Journal of Theoretical and Applied Information Technology
31st January 2019. Vol.97. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

THE SEARCH OF BALANCE BETWEEN DIVERSIFICATION AND INTENSIFICATION IN ARTIFICIAL BEE COLONY ALGORITHM TO SOLVE JOB SHOP SCHEDULING PROBLEM

[1] JEBARI HAKIM, [2] RAHALI EL AZZOUZI SAIDA, [3] SAMADI HASSAN, [4] SIHAM REKIEK

[1,2,3]National School of Applied Sciences of Tangier, University Abdelmalek Essaâdi, Morocco

[4]Faculty of Science and Technique of Tangier, University Abdelmalek Essaâdi, Morocco

E-mail: [1]hjebari1@yahoo.fr, [2]rahali_elazzouzi@yahoo.fr, [3]ha.samadi@gmail.com, [4]srekiek@gmail.com

ABSTRACT

Several intelligent theories have been proposed to solve job shop scheduling problems with the target of minimizing makespan such as hybrid metaheuristics. The core advantage of hybridization is to improve the effectiveness of the algorithm leading to better quality of solutions. The paper establishes seven hybrid methods namely HABCGA according to sixteen configurations (four crossover operators: PMX, OX, CX and PBX, and four mutation operators: swap, inversion, insertion and displacement) to solve job shop scheduling problems with the objective of minimizing makespan and to test these algorithms on 250 Benchmark instances to evaluate the performance and to find the balance that can lead to the optimal performance of the hybrid ABC. The results indicate how the multiple hybridizations, the crossover operator type, and the mutation operator type affect in a positive way the balance between diversification and intensification in artificial bee colony algorithm to solve job shop scheduling problem.

Keywords: *Scheduling, Job Shop, Multiple Hybridization, ABC, GA.*

1. INTRODUCTION

The scheduling problems are often combinatorial optimization problems of NP-difficult type. Their resolution requires methods dedicated to their degree of difficulty. Therefore, many different metaheuristics are established for this problem. These metaheuristics generate optimal or near-optimal solutions. Among the metaheuristics approaches, hybrid and Adaptive techniques provide very good solutions.

The design of metaheuristics has classified by Talbi [1] in two categories:

- ✓ In the case of low-level, a specified function of metaheuristic is substituted by another method. In the case of high-level, the different methods are unattached.
- ✓ In the case of the relay, a set of methods is executed one after another, each using the output of the previous as its inputs, acting in pipeline mode. In the case of the teamwork, represents the cooperation of optimization models.

Diversification makes possible to explore the search zone more efficiently, and it generate solutions with enough diversity. On the other hand, intensification uses any information obtained from the specific problem to produce new solutions that are better than existing solutions.

A good balance between diversification and intensification is essential to lead to the optimal performance of an algorithm; it can depend on many elements such as the working mechanism of an algorithm, its parameters and control of these parameters.

Many authors considered basic theory while solving job shop scheduling problem [2].

The mathematical formulation of job shop scheduling problem notation is as follows:

$i : Index\ of\ job, i = 1,2,3, \dots, n$

$j : Index\ of\ machine, j = 1,2,3, \dots, m$

$k : Index\ of\ operation, k = 1,2,3, \dots, m$

$n : Total\ number\ of\ jobs$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

m : Total number of machines C_{mac} Makespan

M_j : The j^{th} machine, $j = 1,2,3,\dots,m$

J_i : The i^{th} job, $i = 1,2,3,\dots,n$

O_{ikj} : The k^{th} operation of job J_i operated in machine M_j

P_{ikj} : Processing time of operation O_{ikj}

t_{ikj} : Completion time of operation O_{ikj} on machine M_j for each job J_i

The problem can be described as an N job M machine JSSP by simple Eqs. (1) and (2) as follows:

$$\begin{cases} Min\ C_{max} = max_{ikj}\ \{t_{ikj}\} & (1) \\ t_{i,k-1,h} + P_{ikj} \le t_{ij}, \vee\ i,k,h,j & (2) \\ t_{ikj} \ge 0, \quad \vee\ i,k,j & \end{cases}$$

The Equation (1) explain the Makespan minimization. The Equation (2) refer to the operation precedence constraint; the $(k-1)^{th}$ operation of job i must be executed before the $(k)^{th}$ operation of the same job.

In the last decade, authors has extended towards flexible job shop scheduling problems in which more than one machine for performing each operation are considered. It present more complexity.

The flexible job shop scheduling problems is a continuation of classical job shop scheduling problem and it is more complex, because each operation can be treated on any machine from a specified group.

For several years, many researchers have developed various methods to solve job shop-scheduling problem: simulated annealing [3, 4], genetic Algorithm [5, 6], Search Tabu [7, 8], Ant Colony Optimization [9, 10], Neural Network [11], Shifting Bottleneck procedure [12, 13], guided local search [14], GRASP [15] and propagation of constraints [16]. Table 1 summarizes some contributions to solve job shop scheduling problem.

Table 1: The Proposed Methods For Solving Job Shop Scheduling Problem In The Literature.

Article	Proposed Method
[17]	GA
[18]	GA
[19]	GA
[20]	GA
[21]	GA
[22]	GA
[23]	GA
[24]	GA
[25]	GA
[26]	ABC
[27]	Co-evolutionary GA
[28]	Hybrid GA
[29]	Hybrid GA
[30]	PSO
[31]	ABC
[32]	ABC
[33]	Hybrid GA with scheduling rules
[34]	Hybrid ACO with TS
[35]	Hybrid GA with local search
[36]	Hybrid TS with SA
[37]	Hybrid PSO with TS
[38]	Hybrid GA with ACO
[39]	Hybrid ACO with knowledge model
[40]	Hybrid PSO with TS
[41]	Hybrid GA with ACO
[42]	Hybrid GA with local search
[43]	Hybrid GA with SA
[44]	Hybrid GA with heuristics
[45]	Hybrid PSO with local search
[46]	Hybrid AIS with SA
[47]	Hybrid DE algorithm with local search
[48]	Hybrid GA with VNS with affinity function
[49]	Hybrid Gravitational search algorithm with PSO
[50]	Hybrid Harmony search with large neighborhood search
[51]	Hybrid HHS with local search
[52]	Hybrid Memetic algorithm with critical path method

However, to the authors' knowledge, very few publications are available in the literature that studied seven hybrid artificial bee colony algorithms according to sixteen configurations and that evaluated the performance of the proposed methods using a set of 250 benchmark instances and that compared the obtained results for finding the balance that may lead to the optimal performance of the hybrid ABC.

The objective of this paper is as follows:

- ✓ To do a study of seven hybrid ABC with GA according to sixteen configurations.
- ✓ To make tests on 250 Benchmark instances from the classical OR-library.
- ✓ To find the balance that can lead to the optimal performance of the hybrid ABC to solve the job shop-scheduling problem.

The rest of the paper is structured into four sections:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The Fundamentals of ABC and GA will be described in Section 2. In Section 3, the authors presented the implementation of the proposed methods. The results and discussion are showed in section 4. In Section 5, the conclusion is presented.

