

## A MARRIAGE IN HONEY BEE OPTIMISATION APPROACH TO THE ASYMMETRIC TRAVELLING SALESMAN PROBLEM

YUKSEL CELIK<sup>1</sup> AND ERKAN ULKER<sup>2</sup>

<sup>1</sup>Computer Programming Department  
Vocational School  
Karamanoglu Mehmetbey University  
Karaman 70100, Turkey  
celik-yuksel@hotmail.com

<sup>2</sup>Computer Engineering Department  
Selcuk University  
Konya, Turkey  
eulker@selcuk.edu.tr

Received December 2010; revised April 2011

**ABSTRACT.** *In the travelling salesman problem (TSP), a travelling salesman completes a tour of “n” number of cities by stopping once in each city and completes the tour by returning to his starting point, while minimizing the distance and the cost. The asymmetric travelling salesman problem (ATSP) is the problem in which the cost of travel from city A to B is different from that from B to A. Marriage in Honey Bee Optimisation (MBO) is a meta-heuristic procedure inspired by the mating and insemination process of honey bees. In this study, we seek to use an MBO algorithm for an optimal solution to the ATSP problem, which has previously been solved by different methods. The results of the MBO algorithm for ATSP are compared with Genetic Algorithm (GA), another meta-heuristic method.*

**Keywords:** Marriage in honey bee optimisation (MBO), Travelling salesman problem (TSP), Asymmetric travelling salesman problem (ATSP), Swarm intelligence

1. **Introduction.** The Travelling Salesman Problem (TSP), for N given nodes (cities), is the problem of finding the shortest (with the least cost) route that comes back to the starting point after stopping by each node once. It is easily defined but is an NP-Hard problem [1]. The solution to this problem can also be viewed as a Hamilton circle [2].

TSP can be defined mathematically in two ways: the graph problem and the permutation problem [3]. The TSP can be represented as a graph problem as follows: given a graph  $G = (V, E)$ , let  $F$  denote all Hamilton circles on  $G$ . For every edge  $E$ , there is a given weight ratio. The TSP is the problem of finding one tour (Hamilton circle) that visits all nodes with the least cost [3].

Three basic variants of the TSP problem have been proposed: Symmetric, Asymmetric and Euclidean. If the distances from city A to B and from city B to A are always equal, then this is called the symmetric TSP. The opposite case, in which the return distances are not always the same, is called the asymmetric TSP. The Euclidean TSP is the case in which the  $n$  nodes of the graph are points in  $R^d$ , and the distances between them are the Euclidean distances [4]. Over the last decade, there has been widespread usage of meta-heuristic algorithms' searching and optimisation tools in various problem areas. Meta-heuristic algorithms are easy to use and provide a general perspective that can be exploited in their own comprehensive applications as well in search and other optimisation tools.

Among these, Genetic Algorithm (GA) is a search method that follows the evolution mechanisms in nature to find specific data out of a data set. GA are iterative, probabilistic search methods that were developed based on an analogy to evolutionary genetics and Darwin's theory of natural selection. The main aim of genetic algorithms is to create artificial search algorithms that require minimal problem knowledge [5].

TSP is not only used for genetic algorithms but also to compare, evaluate and measure all new heuristic algorithms (e.g., ant system, evolutionary methods, neural networks, tabu search, simulated annealing and greedy seeking) [6] that seek to provide good results in a reasonable period. Johnson et al. [7] also examined some local search methods that produce approximate results.

Choi et al. have carried out an area search combined with a genetic algorithm [8]. Ausiello et al. [9] have suggested an on-line ATSP, and Brest et al. have contributed a heuristic suggestion for ATSP [10].

Simulated Annealing (SA) though the conventional gradient method is a fast algorithm, it requires a good initial estimate in order to avoid local minima. SA is an optimization technique based on calculation of the state function in statistical mechanics and has been used for many applications [11]. Gao et al. [12] has proposed that the probability of local minimum can be reduced by using SA and Clonal Selection Algorithm. Takano et al. [13], improved the accuracy of the solutions Combining SA and Self Organizing Map.

