Swarm intelligence algorithms’ solutions to the travelling salesman’s problem

J B. Odili1, A. Noraziah2,3, Roslina Mohd Sidek2
1Department of Mathematical Sciences, Anchor University Lagos, Nigeria
2Faculty of Computing, Universiti Malaysia Pahang, Kuantan 26300, Malaysia
3IBM Centre of Excellence, Universiti Malaysia Pahang, Kuantan 26300, Malaysia.

E-mail: jodili@aul.edu.ng, noraziah@ump.edu.my, roslinams@ump.edu.my

Abstract. This paper presents research findings on the application of swarm intelligence techniques in computational intelligence to solve the travelling salesman’s problem. The travelling salesman’s problem finds real-life application in post office mail delivery, school bus routing, delivery of food to home-bound people etc. After a number of experimental procedures, the study concludes that all the comparative algorithms are very efficient in providing solutions to the benchmark travelling salesman’s problems considered, though the Discrete Cuckoo Search and the African Buffalo Optimization have a slight edge in performance over the other comparative algorithms. In all, the study agrees with earlier studies in reaching the conclusion that swarm-based optimization techniques are not only effective but also are very efficient in providing solutions to the travelling salesman’s problems.

1. Introduction
Since the beginning of the 20th century, man has deliberately been seeking ways to enhance their quality of living leading to several successful research investigations. One of the prominent scientific developments has been the travelling salesman’s problem (TSP) developed in the 1930s [1]. Since its development, the TSP has attracted the attention of several researchers in Mathematics, Operations Research, Computational Intelligence and different aspects of engineering, thus leading to the concentration of several research procedures [2]. These research procedures have resulted in several outputs that have been of immense benefits to humanity today. Some notable real-life application domains of the TSP includes transportation (by air, land or sea) [3], job scheduling [4], vehicle routing [5], collision avoidance in robotics engineering [6], drilling of holes in motherboards [7], logistics planning [8], microchips manufacturing [9], DNA sequencing [10] and, most recently, in autonomous vehicles technology [11, 12].

In Computer Science, particularly, TSP has become, consciously or unconsciously, a veritable yardstick for validating search capacity of optimization algorithms [13]. To this effect, several optimization algorithms ranging from those developed late in the last century such as the Ant Colony Optimization, Particle Swarm Optimization, Genetic Algorithm, etc. the 21st century optimization algorithms such as the Cuckoo Search, Bat Algorithm, Firefly Algorithm, African Buffalo Optimization, Artificial Bee Colony etc. have all investigated the TSP.

This study is a comparative investigation of search capabilities of different metaheuristics in arriving at solutions to the TSP with the aim of assisting the research community in making informed choices about a particular swarm intelligence metaheuristic whenever they are confronted with a problem that requires the application of the TSP.

The rest of this paper is organized thus: section two examines the concept of the TSP, section three discusses the comparative algorithms; section four highlights the experimental setting; section five deals
with the experimental procedures cum discussion of results and section six draws conclusions on the study.

2. The travelling salesman’s problem
The TSP is a simulation of a particular anonymous salesman who has customers spread across different locations of a large city or in a number of cities/towns in a given location. The duty of the salesman is to visit each of his customers, and then return to his starting point [13]. Travelling Salesman’s Problems could either be symmetric or asymmetric. Symmetric TSP chronicles such problems where the travelling cost/distance between any two given locations in the graph is same in either to or fro directions. On the other hand, asymmetric TSP describes situations where the cost/distance between, at least two locations in the graph is not the same. The asymmetric situation finds practical application in one-way traffic or other civil engineering or commercial considerations [14].

In the simplest form, the TSP is described mathematically as: given a number of nodes/locations along with the traveling cost/weights between each pair, find the cheapest route that visits all of the nodes/locations and then returns to the starting node/location. In other words, mathematically, a TSP problem that has n nodes is described as a graph: \( G = (N, A) \), where \( N = \{1, 2, \ldots, n\} \) represents a number of nodes, \( A = \{(i, j) | i, j \in N \} \) represents a number of arcs. Each node represents a specific \( n \in N \) and each arc \( k, l \) serves as connector between nodes \( k \) and \( l \). The traveling cost from node \( k \) to node \( l \), utilizing a particular arc, is represented by \( c_{kl} \), meaning cost between node \( k \) to node \( l \).