2. FUNDAMENTALS OF THE USED METHODS

2.1 Fundamentals of Artificial Bee Colony Algorithm

Artificial bee colony algorithm is established by Karaboga [53]; Karaboga and Basturk [54] described and evaluated its performance for optimization of numerical problems. The swarm-based metaheuristic algorithm is inspired by the intelligent comportment of honeybees, that is, foraging comportment of honeybee colonies.

Four phases forms ABC: Initialization bee phase, employed bees phase associate with specific food sources, onlooker bees phase look at the dance of employed, and scout bees phase search arbitrarily food sources.

The stopping criterion of the Artificial Bee Colony defines the maximum number of cycles that a food source can keep without improving before being replaced.

The standard pseudo-code that describes the artificial bee colony algorithm is as follows:

```
Initialization
Repeat
Employed Bees Phase
Onlooker Bees Phase
Scout Bees Phase
Remember the best solution attained
Until the stop criterion.
```

2.2 Fundamentals of Genetic Algorithm

Genetic Algorithm is a classic algorithm, which is a bio-inspired and population-based technology for complex problems, developed by [55]. The inspiration of GA is to upgrade the survival and duplication of the best suitable individuals to the environment. Five phases forms GA: initial population phase, fitness function phase, selection phase, crossover phase, and mutation phase.

The standard pseudo-code that describes the Genetic Algorithm is as follows:

```
Generate the initial population
Compute fitness
Repeat
Selection
Crossover
Mutation
Compute fitness
Until population has converged
```

The goal of crossover operator is to generate new individuals in the process of the algorithm. It is affects the exploration and the convergence.

In the literature, numerous crossover operators have been suggested in Job shop scheduling such as the subsequence-exchange operator, the PMX, the OX, the linear order crossover, the PBX, the uniform crossover, the CX, the extrapolation-directed, the partial schedule exchange crossover [56-58] and extended precedence preservative crossover, the order-based crossover, the job based order crossover.

In this paper, the authors take in consideration four crossover operators: PMX, OX, CX and PBX.

The PMX [59-61]: Two crossover points are selected randomly and string between these points are exchanged to form new offspring's to create new chromosomes of the next generation. Then the inverse replacement is applied to the job indices occurring a second time outside the chosen segment. It tries to keep the positions of the jobs in the offspring when copying them from the parents.

The OX [62] designed for the TSP, determines randomly two cut points.

The CX [62] determines a subgroup of jobs occupying the same set of positions in both parents.

The procedure that describes CX is as follows:

1. Discover the cycle that is defined the corresponding positions of job between parents.
2. Copy the job in the cycle to a child with the corresponding positions of one parent.
3. Define the unfinished positions from the child by deleting those jobs, which are already in the cycle from the other parent.
4. Fulfill the child with the unfinished position.

The PBX [63, 64]: Selects a subgroup of positions in the first parent and copies the jobs at these positions into the offspring. The remaining positions are filled with the missing jobs appearing in the same relative order as in the second parent.

The mutation operator in the GA is similar to the biological mutation, which is used to conserve genetic diversity. Different mutation operators have been suggested in the literature such as swap, inversion, insertion and displacement mutation.

Swap mutation [65]: Two genes were randomly selected and replaced into the mutation process.

Insertion mutation [66, 67]: which the procedure is as follows:

1. Choose randomly a gene.
2. Insert this gene into a randomly designated position.

Inversion mutation [68, 69]: It is based on reversing the order of the genes between two randomly designated points in the chromosome.

Displacement mutation [70]: It selects randomly a substring and inserts it into a position that is also designated randomly. All these displacements take place within the selected chromosome.

3. THE PROPOSED METHODS

The main objective of ABC hybridization with GA is to overcome the limitations of ABC and to solve job shop scheduling problem. The authors developed a new approach of hybridization of ABC with GA, this approach consist to use in the employed bees phase or/and in the onlooker bees phase or/and in the scout bees phase, the crossover operator and the mutation operator. This is illustrated in Table 2.

Table 2: The Structure Of Hybrid ABC With GA.

	ABC		
Hybrid ABC GA	Employed bee phase	Onlooker bee phase	Scout bee phase
HABCGA1	crossover, mutation operations		
HABCGA2	crossover, mutation operations	crossover, mutation operations	
HABCGA3	crossover, mutation operations	crossover, mutation operations	crossover, mutation operations
HABCGA4		crossover, mutation operations	
HABCGA5		crossover, mutation operations	crossover, mutation operations
HABCGA6			crossover, mutation operations
HABCGA7	crossover, mutation operations		crossover, mutation operations

Therefore, the hybridization of ABC with GA is according to the configurations given in Table 3.

Table 3: The Configurations Of Hybridization Of ABC With GA.

Configuration	Crossover	Mutation
Configuration 1	PMX	Inversion
Configuration 2	OX	Inversion
Configuration 3	CX	Inversion
Configuration 4	PBX	Inversion
Configuration 5	PMX	Insertion
Configuration 6	OX	Insertion
Configuration 7	CX	Insertion
Configuration 8	PBX	Insertion
Configuration 9	PMX	Swap
Configuration 10	OX	Swap
Configuration 11	CX	Swap
Configuration 12	PBX	Swap
Configuration 13	PMX	Displacement
Configuration 14	OX	Displacement
Configuration 15	CX	Displacement
Configuration 16	PBX	Displacement

As a result, the procedure of the hybrid ABC with GA (HABCGA3) according to the configuration 14 can be found in Figure 1.

4. RESULTS AND DISCUSSION

The proposed methods were programmed in Eclipse. The PC configuration is as follows: Processor: Intel Core i7, OS: Windows 8.1, CPU speed: 2.10 GHz, RAM: 6 GB. There were tested on

250 Benchmark instances from the classical OR-library.

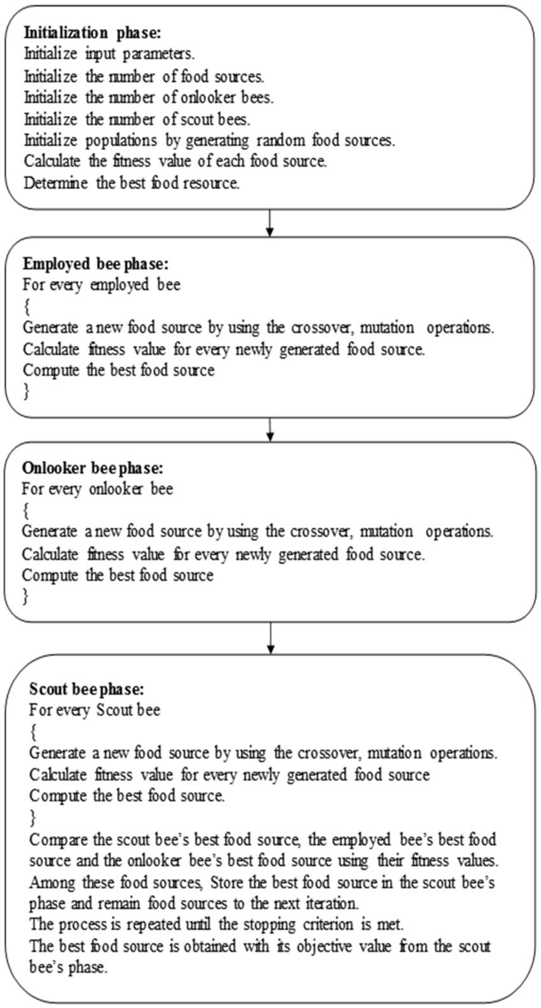

Figure 1: The Procedure Of HABCGA3 Method According To The Configuration 14.