In addition to evolutionary algorithms, the area of swarm intelligence has developed over the last decade, seeking to solve search problems by modelling the behaviours of social insects such as bees and ants. The use of the Ant colony algorithm (ACO) as the most appropriate evolutionary algorithm was first suggested by Dorigo [13] and Dorigo et al. [14]. Recently, honey bees are among the social insects that are the most studied. Marriage in Honey Bee Optimisation (MBO), one of the search algorithms inspired by the real courtship process of honey bees, is a new swarm intelligence optimisation method that is used for simulating social systems. In 2001, Abbass [15,16] firstly suggested that MBO, a meta-heuristic method inspired by the social phylogenetics of honey bees, be used to solve integrated optimisation problems of probability and applied this method to a number of suggestive satisfaction problems. His studies indicate that MBO made progress in solving satisfaction problems that are well known to be empirical. In 2006, H. S. Chang [17] provided some results on the MBO convergence of the probabilistic dynamic programming application. In 2007, Abbass et al. defined the Honey Bee Marriage Optimisation (HMBO) algorithm, which is another honey bee optimisation, and applied it to the problem of optimal water tank operation.

In this study, we have sought to use the MBO algorithm to obtain an optimal solution for ATSP, which has been solved by different optimisation methods in the past. This article is organised as follows. In Section 2, the MBO and TSP are described in detail. Section 3 presents the proposed ATSP solution procedure using MBO. Section 4 compares the empirical studies and ATSP test results of MBO and GA. Section 5 is the conclusion of the article.

## 2. Material and Method.

**2.1. The marriage in honey bee optimisation (MBO) algorithm.** Honey bees are an important species in behavioural genetics [18]. Social insects demonstrate several interesting behaviours: division of labour, individual and group communication and association. These behaviours are due to a combination of the honey bees' genes, nest conditions and ecological environment. Studies on honey bees have revealed much information on molecular genetic problems and the complicated area of socio-genetics. The

male haploid structure enables a unique genetic analysis based on the presence of haploid/diploid individuals [18]. New swarm intelligence algorithms based on the haploid and diploid genetic developing operations known as MBO in honey bees have been created and improved for combinatorial optimisation problem solutions. We now present a brief explanation of honey bees' natural behaviours and artificial analogue models of them.

**Colony structure:** Normal honey bee colonies are comprised of a long lived queen that hatched from one typical egg, anywhere from zero to a few thousand drones (depends on the season) and in general 10,000 to 60,000 workers [19]. The queens specialise in laying eggs [19]. A colony may contain one or more queen, in which case it is called monogamous or polygamous, respectively. Only the queen bee is fed the coloured jelly like white milk called royal jelly. A queen bee lives 5 or 6 years, but worker bees and drones cannot live more than 6 months. Generally, they live together with a few hundred drones, a queen and worker bees in a colony. Drones are the fathers of the colony and are kept in the colony to provide sperm for the queen. They die after mating. Worker bees are responsible for protecting broods and preparing eggs. Broods develop from fertilised or unfertilised eggs into a queen, worker or drone [5].

**The mating-flight:** The mating flight starts with the queen's swinging dance. Next, the queen bee passes to the mating flight and drones follow her. In each flight, the virgin queen meets 7 or 20 drones. The sperm are taken from drones and saved in the spermatheca for use in fertilization. For the each fertilised egg, one spermatozoon from the sperm mixture in the spermatheca is chosen randomly [5]. When the mating is occurring, the queen is followed by a large drone swarm (male bee comet). Fertilisation ends with the death of the drones. The queen couples many times, but the drones do so only once. This feature makes the bees' mating process a perfect model in between insect.

**The artificial analogue model:** In the MBO model, the mating flight of the queen with the drones is followed by 5 steps: queen bees producing a new broods, worker bees caring for the health of the brood, the improvement of the broods' fitness by workers, the adaptation of the workers' fitness, and replacing the least fittest queen bee with the fittest brood [5].

The mating flight continues until the queen's energy is exhausted or until the spermatheca is filled; then, the queen returns to the nest for brood generation. The gene pool is subjected to crossbreeding by selecting one of the broods in the spermatheca at random, and the brood is produced by completing the queen's genes with this genome. After this operation, mutation is applied to completing the brood generation process. Next, worker bees improve the health of the broods. The workers' degrees of health are increased depending upon their improvement in the broods. As long as there are broods that are healthier than queen in the nest, the queen are replaced by the healthiest broods and the new mating flight cycle is begun by killing the remaining broods. This process is repeated until all the mating flights have finished or the halting condition is satisfied [5]. A flowchart for the MBO algorithm is given in Figure 1.