The benchmark TSP instances used in this study are available in 769 TSPilib95 [15]. The resultant output obtained from this experiment were used to compare those of other bio-inspired techniques such as the Bat Algorithm (BA), Firefly Algorithm (FFA), Cuckoo Search (CS), African Buffalo Optimization (ABO), Hybrid Algorithm, Ant Colony Optimization and the Artificial Bee Colony (ABC). The choice of the comparative algorithms is borne out of the fact that, aside the ACO, they are all recently developed algorithm, otherwise called the 21st century algorithms in addition to their posting some of the best results in literature. Moreover, the above-listed algorithms are very popular among researchers and have proven to be among the most effective swarm intelligence techniques ever developed [16].

3. Comparative algorithms
In view of the relevance of the TSP in terms of its practical applications in mathematics, engineering, science and technology, several algorithms have been applied to solve this problem. The algorithms of interest to us in this study are the 21st century algorithms since their results, generally, tend to be better than those of the 20th century algorithms such as the GA, ACO and PSO. The superiority of the 21st century algorithms over its peers developed in the 20th century may be due to hardware improvements, the use of more sophisticated programming languages as well as improved programming skills. The comparative algorithms in this study are the Ant Colony Optimization [17], Artificial Bee Colony [18], Hybrid Algorithm [19], African Buffalo Optimization [20], Discrete Cuckoo Search [21], Firefly Algorithm [22] and the Bat Algorithm [23].

A closer look at our choice of algorithm indicates that, aside being population-based and recently-designed (except, of course the ACO, chosen for its exceptional search effectiveness), it incorporates a hybrid (HA) which is hybridization of the Ant Colony Optimization with the Artificial Bee Colony [19] as well as other optimization techniques that deploy different communication methods in organizing the herd/flock for their search enterprise. For instance, while the ACO employs ring topology that transmits from one ant to the other, the PSO employs the Von Neumann communication topology where a particle relates information to the immediate neighbours and the ABO and FFA employ the star topology where the search agents disseminate search information to the entire population [24]. Overall, these algorithms were chosen for this comparative investigation because of their effectiveness and popularity among researchers. A brief review of each of the comparative algorithms is hereby presented.

3.1. African buffalo optimization
African Buffalo Optimization (ABO) basically models the communal decision-making procedures communicative prowess, harnessing the collective intelligence of the entire herd and the extensive
memory capacities of the African buffalos in their migration from one location to another in search of food in the vast African savannah and forests. Using two major vocalizations: attract /maaa/ and repulse /waaa/ calls, the buffalos are able to organize themselves in their search for solutions [25]. The /waaa/ calls is deployed to warn the buffalo herd of impending dangers arising from the presence of predators; express the lack of pastures in a particular location; scare an approaching inferior; assert dominance and, therefore, urge the herd to move on to better, safer or more rewarding locations (exploration). On the other hand, the /maaa/ call is used to reassure an inferior; express satisfaction with the security situation of a location and encourage the buffalos to further graze a field (exploitation) [26]. The ABO algorithm is presented in Figure 1:

1. Begin
2. Randomly initialize the buffalos to nodes at the search space;
3. While (not termination), do
4. For k=1: n (n= population), do
5. Evaluate the buffalos’ exploitation fitness using: 

\[ m_k' = m_k + l_p1(bg - w_k) + l_p2(bp_k - w_k) \]

\[ w_k' = \frac{(w_k + m_k)}{\lambda} \]

6. Is bgmax updating? Yes, go to 11. If No in 10 iterations, go to 2
7. End for
8. End while
9. Output best solution.
10. End

Figure 1: ABO algorithm

So far, the ABO has been successfully applied to solve a few optimization problems. Some of ABO application areas includes parameter-tuning of PID Controllers for Automatic Voltage Regulators [27], energy and delay routing in Mobile Ad-hoc Network [28], numerical function optimization [29] etc. Because the application areas of this optimization technique are not yet widespread, it is difficult to pinpoint its weaknesses. Its strengths, however, includes fast speed due to the use of few parameters. Again, being a parameter-less optimization technique, it is user-friendly since the user does not have to bother with tedious parameter-tuning processes.