The benchmark instances are as follows:

- ✓ FT instances [71] is a group of three instances of the formats (6, 6), (10, 10), (20, 5).
- ✓ LA instances [72] is a group of 40 instances including five instances of the formats (n, m) ∈ {(10, 5), (15, 5), (20, 5), (10, 10), (15, 10), (20, 10), (30, 10), (15, 15)}.
- ✓ ORB instances [73] is a group of 10 instances of the format (n, m) = (10, 10).
- ✓ SWV instances [74] is a group of 20 instances including 20 instances for each of the formats (n, m) ∈ {(20, 10), (20, 15), (50, 10))}.
- ✓ ABZ instances [75] is a set of five problems of two different sizes (10×10), (20×15).
- ✓ YN instances [76] is group of four instances of the format (n, m) = (20, 20).
- ✓ TA instances [77] is group of 80 instances including 10 instances for each of the combinations (n, m) ∈ {(15, 15), (20, 15), (20, 20), (30, 15), (30, 30), (50, 15), (50, 20), (100, 20)}.
- ✓ DMU instances [78] is a group of 80 instances including 10 instances for each of the formats (n, m) ∈ {(20, 15), (20, 20), (30, 15), (30, 20), (40, 15), (40, 20), (50, 15), (50, 20)}.
- ✓ CAR instances [79] is a group of 8 problems of 8 different sizes: (11×5), (13×4), (12×5), (14×4), (10×6), (8×9), (7×7), (8×8).

The GAP is obtained by a particular method from the optimum or the global makespan is used to make the makespan comparisons. The Equation (3) that describe the relative deviation GAP is as follows:

$$\%\,\text{GAP} = \frac{\text{Makespan Found} - \text{BKS}}{\text{BKS}} \qquad (3)$$

The obtained results are compared with other best-known solutions obtained by using other methods.

The global results of the proposed methods are shown in Table 4. The results show that the proposed method HABCGA3 in all configurations is given 100% results equal to the best-known solution in all benchmark instances; this is illustrated in Figure 2. In this hybridization type, the proposed method is given the best results independent of the crossover operator type and the mutation operator type in all benchmark instances.

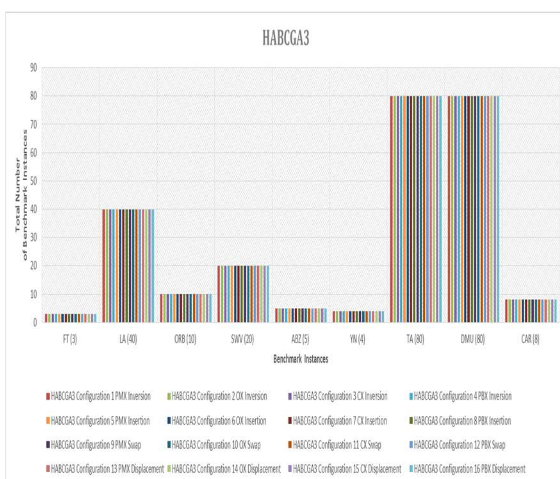

Figure 2: The Obtained Result Of The Proposed Method HABCGA3 In All Configurations.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The results show that all proposed methods in all configurations are produced 100% results equal to the best-known solution in FT (3), ORB (10), ABZ (5), YN (4) and CAR (8) instances as shown in Figure 3.

These methods are given the best results independent of the hybridization type, the crossover operator type and the mutation operator type in FT (3) , ORB (10), ABZ (5), YN (4) and CAR (8) instances.

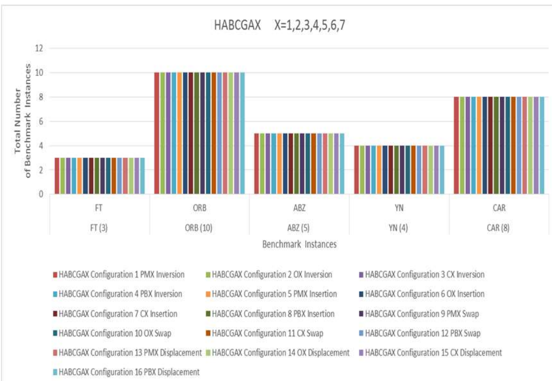

Figure 3: The Obtained Results Of All Proposed Methods In FT (3), ORB (10), ABZ (5), YN (4) and CAR (8) Instances.

The results show that only tree proposed methods HABCGA2, HABCGA5 and HABCGA7 in all configurations are given 100% results equal to the best-known solution in LA (40) and SWV (20) instances as shown in Figure 4. In these hybridization type, the proposed methods are given the best results independent of the crossover operator type and the mutation operator type in LA (40) and SWV (20) instances.

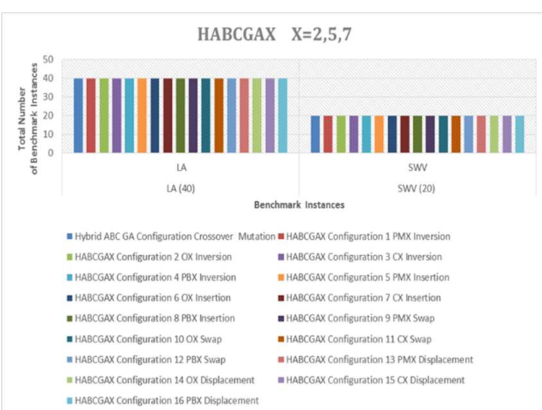

Figure 4: The Obtained Results Of The Proposed Methods HABCGA2, HABCGA5 And HABCGA7 In All Configurations To Solve LA (40) And SWV (20) Instances.

The results show that all proposed methods in configuration 2, configuration 4, configuration 6, configuration 8, configuration 10, configuration 12, configuration 14 and configuration 16 (that used the OX operator with the inversion mutation or with the insertion mutation or with the swap mutation or with the displacement mutation and that used the PBX operator with the inversion mutation or with the insertion mutation or with the swap mutation or with the displacement mutation), are given 100% results equal to the best-known solution in all benchmark instances. These are presented in Figure 5.

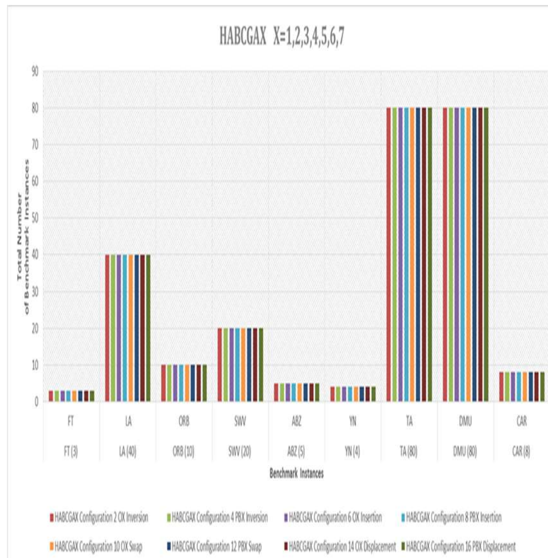

Figure 5: The Obtained Results Of All Proposed Methods In Configuration 2, Configuration 4, Configuration 6, Configuration 8, Configuration 10, Configuration 12, Configuration 14 And Configuration 16.