**2.2. Travelling salesman problem (TSP).** TSP is a problem that aims to find the shortest or the least cost tour that stops exactly once at each of  $N$  points, whose distances are known, (such as cities, parts, nodes). It is considered a discrete and combinatorial optimisation problem. The set of all solutions is the of size  $N!$ , but because  $N!$  is very large, searching the entire solution space would take a long time. For this reason, many methods have been developed for fast and effective solutions. Iterative improvement, branch-and-bound, based on linear coding, and branch-and-cut formed by the hybridisation of this method with plane cutting method are among the methods that produce exact solutions. As for the methods that produce heuristic-based solutions or approximate the best solution

```

Initialise workers; randomly generate the queens; apply local search
To obtain a better queen for a pre-defined maximum number of mating-flights
For each queen in the queen list
  Initialise energy, speed and position
  While the queen's spermatheca is not full and energy > threshold
    The queen moves between states and probabilistically chooses drones
    If a drone is selected, then
      Add its sperm to the queen's spermatheca
    End if
    Update the queen's internal energy and speed
  End while
End for each

Generate broods by crossover and mutation
Use workers to improve broods
Update workers' fitness

While the best brood is better than the worst queen
  Replace the least-fittest queen with the best brood
  Remove the best brood from the brood list
End while
Kill all broods

```

FIGURE 1. Original MBO algorithm [5]

in a reasonable time, the most important of the heuristic methods are genetic algorithms, simulated annealing, taboo search and ant system based algorithms [20].

2.2.1. *Exact algorithms.* In general, these algorithms are the approaches which are derived from integer linear programming formulas. However, these algorithms are overpriced in terms of computational complexity. The branch and bound algorithm is an example for this approach. The first method that comes to mind is the one each probability has been tried.

2.2.2. *Heuristic algorithms.* The performance of the exact algorithms may not be sufficient for practical purposes. Under these conditions, we can find a solution close to the ideal one without using the exact algorithms. In practical terms, heuristic algorithms are preferred to exact algorithms. There are three types of heuristic algorithms that solve the TSP: heuristics that form a tour, heuristics that improve the tour and heuristics in which these two methods are used in a hybrid way [1].

A common trait of the heuristic algorithms that form a tour is that they do not seek to improve a result after finding it but terminate at this point. The following are known heuristic algorithms: the nearest neighbour, Greedy Heuristic, Adding Heuristic and Christofides algorithms. The most optimum rates of these algorithms are between 10% and 15%. Other heuristic algorithms aim to improve an existing tour. In addition to the local optimisation algorithms, such as 2-opt, 3-opt and Lin-Kernighan, taboo search, genetic algorithms, and artificial intelligence methods such as simulated annealing and ant colony algorithms can also be given as examples of such methods (Potvin [1], Celik [2], Dorigo [21]). As for the hybrid methods, they are the ones that use both tour forming and tour developing heuristics together. Iterated Lin-Kernighan [2] is one of these algorithms. The most successful results have been obtained from hybrid methods.

### 2.3. TSP approach methods.

2.3.1. *Symmetric TSP.* In the travelling salesman problem (TSP), the tour is completed with a minimum total distance by visiting all other cities once and returning to the city from which it started (i.e., a Hamiltonian circle). Here, the distance matrix ( $C = \text{city}, i, j = \text{indise}$ ) is the  $C_{ij}$  and  $C_{ji}$  distances. If, for all cities,  $C_{ij}$  is equal to  $C_{ji}$ , this is called a symmetric travelling salesman problem [22].

2.3.2. *Asymmetric TSP.* In some cases, the cost of travelling from City  $C_{ij}$  to  $B$  is different from the cost of travelling from  $C_{ji}$  to  $C_{ij}$ . When situations such as one-way roads between some cities and the time difference caused by traffic jams are taken into consideration, the problem becomes an Asymmetric Travelling Salesman Problem (ATSP).

In the ATSP, there is no assumption that the distance between two cities should be the same in both directions. Thus, there are more choices for the salesman to visit a given location [22].

2.3.3. *Euclidean TSP (triangle inequality).* Given  $n$  nodes in  $R^d$ , the problem is to find the shortest tour that visits all nodes [4]. The problem has been approached directly mathematically. A city with the least distance to the others is chosen. This algorithm produces a single solution without any alternatives.

3. **Solving ATSP with MBO.** In this article, the asymmetric travelling salesman problem has been solved by the MBO algorithm. We now present the steps of this algorithm in detail.