B. Cuckoo Search

The Cuckoo Search (CS) which is a simulation of the subtle attitude of the cuckoo bird was designed by X. Yang and S. Deb[30]. The cuckoo lays her eggs in the nests of other birds, sometimes of other species with the hope that those other birds will execute the maternal function of incubating those eggs. Whenever the host birds discover the cuckoo prank, it either abandons the nest or throw such strange eggs away. Otherwise, it goes ahead to incubate the eggs. The cuckoo bird, on its own perfects the act of subtlety by imitating the egg of its host to perpetuate its kind by such fraud. In CS, the host eggs in a nest represents an optimization solution while the cuckoo egg is a representation of a newer solution with the objective of using the newer solution to replace the existing one.
In CS, it is assumed that a cuckoo bird lays an egg at a time in any nest chosen randomly. The nest with the best quality/number of eggs carries on to the next generation. Moreover, there are a fixed number of nests and the cuckoo egg is discovered by the host bird with a given probability, usually between 0 and 1. The CS pseudocode [31] is presented in Figure 2. The CS is quite an effective algorithm obtaining results where other algorithms struggle [21]. In spite of seeming wide applications, the Cuckoo search has the problem of speed due to its use of several parameters and sometimes falling into local optima hence the emergence of several varieties of the algorithms such as discrete CS, Improved CS etc.

4. Experimental overview
In this study, the ABO was implemented using MATLAB on a desktop computer: Intel Duo Core™ i7-3770 CPU, 3.40 GHz with 4GB RAM on a set of three symmetric TSP instances available in TSPLIB95. Also, please note that the statistical evaluations in this study were done using the IBM SPSS, version 22. The experimental parameters are presented in Table 1. These chosen parameter values are popular among researchers for their capacity to produce very good outcomes in solving TSP problems.

4.1 Experimental results and discussions
The first experiment involves eight benchmark TSP instances ranging from 52 to 225 cities. The experimental datasets are Berlin52, St70, Eil76, Pr76, KroA100, Eil101, Ch150, Tsp225. These TSP instances chosen in this study are not only complex but popular among researchers. The results obtained from the ABO was used to compare those from ABC, HA and ACO. The simulation result is presented in Table 2. Please note that in Table 2, the “Mean” refers to the mean fitness of each algorithm after 20 runs and the “PDB %” (Percentage Deviation from Best %) values are obtained using Equation 1

\[
\text{PDB} \% = \left( \frac{\text{Best Value} - \text{Optima}}{\text{Best value}} \right) \times 100
\]

where ‘Optima’ simply refers to the benchmark optimum and ‘Best Value’ is the overall best outcome of each algorithm after 20 runs of each algorithm.

Table 1 reveals the comparatively good performances of the four algorithms under consideration with the ABO slightly outperforming the other algorithms in all test cases. For instance, the ABO managed to obtain the optimal outcome to two TSP instances: Berlin52 and Eil76. The other algorithms in spite of their good results could not obtain the optimal outcome to any of the problems. It must be emphasized
that in as much as it is beneficial to obtain the optimum solution, metaheuristics, unlike their deterministic algorithms do not guarantee the optimum results.

Another clear observation is that the ABO obtained the nearest-optimal solution to the other six instances than its peers. Moreover, using the mean score as a yardstick of measurement, the ABO outcomes seem to be competitive. It posted the best average score in all instances. Of the three other algorithms, it is rather difficult to explain the below-par performance of the Hybrid Algorithm (HA) that uses a similar memory matrix like the ABO. The performance of the HA could be traceable to its use of several parameters. This is because the HA combines the ABC with ACO in its attempt at solutions. ABC and ACO algorithms deploy large number of parameters to obtain good solutions.