As follows from Figure 5:

- ✓ These proposed methods are given the best results independent of the hybridization type in all benchmark instances.
- ✓ These proposed methods that used the OX and the PBX crossover operators are produced better results than the proposed methods that used the CX and the MPX crossover operators.
- ✓ These proposed methods that used the OX independent of the mutation operator type are given 100% results equal to the best-known solution in all benchmark instances.
- ✓ These proposed methods that used the PBX independent of the mutation operator type are given 100% results equal to the best-known solution in all benchmark instances.

Journal of Theoretical and Applied Information Technology
31st January 2019. Vol.97. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 4: The Results Of Hybridization Of ABC With GA In All Configurations.

			Benchmark Instances	FT (3)		LA (40)		ORB (10)		SWV (20)		ABZ (5)		YN (4)		TA (80)		DMU (80)		CAR (8)		Total Number of Benchmark	
				FT		LA		ORB		SWV		ABZ		YN		TA		DMU		CAR			
Hybrid ABC GA	Configuration	Crossover	Mutation	3		40		10		20		5		4		80		80		8		250	
HABCGA1	Configuration 1	PMX	Inversion	3	100%	32	80%	10	100%	18	90%	5	100%	4	100%	72	90%	76	95%	8	100%	228	91%
HABCGA1	Configuration 2	OX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA1	Configuration 3	CX	Inversion	3	100%	35	88%	10	100%	20	100%	5	100%	4	100%	71	89%	71	89%	8	100%	227	91%
HABCGA1	Configuration 4	PBX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA1	Configuration 5	PMX	Insertion	3	100%	33	83%	10	100%	20	100%	5	100%	4	100%	75	94%	70	88%	8	100%	228	91%
HABCGA1	Configuration 6	OX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA1	Configuration 7	CX	Insertion	3	100%	35	88%	10	100%	18	90%	5	100%	4	100%	70	88%	72	90%	8	100%	225	90%
HABCGA1	Configuration 8	PBX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA1	Configuration 9	PMX	Swap	3	100%	37	93%	10	100%	17	85%	5	100%	4	100%	73	91%	71	89%	8	100%	228	91%
HABCGA1	Configuration 10	OX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA1	Configuration 11	CX	Swap	3	100%	34	85%	10	100%	15	75%	5	100%	4	100%	75	94%	74	93%	8	100%	228	91%
HABCGA1	Configuration 12	PBX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA1	Configuration 13	PMX	Displacement	3	100%	31	78%	10	100%	15	75%	5	100%	4	100%	78	98%	74	93%	8	100%	228	91%
HABCGA1	Configuration 14	OX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA1	Configuration 15	CX	Displacement	3	100%	38	95%	10	100%	16	80%	5	100%	4	100%	71	89%	72	90%	8	100%	227	91%
HABCGA1	Configuration 16	PBX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 1	PMX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 2	OX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 3	CX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	79	99%	8	100%	249	100%
HABCGA2	Configuration 4	PBX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 5	PMX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 6	OX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%

HABCGA2	Configuration 7	CX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	78	98%	80	100%	8	100%	248	99%
HABCGA2	Configuration 8	PBX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 9	PMX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 10	OX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 11	CX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	79	99%	79	99%	8	100%	248	99%
HABCGA2	Configuration 12	PBX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 13	PMX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 14	OX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA2	Configuration 15	CX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	79	99%	78	98%	8	100%	247	99%
HABCGA2	Configuration 16	PBX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 1	PMX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 2	OX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 3	CX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 4	PBX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 5	PMX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 6	OX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 7	CX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 8	PBX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 9	PMX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 10	OX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 11	CX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 12	PBX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 13	PMX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 14	OX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 15	CX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA3	Configuration 16	PBX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA4	Configuration 1	PMX	Inversion	3	100%	32	80%	10	100%	18	90%	5	100%	4	100%	72	90%	70	88%	8	100%	222	89%
HABCGA4	Configuration 2	OX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA4	Configuration 3	CX	Inversion	3	100%	35	88%	10	100%	15	75%	5	100%	4	100%	71	89%	71	89%	8	100%	222	89%
HABCGA4	Configuration 4	PBX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%

HABCGA4	Configuration 5	PMX	Insertion	3	100%	33	83%	10	100%	15	75%	5	100%	4	100%	70	88%	70	88%	8	100%	218	87%
HABCGA4	Configuration 6	OX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA4	Configuration 7	CX	Insertion	3	100%	31	78%	10	100%	14	70%	5	100%	4	100%	70	88%	72	90%	8	100%	217	87%
HABCGA4	Configuration 8	PBX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA4	Configuration 9	PMX	Swap	3	100%	33	83%	10	100%	13	65%	5	100%	4	100%	71	89%	71	89%	8	100%	218	87%
HABCGA4	Configuration 10	OX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA4	Configuration 11	CX	Swap	3	100%	30	75%	10	100%	14	70%	5	100%	4	100%	75	94%	74	93%	8	100%	223	89%
HABCGA4	Configuration 12	PBX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA4	Configuration 13	PMX	Displacement	3	100%	31	78%	10	100%	15	75%	5	100%	4	100%	73	91%	71	89%	8	100%	220	88%
HABCGA4	Configuration 14	OX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA4	Configuration 15	CX	Displacement	3	100%	32	80%	10	100%	16	80%	5	100%	4	100%	71	89%	72	90%	8	100%	221	88%
HABCGA4	Configuration 16	PBX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA5	Configuration 1	PMX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	74	93%	74	93%	8	100%	238	95%
HABCGA5	Configuration 2	OX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA5	Configuration 3	CX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	70	88%	71	89%	8	100%	231	92%
HABCGA5	Configuration 4	PBX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA5	Configuration 5	PMX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	75	94%	73	91%	8	100%	238	95%
HABCGA5	Configuration 6	OX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA5	Configuration 7	CX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	71	89%	70	88%	8	100%	231	92%
HABCGA5	Configuration 8	PBX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA5	Configuration 9	PMX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	75	94%	73	91%	8	100%	238	95%
HABCGA5	Configuration 10	OX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA5	Configuration 11	CX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	72	90%	68	85%	8	100%	230	92%
HABCGA5	Configuration 12	PBX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA5	Configuration 13	PMX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	74	93%	76	95%	8	100%	240	96%
HABCGA5	Configuration 14	OX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA5	Configuration 15	CX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	70	88%	71	89%	8	100%	231	92%
HABCGA5	Configuration 16	PBX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA6	Configuration 1	PMX	Inversion	3	100%	33	83%	10	100%	17	85%	5	100%	4	100%	74	93%	74	93%	8	100%	228	91%
HABCGA6	Configuration 2	OX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