3.1. **Generating a genotype.** As the travelling salesman problem is a sorting problem, in coding the genotype, the character series determines the order in which the cities will be visited. The genotype size has been taken as the number of cities, and the city order for each genotype is assigned randomly. The population size can be entered by default or by the user. The number of iterations can also be entered by default or by the user. For MBO, each genotype is represented as one mating of the queen bee. The population size that is settled in each flight is defined as the size of the queen's spermatheca, which she fills at the end of each flight.

To solve the ATSP with the MBO, the original MBO algorithm has been modified. The algorithm suggested for solving the ATSP with MBO is given below.

- Step 1.* The genotype size/city number ( $C$ ) is defined.
- Step 2.* A random genotype is produced and defined as the best genotype (best queen (Bq)).
- Step 3.* The iteration number/queen's flight number ( $N$ ) is set.
- Step 4.* The population/spermatheca size ( $M$ ) is set.
- Step 5.* Each genotype in the population is crossed with the best queen.
- Step 6.* The new genotype obtained is mutated.
- Step 7.* Elements of the new genotype, obtained after the mutation, are developed.
- Step 8.* The best genotype of the population is chosen.
- Step 9.* The best of the population and the best queen are compared, and the better one is named as the new queen.
- Step 10.* The procedures from 4 to 9 are repeated up to the desired number of iterations.
- Step 11.* The best solution is presented.

FIGURE 2. The algorithm suggested for solving the ATSP with MBO

**3.2. Crossover for MBO.** In the suggested study, two-point crossover is used and both points are produced randomly. Unlike in the GA algorithm, in the MBO algorithm, each genotype is crossed with the queen's genotype. In the new genotypes that are obtained, some cities may appear twice. In this case, because there is a requirement to visit each city only once in the TSP, the cities written twice are replaced in the crossover so that they are written only once.

**3.3. Mutation for MBO.** According to the mutation rate given before, the genotype is mutated at random points. The algorithm is carefully designed so that each city appears only once. Based on the concept of this algorithm, the longest city distance in each flight is assessed, and this city is exchanged with another city at random.

**3.4. The brood improvement by worker in MBO.** The furthest city is found in each of the genotypes obtained after mutation, and then, a reformation has been made in its genetic structure by replacing it with another city selected randomly.

**3.5. Speed and energy.** In the ATSP problem that we had selected, we calculated the sum of the energy spent in each tour, but as the energy is directly proportional to the total distance, this calculation does not affect the result. The speed variable has not been taken into consideration in problem solving.

**4. Experimental Results.** In this study, FULL-MATRIX(Fischetti) ftv33, ftv44, ftv64 are used from Standard asymmetric TSP libraries [23]. A program in the Delphi7 programming language was developed for the tests of MBO, SA and GA. The performance tests were performed with a laptop with Intel Core2 Duo 2.53 GHz, 4GB RAM. The software interface is given in Figure 3.

There are 33 cities in the FTV33, 44 cities in the FTV44, and 64 cities in the FTV64 library. Performance tests were executed for GA, SA and MBO. Parameters list is shown in Table 1.