Table 1: ABO, ACO, ABC compared

| Problem | No of Cities | Opt | Method | Best   | Mean   | PDB (%) |
|---------|--------------|-----|--------|--------|--------|---------|
| Berlin52| 52           | 7542| ABO    | 7542   | 7616   | 0.98    |
|         |              |     | ACO    | 7548.99| 7659.31| 1.52    |
|         |              |     | ABC    | 9479.11| 10,390.26| 37.72  |
|         |              |     | HA     | 7544.37| 7544.37| 0.03    |
| St70    | 70           | 675 | ABO    | 676    | 678.33 | 0.01    |
|         |              |     | ACO    | 696.05 | 709.16 | 4.73    |
|         |              |     | ABC    | 1162.12| 1230.49| 81.73   |
|         |              |     | HA     | 687.24 | 700.58 | 3.47    |
| Eil76   | 76           | 538 | ABO    | 538    | 563.04 | 0       |
|         |              |     | ACO    | 554.46 | 561.98 | 3.04    |
|         |              |     | ABC    | 877.28 | 931.44 | 70.78   |
|         |              |     | HA     | 551.07 | 557.98 | 2.31    |
| Pr76    | 76           | 108159| ABO | 108167 | 108,396| 0.01    |
|         |              |     | ACO    | 115,166.66| 116,321.2| 7.55   |
|         |              |     | ABC    | 195,198.9| 205,119.6| 89.65  |
|         |              |     | HA     | 113,798.5| 115,072.| 6.39    |
| Kroa100 | 100          | 21282| ABO   | 21311  | 22163.8| 0.001   |
|         |              |     | ACO    | 22,455.89| 22,880.12| 7.49   |
|         |              |     | ABC    | 49,519.51| 53,840.03| 152.94 |
|         |              |     | HA     | 22,122.75| 22,435.3| 5.40    |
| Eil101  | 101          | 629 | ABO    | 640    | 640    | 0.17    |
|         |              |     | ACO    | 678.04 | 693.42 | 7.96    |
|         |              |     | ABC    | 1237.31| 1315.95| 104.88  |
|         |              |     | HA     | 672.71 | 683.39 | 6.39    |
| Ch150   | 150          | 6528| ABO    | 6532   | 6601   | 0.001   |
|         |              |     | ACO    | 6648.51| 6702.87| 2.61    |
|         |              |     | ABC    | 20,908.89| 21,617.48| 230.93 |
|         |              |     | HA     | 6641.69| 6677.12| 2.21    |
| Tsp225  | 225          | 3916| ABO    | 3917   | 3982   | 0.0002  |
|         |              |     | ACO    | 4112.35| 4176.08| 8.22    |
|         |              |     | ABC    | 16,998.41| 17,955.12| 365.27 |
|         |              |     | HA     | 4090.54| 4157.85| 7.74    |

However, statistically, using the Wilcoxon Signed Rank Test, there is a significant difference between the best performances of the ABO with those of the optimal solution as can be seen from Table 3. Since ABO had the best result then there is no need to compare the outcome of the other comparative algorithms with the Optimal solution from the TSPLIB95. Rather our statistical comparison should be between the best results of the comparative algorithms. The comparative results are presented in Tables 2-5.

H0: There is no significant difference between the results of ABO with those of ABC, ACO and HA
Hα: There is a significant difference between the results of ABO with those of ABC, ACO and HA
In comparing the best results of the ABO with those of the ACO, it was discovered that there was a significant difference in outcome in favour of the ABO (See Table 3). However, statistically there is no significant difference between the ABO best outcomes with those of HA best outcomes, hence we retain the null hypothesis (See Table 3).

Table 3: Wilcoxon Signed Rank Test

Asymptotic significances are displayed. The significance level is .05.

Similarly, there is no significant difference between the best results of the ABO with those of the ABC as can be seen in Table 5. In any case a closer look at statistical Table 6 reveals that the Optima-ABO best has the best significant level of 0.094 compared to those of ABC (0.123), HA (0.256) and ACO (0.251). However, we reject the null hypothesis (H₀) and accept the alternative hypothesis (H₁).

Table 5: Wilcoxon Signed Rank Test

Asymptotic significances are displayed. The significance level is .05.
4.2. ABO and Cuckoo search on TSP

The second set of experiments on the symmetric TSP was performed involving some of the most recently-designed algorithms: the ABO and the Cuckoo Search (CS) algorithm. The experiment involved 35 symmetric TSP ranging from 51 to 1379 cities. Table 6 presents the performance of two effective algorithms is solving TSP. The CS obtained 24 optimal solutions out of the 35.