HABCGA6	Configuration 3	CX	Inversion	3	100%	34	85%	10	100%	19	95%	5	100%	4	100%	71	89%	73	91%	8	100%	227	91%
HABCGA6	Configuration 4	PBX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA6	Configuration 5	PMX	Insertion	3	100%	33	83%	10	100%	18	90%	5	100%	4	100%	76	95%	71	89%	8	100%	228	91%
HABCGA6	Configuration 6	OX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA6	Configuration 7	CX	Insertion	3	100%	33	83%	10	100%	18	90%	5	100%	4	100%	70	88%	74	93%	8	100%	225	90%
HABCGA6	Configuration 8	PBX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA6	Configuration 9	PMX	Swap	3	100%	36	90%	10	100%	15	75%	5	100%	4	100%	74	93%	71	89%	8	100%	226	90%
HABCGA6	Configuration 10	OX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA6	Configuration 11	CX	Swap	3	100%	32	80%	10	100%	17	85%	5	100%	4	100%	73	91%	74	93%	8	100%	226	90%
HABCGA6	Configuration 12	PBX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA6	Configuration 13	PMX	Displacement	3	100%	31	78%	10	100%	17	85%	5	100%	4	100%	75	94%	74	93%	8	100%	227	91%
HABCGA6	Configuration 14	OX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA6	Configuration 15	CX	Displacement	3	100%	38	95%	10	100%	17	85%	5	100%	4	100%	71	89%	71	89%	8	100%	227	91%
HABCGA6	Configuration 16	PBX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA7	Configuration 1	PMX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	77	96%	78	98%	8	100%	245	98%
HABCGA7	Configuration 2	OX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA7	Configuration 3	CX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	75	94%	74	93%	8	100%	239	96%
HABCGA7	Configuration 4	PBX	Inversion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA7	Configuration 5	PMX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	76	95%	79	99%	8	100%	245	98%
HABCGA7	Configuration 6	OX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA7	Configuration 7	CX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	72	90%	77	96%	8	100%	239	96%
HABCGA7	Configuration 8	PBX	Insertion	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA7	Configuration 9	PMX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	74	93%	8	100%	244	98%
HABCGA7	Configuration 10	OX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA7	Configuration 11	CX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	77	96%	74	93%	8	100%	241	96%
HABCGA7	Configuration 12	PBX	Swap	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA7	Configuration 13	PMX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	78	98%	77	96%	8	100%	245	98%
HABCGA7	Configuration 14	OX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%
HABCGA7	Configuration 15	CX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	76	95%	72	90%	8	100%	238	95%
HABCGA7	Configuration 16	PBX	Displacement	3	100%	40	100%	10	100%	20	100%	5	100%	4	100%	80	100%	80	100%	8	100%	250	100%

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The results show that all the proposed methods in configuration 1, configuration 3, configuration 5, configuration 7, configuration 9, configuration 11, configuration 13 and configuration 15 (that used the PMX operator with the inversion mutation or with the insertion mutation or with the swap mutation or with the displacement mutation and that used the CX operator with the inversion mutation or with the insertion mutation or with the swap mutation or with the displacement mutation) are given 100% results equal to the best-known solution in FT (3), ORB (10), ABZ (5), YN (4) and CAR (8) instances as shown in Figure 6.

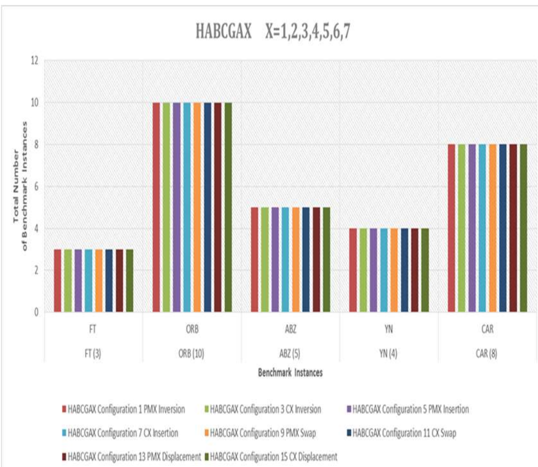

Figure 6: The Obtained Results Of All Proposed Methods In Configuration 1, Configuration 3, Configuration 5, Configuration 7, Configuration 9, Configuration 11, Configuration 13 And Configuration 15 To Solve FT (3), ORB (10), ABZ (5), YN (4) And CAR (8) Instances.

As follows from the Figure 6:

- ✓ These proposed methods are given the best results independent of the hybridization type in FT (3), ORB (10), ABZ (5), YN (4) and CAR (8) instances.
- ✓ These proposed methods that used the PMX independent of the mutation operator type are given 100% results equal to the best-known solution in FT (3), ORB (10), ABZ (5), YN (4) and CAR (8) instances.
- ✓ These proposed methods that used the CX independent of the mutation operator type are given 100% results equal to the best-known solution in FT (3), ORB (10), ABZ (5), YN (4) and CAR (8) instances.

It has been found that only the proposed method HABCGA2 in configuration 1, configuration 5, configuration 9 and configuration 13 is given 100% results equal to the best-known solution in LA (40), SWV (20), TA (80) and DMU (80) instances as shown in Figure 7.

In this hybridization type, the proposed method HABCGA2 that used the PMX independent of the mutation operator type is given 100% results equal to the best-known solution in LA (40), SWV (20), TA (80) and DMU (80) instances.

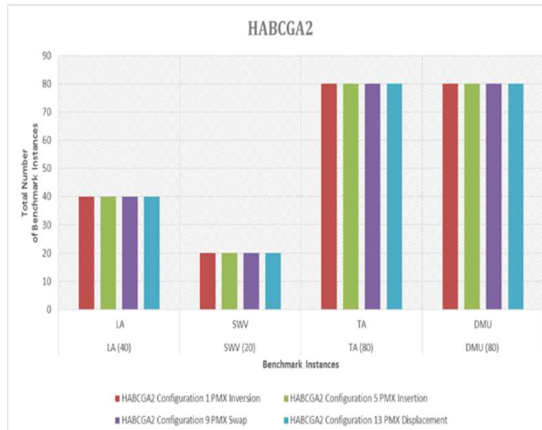

Figure 7: The Obtained Results Of The Proposed Method HABCGA2 In Configuration 1, Configuration 5, Configuration 9 And Configuration 13 To Solve LA (40), SWV (20), TA (80) And DMU (80) Instances.

The ranking of the proposed methods in terms of performance is:

- ✓ HABCGA3 is better than HABCGA2.
- ✓ HABCGA2 is better than HABCGA7.
- ✓ HABCGA7 is better than HABCGA5.
- ✓ HABCGA5 is better than HABCGA1.
- ✓ HABCGA1 is better than HABCGA6.
- ✓ HABCGA6 is better than HABCGA4.

This is illustrated in Figure 8 and in the Table 5.