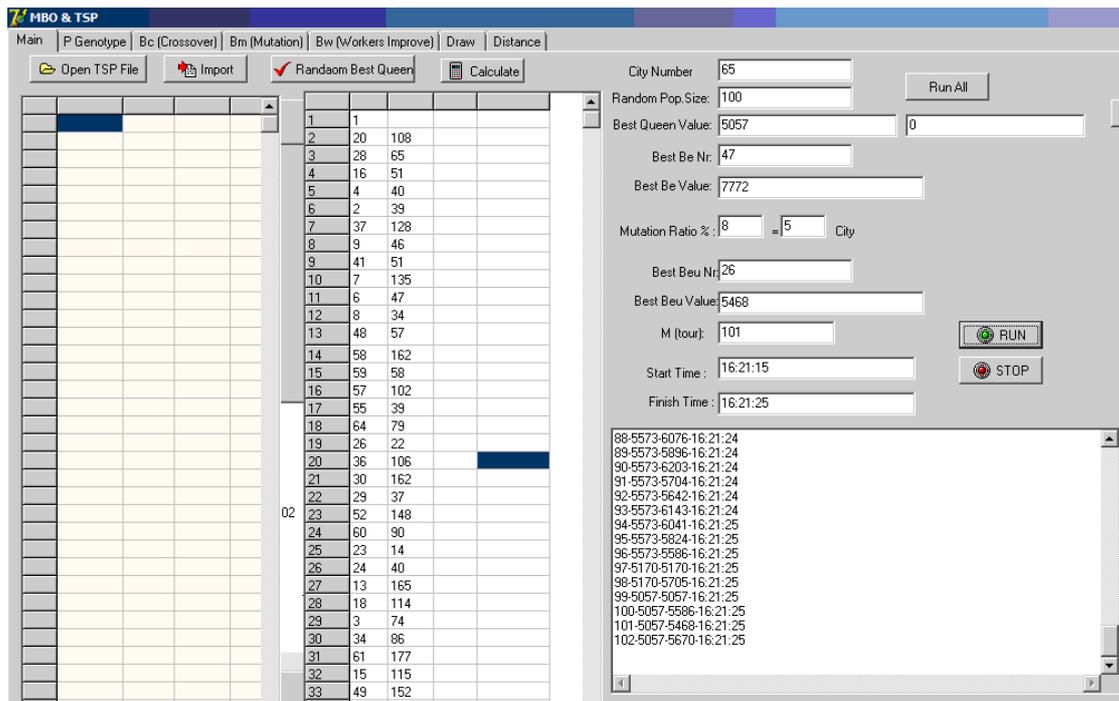
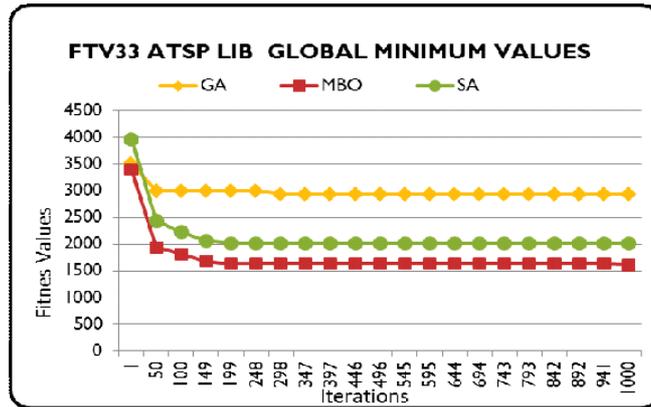


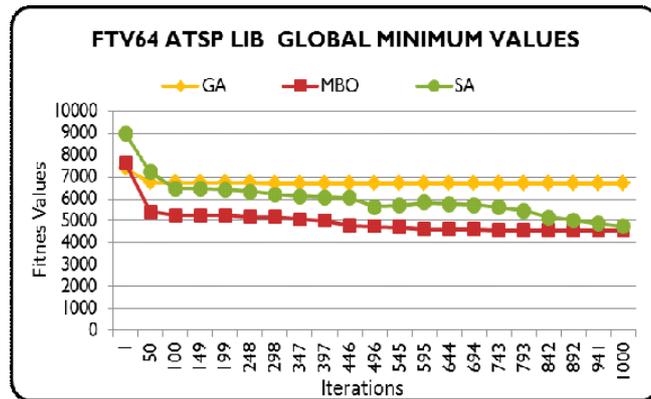
FIGURE 3. Delphi packaged programmed prepared for the test

TABLE 1. Parameters of GA, SA and MBO

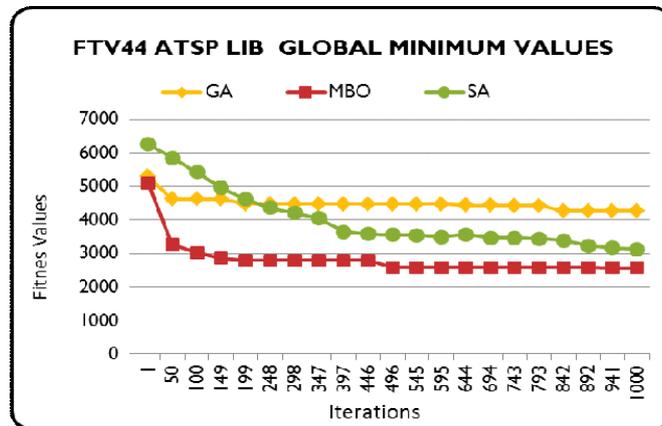
Parameter	GA	SA	MBO
Iteration	1000	1000	1000
Mutation Rate	8%	–	8%
Population	100	100 (L)	100
Cooling Value	–	0,8	–
Selection	Roulette Wheel	–	–