Table 6: ABO and Cuckoo Search

| TSP Cases | CS Opt | CS Best | CS Mean | SD | ABO Best Mean | SD |
|-----------|--------|---------|---------|----|---------------|----|
| eil51     | 426    | 426     | 426     | 0.00 | 426           | 427 | 30.05 |
| berlin52  | 7542   | 7542    | 7542    | 0.00 | 7542          | 7659 | 36.04 |
| st70      | 675    | 675     | 675     | 0.00 | 676           | 676  | 0.00  |
| pr76      | 108159 | 108159  | 108159  | 0.00 | 108167        | 108167 | 0.00 |
| eil76     | 538    | 538     | 538.03  | 0.17 | 538           | 538  | 0.00  |
| kroA100   | 21282  | 21282   | 21282   | 0.00 | 21554         | 21554 | 0.00 |
| kroB100   | 22141  | 22141   | 22141.53| 2.87 | 22160         | 22509 | 25.58 |
| kroC100   | 20749  | 20749   | 20749   | 0.00 | 20755         | 20755 | 0.00 |
| kroD100   | 21294  | 21294   | 21304.33| 21.79| 21347         | 21462 | 158.37|
| kroE100   | 22068  | 22068   | 2281.26 | 18.50| 22088         | 22088 | 0.00  |
| eil101    | 629    | 629     | 630.43  | 1.14 | 640           | 640  | 0.00  |
| lin105    | 14379  | 14379   | 14379   | 0.00 | 14419         | 14453 | 254.12|
| pr107     | 44303  | 44303   | 44307.06| 12.90| 44407         | 44407 | 0.00  |
| pr124     | 59030  | 59030   | 59030   | 0.00 | 59058         | 59058 | 0.00  |
| bier127   | 118282 | 118282  | 118359.63| 12.73| 118297        | 118863 | 577.79|
| ch130     | 6110   | 6110    | 6135.96 | 21.24| 6111          | 6111  | 0.00  |
| pr136     | 96772  | 96790   | 97009.26| 134.43| 96784         | 96784 | 0.00  |
| pr144     | 58537  | 58537   | 58537   | 0.00 | 58587         | 58587 | 0.00  |
| ch150     | 6528   | 6528    | 6549.9  | 20.51| 6533          | 6601  | 62.41 |
| kroB150   | 26130  | 26130   | 26159.3 | 34.72| 26169         | 26431 | 383.94|
| pr152     | 73682  | 73682   | 73682   | 0.00 | 73730         | 73730 | 0.00  |
| kroA200   | 29368  | 29382   | 29446.66| 95.68| 29370         | 29370 | 0.00  |
| kroB200   | 29437  | 29448   | 29542.49| 92.17| 29487         | 29534 | 67.18 |
| tsp225    | 3916   | 3916    | 3958.76 | 20.73| 3917          | 3917  | 0.00  |
| gil262    | 2378   | 2382    | 2394.5  | 9.56 | 2378          | 2378  | 0.00  |
| a280      | 2579   | 2579    | 2592.33 | 11.86| 2579          | 2579  | 0.00  |
| pr299     | 48191  | 48207   | 48470.53| 131.79| 48211         | 48211 | 0.00  |
| lin318    | 42029  | 42125   | 42434.73| 185.43| 42101         | 42336 | 1543.43|
| rd400     | 15281  | 15447   | 15533.73| 60.56| 15300         | 15300 | 0.00  |
| fl417     | 11861  | 11873   | 11910.53| 20.45| 11862         | 11862 | 0.00  |
| pr439     | 107217 | 107447  | 107960.5| 438.15| 107340        | 107340 | 0.00 |
| rat575    | 6773   | 6896    | 6956.73 | 35.74| 6774          | 6810  | 7.67  |
| rat783    | 8806   | 9043    | 9109.26 | 38.09| 8811          | 8881  | 0.00  |
| pr1002    | 259045 | 266508  | 268630.03| 1126.8| 259132        | 259132 | 0.00 |
| nrw1379   | 56638  | 58951   | 59349.53| 213.89| 56653         | 56653 | 0.00  |
On the other hand, the ABO obtained six optimal results. Similarly, The CS obtained 10 optimal solutions in all 20 runs of the algorithm as could be seen in the Mean calculation of results. The runs that produced the accurate results in all runs of the CS are the eil51, berlin52, st70, pr76, kroA100, kroC100, lin105, pr124, pr144 and pr152. On this count, the ABO obtained three accurate solutions in all runs and these are in eil76, gil262 and a280. A closer examination, however, reveals that that the DCS obtained 'perfect' runs in TSP instances with cities ranging from 51-152.