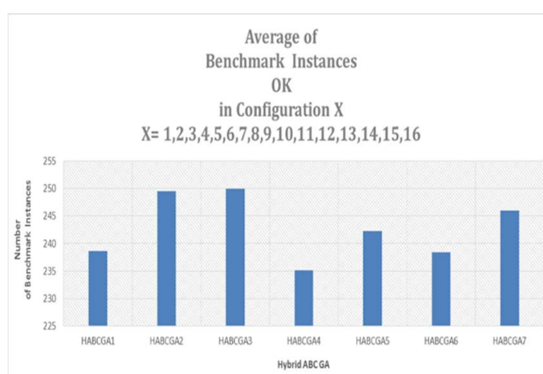

Figure 8: The Ranking Of The Proposed Methods.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 5: The Ranking Of The Proposed Methods According To The Hybridization

Ranking	Hybrid ABC GA	ABC			Total Number of Benchmark Instances	
		Employed bees phase	Onlooker bees phase	Scout bees phase		
1	HABCGA3	GA	GA	GA	250.00	100.00%
2	HABCGA2	GA	GA		249.50	99.80%
3	HABCGA7	GA		GA	246.00	98.40%
4	HABCGA5		GA	GA	242.31	96.93%
5	HABCGA1	GA			238.69	95.48%
6	HABCGA6			GA	238.38	95.35%
7	HABCGA4		GA		235.06	94.03%

5. DISCUSSION

In this paper, the authors study seven hybrid methods to solve job-shop scheduling problem.

The authors' attention was concentrated not only on studies and on tests of these hybrid methods on 250 Benchmark instances but also on their performance evaluation to discover the balance that can lead to the optimal performance of the hybrid ABC.

Summing up the results, it can be concluded that:

- ✓ Hybridization of ABC in its three phases (employed bees phase, onlooker bees phase and scout bees phase) is produced the best result in all configurations independent of crossover operator type and mutation operator type.
- ✓ Hybridization of ABC in its two phases (employed bees phase and/or onlooker bees phase and/or scout bees phase) is produced the best result than hybridization of ABC in its one phase (employed bees phase or onlooker bees phase or scout bees phase).
- ✓ Hybridization of ABC in its two phases (employed bees phase and/or onlooker bees phase and/or scout bees phase) is given 100% results equal to the best-known solution in LA (40) and SWV (20) instances independent of the crossover operator type and the mutation operator type.
- ✓ The hybrid ABC in its two phases or in its one phase that use the OX and the PBX operators are produced better results than the hybrid ABC that use the CX and the MPX operators.
- ✓ Hybridization of ABC in the employed bees phase is produced the best result than hybridization of ABC in the scout bees phase.
- ✓ Hybridization of ABC in the scout bees phase is generated the best result than hybridization of ABC in the onlooker bees phase.
- ✓ Hybridization of ABC in its two phases employed bees phase and onlooker bees phase is given 100% results equal to the best-known solution in LA (40), SWV (20), TA (80) and DMU (80) instances using the MPX operator with inversion mutation or insertion mutation or swap mutation or displacement mutation.

The main limitation of the research is that we have used only four crossover operators: PMX, OX, CX and PBX.

6. CONCLUSION

Due to high level of complexity of job shop-scheduling problem, efficient methods for solving this type of problem are required.

The paper studies seven hybrid methods namely ABC, GA according to sixteen configurations to solve job shop scheduling problems with the objective of minimizing makespan and to test these algorithms on 250 Benchmark instances from the classical OR-library to evaluate the performance and to find the balance that can lead to the optimal performance of the hybrid ABC.

Based on the overall results, it can be decided that the proposed methods were capable to solve the job shop-scheduling problem successfully, efficiently, and robustly, which have been demonstrated by simulation tests on 250 Benchmark instances and that the paper presents a pilot study to find the answer about the balance between diversification and intensification in the artificial bee colony algorithm to solve the job shop scheduling problem.

These results are in good agreement with other studies which have shown that the hybridization improves the effectiveness of the algorithm and leading to better quality of solutions.

These results indicate how the multiple hybridizations, the crossover operator type, and the mutation operator type affect in a positive way the balance between diversification and intensification in the artificial bee colony algorithm to solve Job shop scheduling problem.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The proposed methods can be successfully used to solve multi objective job shop scheduling problem and flow shop scheduling problem and to solve mono objective flow shop scheduling problem.

In our future research, we intend to concentrate for solving a multi objective job shop and flow shop scheduling problem using hybrid methods.

REFRENCES:

[1] E.G. Talbi, "A taxonomy of hybrid metaheuristics", *Journal of Heuristics,* Vol. 8, No. 5, 2002, pp. 541-564.

[2] T. Bagchi, "*Multiobjective Scheduling By Genetic Algorithm* (1999 edition)", New York: Springer US, 1999.

[3] M.E. Aydin, and T.C. Fogarty, "A Distributed Evolutionary Simulated Annealing Algorithm for Combinatorial Optimization Problems", *Journal of Heuristics*, Vol. 10, No. 3, 2004, pp. 269-292.

[4] T. Satake, K. Morikawa, K. Takahashi, and N. Nakamura, "Simulated Annealing Approach for Minimizing the Makespan of the General Job- Shop", *International Journal of Production Economics*, Vol. 60, No. 1, 1999, pp. 515-522.

[5] C. Bierwith, "A Generalized Permutation Approach to Job Shop Scheduling with Genetic Algorithms", *OR Spektrum*, Vol. 17, No 2-3, 1995, pp. 87-92.

[6] J.F. Goncalves, J.M. Mendes, and M. Resende, "A hybrid genetic algorithm for the job shop scheduling problem", *European Journal of Operations Research*, Vol. 167, No. 1, 2004, pp. 77-95.

[7] E. Nowicki, and C. Smutnicki, "A Fast Taboo Search Algorithm for the Job Shop Problem", *Management Science*, Vol. 42, No. 6, 1996, pp. 797-813.

[8] F. Pezzella, and E. Merelli, "A Tabu Search Method Guided by Shifting Bottleneck for the Job Shop Scheduling Problem", *European Journal of Operational Research*, Vol. 120, No. 2, 2000, pp. 297-310.

[9] C. Blum, and M. Sampels, "An Ant Colony Optimization Algorithm for Shop Scheduling Problems", *Journal of Mathematical Modelling and Algorithms*, Vol. 3, No. 3, 2004, pp. 285-308.

[10] A. Colorni, M. Dorigo, V. Maniezzo, and M. Trubian, "Ant System for Job-Shop Scheduling", *Belgian Journal of Operations Research, Statistics and Computer Science (JORBEL)*, Vol. 34, No. 1, 1994, pp. 39-53.

[11] T. Satake, K. Morikawa, K. Takahashi, and N. Nakamura, "Neural Network Approach for Minimizing the Makespan of the General Job-Shop", *International Journal of Production Economics*, Vol. 33, No. 1-3, 1994, pp. 67-74.

[12] J. Adams, E. Balas, and D. Zawack, "The Shifting Bottleneck Procedure for Job Shop Scheduling", *Management Science*, Vol. 34, No. 3, 1988, pp. 391-401.

[13] W. Huang, and A. Yin, "An Improved Shifting Bottleneck Procedure for the Job Shop Scheduling Problem", *Computers & Operations Research*, Vol. 31, No. 12, 2004, pp. 2093-2110.

[14] E. Balas, and A. Vazacopoulos, "Guided Local Search with Shifting Bottleneck for Job Shop Scheduling", *Management Science,* Vol. 44, No. 2, 1998, pp. 262-275.

[15] R.M. Aiex, S. Binato, and M.G.C. Resende, "Parallel GRASP with Path-Relinking for Job Shop Scheduling", *Parallel Computing*, Vol. 29, No. 4, 2003, pp. 393-430.

[16] U. Dorndorf, E. Pesch, and T. Phan-Huy, "Constraint Propagation and Problem Decomposition: A Preprocessing Procedure for the Job Shop Problem", *Annals of Operations Research,* Vol. 115, No. 1-4, 2002, pp. 125-145.