(a)

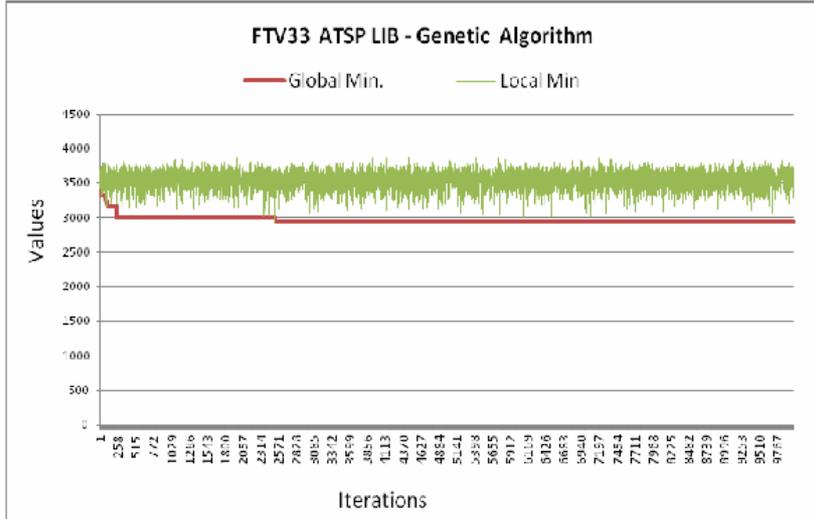


(b)

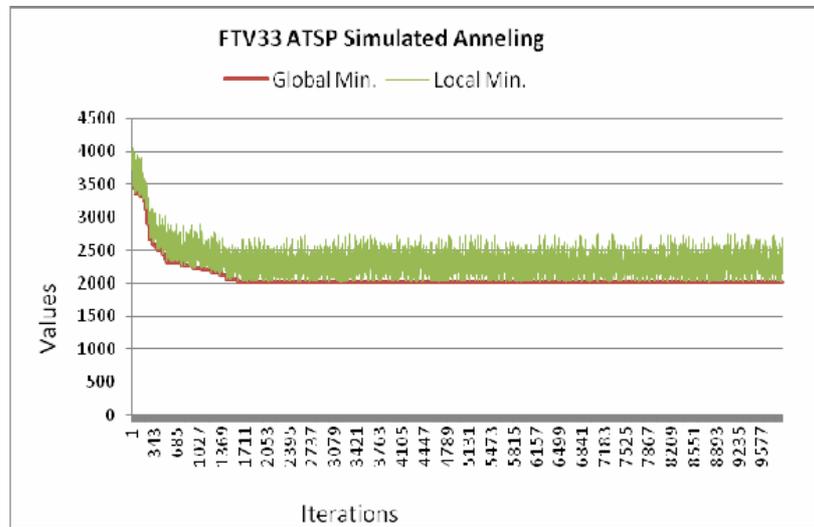


(c)

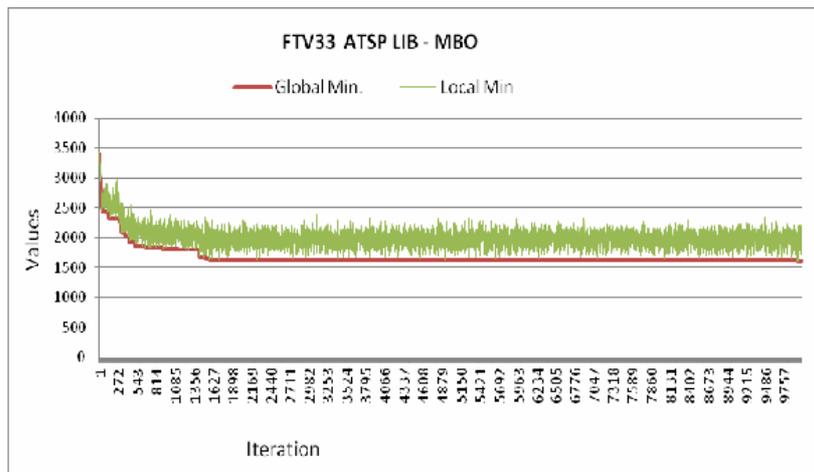
FIGURE 4. Global fitness values in GA, SA and MBO methods for (a) FTV33, (b) FTV64, (c) FTV44 ATSP



(a)



(b)



(c)

FIGURE 5. Global minimum and local minimum values for FTV33 ATSP: (a) according to GA method, (b) according to SA method, (c) according to MBO method

Depending on the change in the global fitness values of the MBO, SA and GA in each iteration are presented in Figure 4. Also, depending on the iteration, the global minimum and local minimum change for FTV33 are shown graphically in Figure 5(a) for GA, Figure 5(b) for SA, and in Figure 5(c) for MBO.