In TSP instances with larger datasets running into hundreds, this effectiveness was halted. On the other hand, the ABO’s ‘good’ runs are more widespread. Versatility, being one of the indices of a good algorithm [36], the ABO appears to be a more versatile algorithm in this case. Nonetheless, the CS proves to be a more effective algorithm than the ABO in terms of obtaining the optimal solutions to the TSP instances under investigation here. Furthermore, in terms of the standard deviation from the mean score, both algorithms performed excellently with DCS having 0.00 standard deviation in 10 instances (the eil51, berlin52, st70, pr76, kroA100, kroC100, lin105, pr124, pr144 and pr152). Here the ABO produced more consistent result with 0.00 standard deviation in 24 instances. This is a proof of good algorithm convergence: attributes of a good optimization method [36]. Statistically, as can be seen from Table 6, using the Wilcoxon Rank Sum Test, there exist significant differences in the results of both the ABO and the Optimal solutions as well as in those of CS and the Optimal results available in TSPLIB95 (See Tables 6a and Table 6b). However, statistically, there exist not significant difference in the performance of the CS and the ABO in solving the TSP under investigation (See Table 6c), resulting in the acceptance of the null hypothesis (H₀): H₀ There is no significant difference in the performance of the CS and the ABO in solving the TSP instances under investigation.

Hₐ: There is a significant difference in the performance of the CS and the ABO in solving the TSP instances under investigation.

### Wilcoxon Signed Ranks Test

#### Table 6(a): Test Statistics

| Null Hypothesis                                                                 | Test          | Sig.  | Decision              |
|---------------------------------------------------------------------------------|---------------|-------|-----------------------|
| The median of differences between ABO and OPT equals 0.                         | Related-Samples Wilcoxon Signed Rank Test | .000  | Reject the null hypothesis |

Asymptotic significances are displayed. The significance level is .05.

#### Table 6(b): Test Statistics

| Null Hypothesis                                                                 | Test          | Sig.  | Decision              |
|---------------------------------------------------------------------------------|---------------|-------|-----------------------|
| The median of differences between CS and ABO equals 0.                          | Related-Samples Wilcoxon Signed Rank Test | .639  | Retain the null hypothesis |

Asymptotic significances are displayed. The significance level is .05.

#### Table 6(c): Test Statistics

| Null Hypothesis                                                                 | Test          | Sig.  | Decision              |
|---------------------------------------------------------------------------------|---------------|-------|-----------------------|
| The median of differences between ABO and OPT equals 0.                         | Related-Samples Wilcoxon Signed Rank Test | .000  | Reject the null hypothesis |

Asymptotic significances are displayed. The significance level is .05.
With the .05 used as the degree of freedom, there exists no significant difference in CS and ABO to solving the TSP instances since the outcome of the ABO in relation to the optimal solutions (OPT) resulted in .638 significances. Since both outcomes are beyond .05, we retain the null hypothesis (H₀).

4.3. ABO with Bat Algorithm and Firefly Algorithm for TSP

The performance of three recently designed and excellent algorithms are examined in attempting solutions to the TSP, which are Bat Algorithm, Firefly Algorithm [22] and ABO as shown in Table 8.

Table 8: ABO with BA and FFA

| TSP Cases | Optima | BA | ABO | FFA |
|-----------|--------|----|-----|-----|
| Oliver30  | 420    | 420| 425 | 420 |
| Eil51     | 426    | 430| 426 | 426 |
| Berlin52  | 7542   | 7542| 7542| 7542|
| St70      | 675    | 675| 676 | 675 |
| Eil76     | 538    | 538| 538 | 543 |
| Krob100   | 21,282 | 21,292| 21554| 21,282|
| Krob100   | 22,140 | 22,373| 22160| 22,183|
| Kro100    | 20,749 | 20,802| 20755| 20,766|
| Kro100    | 21,294 | 21,727| 21347| 21,408|
| Kroe100   | 22,068 | 22,323| 22088| 22,079|
| Eil101    | 629    | 640 | 640 | 643 |
| Pr107     | 44,303 | 44,618| 44407| 44,303|
| Pr124     | 59,030 | 59,030| 59058| 59,030|
| Pr136     | 96,772 | 100,485| 96784| 97,716|
| Pr144     | 58,537 | 58,588| 58587| 58,546|
| Pr152     | 73,682 | 74,172| 73650| 74,033|

As is evident in Table 8, all the comparative algorithms performed excellently well with ABO obtaining four optimal solutions to BA’s five and FFA’s six. Comparatively, therefore, the FFA is more effective, followed by BA and ABO. It is rather interesting to note that BA and ABO obtained optimal results in datasets of less than 100 cities. It is only FA obtained optimal solution in two other TSP instances that involved more than 100 cities, specifically in pr107 and pr124. Similarly, it worthy of note that the performance of the ABO in the remaining instances is commendable, even when it fails to obtain the optimal result, its results are quite near the optimal.