[17] I. Kacem, S. Hammadi, and P. Borne, "Pareto-optimality approach for flexible job-shop scheduling problems: Hybridization of evolutionary algorithms and fuzzy logic", *Mathematics and Computers in Simulation*, Vol. 60, No. 3-5, 2002, pp. 245-276.

[18] B.M. Ombuki, and M. Ventresca, "Local search genetic algorithms for the job shop scheduling problem", *Applied Intelligence*, Vol. 21, No. 1, 2004, pp. 99-109.

[19] M. Watanabe, K. Ida, and M. Gen, "A genetic algorithm with modified crossover operator and search area adaptation for the jobshop scheduling problem", *Computers and Industrial Engineering*, Vol. 48, No. 4, 2005, pp. 743-752.

[20] F.T.S. Chan, S.H. Chung, and P.L.Y. Chan, "An adaptive genetic algorithm with dominated genes for distributed scheduling problems", *Expert Systems with Applications*, Vol. 29, No. 2, 2005, pp. 364-371.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[21] F. Pezzella, G. Morganti, and G. Ciaschetti, "A genetic algorithm for the flexible job-shop scheduling problem", *Computers and Operations Research*, Vol. 35, No. 10, 2008, pp. 3202-3212.

[22] F.T.S. Chan, K.L. Choy, and Bibhushan, "A genetic algorithmbased scheduler for multiproduct parallel machine sheet metal job shop", *Expert Systems with Applications*, Vol. 38, No. 7, 2011, pp. 8703-8715.

[23] R. Qing-dao-er-ji, and Y. Wang, "A new hybrid genetic algorithm for job shop scheduling problem", *Computers and Operations Research*, Vol. 39, No. 10, 2012, pp. 2291-2299.

[24] D. Lei, "Minimizing makespan for scheduling stochastic job shop with random breakdown", *Applied Mathematics and Computation*, Vol. 218, No. 24, 2012a, pp. 11851-11858.

[25] J. Huang, G.A. Süer, and S.B.R. Urs, "Genetic algorithm for rotary machine scheduling with dependent processing times", *Journal of Intelligent Manufacturing*, Vol. 23, No. 5, 2012, pp. 1931-1948.

[26] R. Zhang, "An artificial bee colony algorithm based on problem data properties for Scheduling job shops", *Procedia Engineering*, Vol. 23, No. 1, 2011, pp. 131-136.

[27] D. Lei, "Co-evolutionary genetic algorithm for fuzzy flexible job shop scheduling", *Applied Soft Computing*, Vol. 12, No. 8, 2012b, pp. 2237-2245.

[28] G. Jie, and G. Mitsuo, "A hybrid of genetic algorithm and bottleneck shifting for multiobjective flexible job shop scheduling problems", *Computers & Industrial Engineering*, Vol. 53, No. 1, 2007, pp. 149-162.

[29] H. Zhou, W. Cheung, and L.C. Leung, "Minimizing weighted tardiness of job-shop scheduling using a hybrid genetic algorithm", *European Journal of Operational Research*, Vol. 194, No. 3, 2009, pp. 637-649.

[30] H. Boukef, M. Benrejeb, and P. Borne, "Flexible job-shop scheduling problems resolution inspired from particle swarm optimization", *Studies in Informatics and Control*, Vol. 17, No. 3, 2008, pp. 241-252.

[31] L. Wang, G. Zhou, Y. Xu, S. Wang, and M. Liu, "An effective artificial bee colony algorithm for the flexible job-shop scheduling problem", *International Journal of Advanced Manufacturing Technology*, Vol. 60, No. 1-4, 2012, pp. 303-315.

[32] P. Pansuwan, N. Rukwong, and P. Pongcharoen, "Identifying optimum artificial bee colony (ABC) algorithm's parameters for scheduling the manufacture and assembly of complex products", *Second International Conference on Computer and Network Technology (ICCNT)*, 2010, pp. 339-343.

[33] S.I. Butt, and S. Hou-Fang, "Application of genetic algorithms and rules in scheduling of flexible job shops", *Journal of Applied Sciences*, Vol. 6, No. 7, 2006, pp. 1586-1590.

[34] N. Liouane, I. Saad, S. Hammadi, and P. Borne, "Ant systems & local search optimization for flexible job shop scheduling production", *International Journal of Computers Communications & Control*, Vol. 2, No. 2, 2007, pp. 174-184.

[35] N. Zribi, I. Kacem, A. El Kamel, and P. Borne, "Assignment and scheduling in flexible job-shops by hierarchical optimization", *IEEE Transactions on Systems Man and Cybernetics Part C-Applications and Reviews*, Vol. 37, No. 4, 2007, pp. 652-661.

[36] P. Fattahi, M. Saidi-Mehrabad, and F. Jolai, "Mathematical modeling and heuristic approaches to flexible job shop scheduling problems", *Journal of Intelligent Manufacturing*, Vol. 18, No. 3, 2007, pp. 331-342.

[37] J.Q. Li, Q.K. Pan, S.X. Xie, and Y.T. Wang, "An effective hybrid particle swarm optimization algorithm for flexible jobshop scheduling problem", *International Journal of Advanced Engineering Applications*, Vol. 1, No. 1, 2008, pp. 9-13.

[38] B.S. Girish, and N. Jawahar, "Scheduling job shop associated with multiple routings with genetic and ant colony heuristics", *International Journal of Production Research*, Vol. 47, No. 14, 2009, pp. 3891-3917.

[39] L.N. Xing, Y.W. Chen, P. Wang, Q.S. Zhao, and J. Xiong, "A knowledge-based ant colony optimization for flexible job shop scheduling problems", *Applied Soft Computing*, Vol. 10, No. 3, 2010, pp. 888-896.

[40] J. Li, Q. Pan, and S. Xie, "A hybrid variable neighborhood search algorithm for solving multi-objective flexible job shop problems", *Computer Science and Information Systems*, Vol. 7, No. 4, 2010, pp. 907-930.

[41] L.N. Xing, Y.W. Chen, and K.W. Yang, "Multi-population interactive coevolutionary algorithm for flexible job shop scheduling problems", *Computational Optimization and Applications*, Vol. 48, No. 1, 2011, pp. 139-155.

[42] N. Al-Hinai, and T.Y. Elmekkawy, "An efficient hybridized genetic algorithm architecture for the flexible job shop scheduling problem", *Flexible Services and Manufacturing Journal*, Vol. 23, No. 1, 2011, pp. 64-85.

[43] J. Jiang, M. Wen, K. Ma, X. Long, and J. Li, "Hybrid genetic algorithm for flexible job-shop scheduling with multi-objective", *Journal of Information and Computational Science*, Vol. 8, No. 11, 2011, pp. 2197-2205.

[44] C. Gutiérrez, and I. Garcia-Magarino, "Modular design of a hybrid genetic algorithm for a flexible job-shop scheduling problem", *Knowledge-Based Systems*, Vol. 24, No. 1, 2011, pp. 102-112.

[45] G. Moslehi, and M. Mahnam, "A Pareto approach to multi-objective flexible job-shop scheduling problem using particle swarm optimization and local search", *International Journal of Production Economics*, Vol. 129, No. 1, 2011, pp. 14-22.

