial & Production Engineering & other reputed Journals.

Dr. G ROBERT SINGH is currently working as an Associate Professor in swarnandhra. College of engineering. He has completed BE(Mechanical)., ME(Manufacturing) & Ph.D. He has 10 Years Teaching Experience. He has published 5 International journal papers and 2 International Conference papers. Life Membership in ISTE.

Dr. N MATHAN KUMAR is currently working as an Associate Professor in Akshaya College of Engineering & Technology, Coimbatore, Tamil Nadu. He has completed BE(Mechanical), ME(CAD/CAM) & Ph.D (IIT– ISM in Dhanbad. He has 7 Years Teaching Experience. He published 11 journal papers. He is in life time member of various Engineering Associations and Being journal Editor and Reviewer in various reputed journals.