5. Conclusion

This paper examined the performance of six swarm-intelligence techniques in providing optimize solutions to the symmetric travelling salesman’s problems. It is safe to assert optimize technique that need maximize output cum profits while minimizing the input [31]. It is our belief that the comparative performance of these 21st optimization would assist researchers in choice of swarm intelligence techniques whenever they are confronted with real-life. After a number of experimental procedures, this study concludes that though all the algorithms are very effective in solving the benchmark symmetric travelling salesman’s problems under investigation. Even though, the statistical evaluations indicate that there is no significant difference in the performance of the comparative algorithms, however, a closer observation indicates the DCS and the ABO have slight edge over the other algorithms in providing solutions to the benchmark-travelling salesman’s problem instances investigated.

Acknowledgment

Appreciation conveyed to Anchor University, Lagos; Ministry of Higher Education Malaysia for FRGS RDU190185; and University Malaysia Pahang Grant RDU170303 for financing this research.
References

[1] Odili, J.B.: ‘Application of Ant Colony Optimization to Solving the Traveling Salesman\'s Problem’, Science Journal of Electrical & Electronic Engineering, 2013, 2013.

[2] Dorogo, M., and Gambardella, L.: ‘Ant-Q: A reinforcement learning approach to the traveling salesman problem’, in Editor (Ed.s): ‘Book Ant-Q: A reinforcement learning approach to the traveling salesman problem’ (2016, edn.), pp. 252-260.

[3] Rodriguez, A., and Ruiz, R.: ‘The Effect Of The Asymmetry Of Road Transportation Networks On The Traveling Salesman Problem’, Computers & Operations Research, 2012, 39, (7), pp. 1566-1576.

[4] Conway, R.W., Maxwell, W.L., and Miller, L.W.: ‘Theory of Scheduling’ (Courier Corporation, 2012, 2012).

[5] Hussein, I.L., and Stupanovic, D.M.: ‘Effective Coverage Control for Mobile Sensor Networks with Guaranteed Collision Avoidance’, IEEE Transactions on Control Systems Technology, 2007, 15, (1), pp. 642-657.

[6] Yang, H.C., Liu, K.J., and Hung, M. H.: ‘Drill-Path Optimization with Time Limit and Thermal Protection’, Advanced Materials Research, 2012, pp. 153-159.

[7] Rodriguez, A. and Ruiz, R.: ‘The Effect of The Asymmetry of Road Transportation Networks on The Traveling Salesman Problem’, Computers & Operations Research, 2012, 39, (1), pp. 1566-1576.

[8] Bischl, D., and Soeren, C.G.: ‘The Vehicle Routing Problem in Field Logistics: Part II’, Biosystems Engineering, 2010, 105, pp. 180-188.

[9] Hao, X.: ‘Optimization Models and Heuristic Method Based on Simulated Annealing Strategy for Traveling Salesman Problem’, Applied Mechanics and Materials, 2010, pp. 1180-1184.

[10] Hoffman, K.L., Padberg, M., and Rinaldi, G.: ‘Traveling Salesman Problem’, Encyclopedia of Operations Research and Management Science, ed: Springer, 2013, pp. 1573-1578.

[11] Savla, K., Frazzoli, E., and Bullo, F.: ‘Traveling Salesperson Problems for The Dubins Vehicle’, IEEE Transactions on Automatic Control, 2008, 53, (1), pp. 1378-1391.

[12] Waldrop, M.M.: ‘No Drivers Required’, Ed: Nature Publishing Group Macmillan Building, 4 Crinan St, London N1 9xw, England, 2015.

[13] Applegate, D.L., Bixby, R. E., Chvatal, V., and Cook, W. J.: ‘The Traveling Salesman Problem: A Computational Study’, Princeton University Press, 2011.