[46] V. Roshanaei, A. Azab, and H. Elmaraghy, "Mathematical modelling and a meta-heuristic for flexible job shop scheduling", *International Journal of Production Research*, Vol. 51, No. 20, 2013, pp. 6247-6274.

[47] Y. Yuan, and H. Xu, "Flexible job shop scheduling using hybrid differential evolution algorithms", *Computers & Industrial Engineering*, Vol. 65, No. 2, 2013, pp. 246-260.

[48] M.E.T. Araghi, F. Jolai, and M. Rabiee, "Incorporating learning effect and deterioration for solving a SDST flexible job-shop scheduling problem with a hybrid meta-heuristic approach", *International Journal of Computer Integrated Manufacturing*, Vol. 27, No. 8, 2013, pp. 733-746.

[49] B. Barzegar, and H. Motameni, "Solving flexible job-shop scheduling problem using hybrid algorithm based on gravitational search algorithm and particle swarm optimization", *Journal of Advances in Computer Research Quarterly*, Vol. 4, No. 3, 2013, pp. 69-81.

[50] Y. Yuan, and H. Xu, "An integrated search heuristic for large-scale flexible job shop scheduling problems", *Computers & Operations Research*, Vol. 40, No. 12, 2013, pp. 2864-2877.

[51] Y. Yuan, H. Xu, and J. Yang, "A hybrid harmony search algorithm for the flexible job shop scheduling problem", *Applied Soft Computing*, Vol. 13, No. 7, 2013, pp. 3259-3272.

[52] H. Farughi, B.Y. Yegane, and M. Fathian, "A new critical path method and a memetic algorithm for flexible job shop scheduling with overlapping operations", *Simulation*, Vol. 89, No. 3, 2013, pp. 264-277.

[53] D. Karaboga, "An idea based on honey bee swarm for numerical optimization", *Technical Report TR06, Erciyes University, Engineering Faculty, Computer Engineering Department*, 2005.

[54] D. Karaboga, and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm", *Journal of Global Optimization*, Vol. 39, No. 3, 2007, pp. 459-471.

[55] J.A. Holland, (in press), "Adaptation in natural and artificial systems", MIT Press, (1975).

[56] F. Werner, "Heuristics: Theory and Applications (1st ed.)", New York: Nova Science Publishers, 2013.

[57] T.F. Abdelmaguid, "Representations in Genetic Algorithm for the Job Shop Scheduling Problem: A Computational Study", *Journal of Software Engineering & Applications*, Vol. 3, No. 12, 2010, pp. 1155-1162.

[58] R. Cheng, M. Gen, and Y. Tsujimura, "A Tutorial Survey of Job-Shop Scheduling Problems using Genetic Algorithms, part II: Hybrid genetic search strategies", *Computers & Industrial Engineering*, Vol. 36, No. 2, 1999, pp. 343-364.

[59] L. Asadzadeh, and K. Zamanifar, "An agent-based parallel approach for the job shop scheduling problem with genetic algorithms", *Mathematical and Computer modeling*, Vol. 52, No. 11-12, 2010, pp. 1957-1965.

[60] G. Cavory, R. Dupas, and G. Goncalves, "A genetic approach to solving the problem of cyclic job shop scheduling with linear constraints", *European Journal of Operational Research*, Vol. 161, No. 1, 2005, pp. 73-85.

[61] S.L. Moon, and H. Bae, "Genetic algorithms for job shop scheduling problems with alternative routings", *International Journal of*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Production Research, Vol. 46, No. 10, 2008, pp. 2695-2705.

[62] I.M. Oliver, D.J. Smith, and J.R.C. Holland, "Study of permutation crossover operators on the traveling salesman problem", *Proceedings of the second International Conference on Genetic Algorithms and their applications.* 1987, pp. 28-31.

[63] H. Yu, and W. Liang, "Neural network and genetic algorithm based approach to expand job shop scheduling problem", *Computers & Industrial Engineering*, Vol.39, No. 3-4, 2001, pp. 337-356.

[64] G. Syswerda, "The application of genetic algorithms to resource scheduling", *in Proceedings of the Fourth International Conference on Genetic Algorithms*, 1991, pp. 502- 508.

[65] G. Cavory, R. Dupas, and G. Goncalves, "A genetic approach to solving the problem of cyclic job shop scheduling with linear constraints", *European Journal of Operational Research*, Vol. 161, No. 1, 2005, pp. 73-85.

[66] C. Oguz, and M.F. Ercan, "A genetic algorithm for hybrid flow-shop scheduling with multiprocessor tasks", *Journal of Scheduling*, Vol. 8, No. 4, 2005, pp. 323-351.

[67] G. Zhang, L. Gao, and Y. Shi, "An effective genetic algorithm for the flexible job-shop scheduling problem", *Expert Systems with Applications*, Vol. 38, No. 4, 2011, pp. 3563-3573.

[68] E. Falkenauer, and S. Bouffouix, "A Genetic Algorithm for Job Shop", *in Proceedings of the IEEE International Conference on Robotics and Automation*, Sacramento, California, 1991, pp. 824-829.

[69] U. Dorndorf, and E. Pesch, "Evolution based learning in a job shop scheduling environment", *Computers and Operation Research*, Vol. 22, No. 1, 1995, pp. 25-40.

[70] A.C. Nearchou, "The effect of various operators on the genetic search for large scheduling problems", *International Journal of Production Economics*, Vol. 88, No. 2, 2004, pp. 191-203.

[71] H. Fisher, and G.L. Thompson, "Industrial Scheduling", *Englewood Cliffs, NJ: Prentice-Hall,* 1963.

[72] J. Adams, E. Balas, and D. Zawack, "The shifting bottleneck procedure for job-shop scheduling", *Management Science*, Vol. 34, No. 3, 1988, pp. 391-401.

[73] T. Yamada, and R. Nakano, "A genetic algorithm applicable to large-scale job-shop problems", *Proceedings of the second International Workshop on Parallel Problem Solving from Nature (PPSN'2).* Brussels, Belgium, 1992, pp. 281-290.

[74] S. Lawrence, "Resource constrained project scheduling: an experimental investigation of heuristic scheduling techniques", *Pittsburgh: Graduate School of Industrial Administration*, 1984.

[75] D. Applegate, and W. Cook, "A computational study of the job-shop scheduling problem", *ORSA Journal on Computing*, Vol. 3, No. 2, 1991, pp. 149-156.

[76] R.H. Storer, D. Wu, and R. Vaccari, "New search spaces for sequencing problems with application to job shop scheduling", *Management Science*, Vol. 38, No. 10, 1992, pp. 1495-1509.

[77] E.D. Taillard, "Parallel taboo search techniques for the job shop scheduling problem", *ORSA Journal on Computing*, Vol. 6, No. 2, 1994, pp. 108-117.

[78] E. Demirkol, S. Mehta, and R. Uzsoy, "A computational study of shifting bottleneck procedures for shop scheduling problems", *Journal of Heuristics*, Vol. 3, No. 2, 1997, pp. 111-137.

[79] J. Carlier, and E. Pison, "An algorithm for solving the job shop problem", *Management Science*, Vol. 35, No. 2, 1989, pp. 164-176.