[14] Kefi, S, Rokhani, N, Kromer, P, and Alimi, A.M.: ‘A New Ant Supervised-PSO Variant Applied to Traveling Salesman Problem’, Hybrid Intelligent Systems, ed: Springer, 2016, pp. 87-101.

[15] Reinnert, G: Tsplib95, 1995, URL: http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95, 2012.

[16] Chetty, S, and Adewumi, A.O.: ‘Comparison Study Of Swarm Intelligence Techniques For The Annual Crop Planning Problem’, IEEE Transactions on Evolutionary Computation, 2014, 18, pp. 258-268.

[17] Dorigo, M, and Stützle, T: ‘Ant Colony Optimization: Overview And Recent Advances’, Handbook of Metaheuristics, ed: Springer, 2010, pp. 227-263.

[18] Karaboga, D, Gorkemli, B, Ozturk, C, and Karaboga, N.: ‘A Comprehensive Survey: Artificial Bee Colony Algorithm And Applications’, Artificial Intelligence Review, 2014, 42, pp. 21-57.

[19] UZ M.G., Kiran, and E. Ozcelyan, E.: ‘A Hierarchic Approach Based On Swarm Intelligence To Solve The Traveling Salesman Problem’, Turkish Journal of Electrical Engineering & Computer Sciences, 2015, 23, pp. 103-117.

[20] Odili, J.B., Kahar, M.N., and Noraziah, A.: ‘Solving Traveling Salesman’s Problem using African Buffalo Optimization, Honey Bee Mating Optimization & Lin-Kerninghan Algorithms’, World Applied Sciences Journal, 2016, 34, (7), pp. 911-916.

[21] Ouaarab, A., Ahoud, B., and Yang, X.-S.: ‘Discrete Cuckoo Search Algorithm for The Travelling Salesman Problem’, Neural Computing and Applications, 2014, 24, (7-8), pp. 1659-1669.

[22] Yang, X.-S., Hosseini, S.S.S., and Gandomi, A.H.: ‘Firefly Algorithm for Solving Non-Convex Economic Dispatch Problems with Valve Loading Effect’, Applied Soft Computing, 2012, 12, (3), pp. 1180-1186.

[23] Yang, X.-S., and Hossein Gandomi, A.: ‘Bat Algorithm: A Novel Approach For Global Engineering Optimization’, Engineering Computations, 2012, 29, (5), pp. 464-483.

[24] Sarathambekai, S., and Umamaheswari, K.: ‘Task Scheduling in Distributed Systems using Heap Intelligent Discrete Particle Swarm Optimization’, Computational Intelligence, 2017.

[25] Odili, J.B., and Mohmad Kahar, M.N.: ‘Solving The Traveling Salesman's Problem using The African Buffalo Optimization’, Computational intelligence and neuroscience, 2016, 2016, pp. 1-12.

[26] Odili, J.B., and Kahar, M.N.M.: ‘African Buffalo Optimization (ABO): A New Meta-Heuristic Algorithm’, Journal of Advanced & Applied Sciences 2015, 03, (03), pp. 101-106.

[27] Odili, J.B., Kahar, M.N., and Noraziah, A.: ‘Parameters-tuning of PID Controller For Automatic Voltage Regulators using The African Buffalo Optimization’, PLoS One, 2017, 12, (4), pp. 1-17.

[28] Hassan, M.H., and Muniyandi, R.C.: ‘An Improved Hybrid Technique for Energy and Delay Routing in Mobile Ad-Hoc Networks’, International Journal of Applied Engineering Research, 2017, 12, (1), pp. 134-139.

[29] Odili, J.B., and Kahar, M.N.M.: ‘Numerical Function Optimization Solutions using The African Buffalo Optimization Algorithm (ABO)’, British Journal of Mathematics & Computer Science, 2015, 10, (1), pp. 1-12.

[30] Yang, X.-S., and Deb, S.: ‘Cuckoo Search Via Lévy Flights’, in Editor (Ed.)’(Eds.): ‘Book Cuckoo search via Lévy flights’ (IEEE, 2009, edn.), pp. 210-214.

[31] Odili, J. B., & Kahar, M. N. M. ‘African Buffalo Optimization’, International Journal of Software Engineering and Computer Systems, 2016, 2, (1), pp. 28-50.