The test results show that the values obtained from the MBO optimisation mostly converge to the optimal solution. Moreover, it can be seen that MBO always reaches better global minimum rates than GA and SA in general and that due to its method of operation, the running time of MBO is longer than that of GA and SA.

**5. Conclusion.** In this study, the asymmetric travelling salesman problem, which has previously been solved by using heuristic methods, is attempted to be solved using Marriage in Honey Bee Optimisation, one of the swarm intelligence optimisation methods. To evaluate the performance of the solutions, we compared its results with those obtained from the same parameters and conditions using the genetic algorithm, which is a heuristic method. According to the test results, the MBO algorithm converges more to the optimum solution in fewer iterations.

In this article, the performance of MBO algorithm, one of the heuristic algorithms, has been compared with that of a GA and SA. In a future study, some tests will be carried out with other heuristic algorithms.

#### REFERENCES

- [1] Y. Potvin, Genetic algorithms for the traveling salesman problem, *Annals of Operations Research*, vol.63, pp.339-370, 1996.
- [2] Y. Celik and E. Ulker, Marriage honey bee optimization approach to the symmetric traveling salesman problem, *Energy Education Science and Technology Part B: Social and Educational Studies*, vol.4, no.2, pp.595-602, 2012.
- [3] C. Nilsson, Heuristic algorithms for travelling salesman problem, *Technical Report, Linkoping University: 609.12.2009*, <http://www.ida.liu.se/~TDDDB19/reports.2003/htsp.pdf>, 2009.
- [4] S. Arora, Polynomial time approximation schemes for Euclidean TSP and other geometric problems, *Proc. of the 37th Ann. Symp. Foundations of Computer Sci.*, pp.2-11, 1996.
- [5] A. Afshara, O. B. Haddada, M. A. Marino and B. J. Adams, Honey-bee mating optimization (HBMO) algorithm for optimal reservoir operation, *Journal of the Franklin Institute*, vol.344, pp.452-462, 2007.
- [6] C. F. Tsai, C. W. Tsai and C. C. Tseng, A new hybrid heuristic approach for solving large traveling salesman problem, *Information Sciences*, vol.166, no.1-4, pp.67-81, 2004.
- [7] D. S. Johnson and L. A. McGeoch, The traveling salesman problem: A case study in local optimization, in *Local Search in Combinatorial Optimization*, E. H. L. Aarts and J. K. Lenstra (eds.), 1997.
- [8] C. Choia, S. Kima and H. S. Kimb, A genetic algorithm with a mixedregion search for the asymmetric traveling salesman problem, *Computers & Operations Research*, vol.30, pp.773-786, 2003.
- [9] G. Ausiello, V. Bonifaci and L. Laura, The on-line asymmetric traveling salesman problem, *Journal of Discrete Algorithms*, vol.6, pp.290-298, 2008.
- [10] J. Brest and J. Zerovnik, A heuristic for the asymmetric traveling salesman problem, *Proc. of the 6th Metaheuristics International Conference*, pp.145-150, 2005.
- [11] Y.-W. Chen, Tomographic reconstruction of coded aperture images by simulated annealing, *International Journal of Innovative Computing, Information and Control*, vol.5, no.3, pp.697-705, 2009.
- [12] S. Gao, Z. Tang, H. Dai and J. Zhang, A hybrid clonal selection algorithm, *International Journal of Innovative Computing, Information and Control*, vol.4, no.4, pp.995-1008, 2008.
- [13] H. Takano, Y. Shirai and N. Matsumoto, Performance evaluation of hybrid procedure of self-organizing map and SA for TSP, *International Journal of Innovative Computing, Information and Control*, vol.7, no.5(B), pp.2931-2944, 2011.
- [14] M. Dorigo, V. Maniezzo and A. Coloni, The ant system: Optimization by a colony of cooperating ants, *IEEE Trans. Syst. Man. Cybern.*, vol.26, pp.29-42, 1996.
- [15] H. A. Abbass, Marriage in honey bees optimization: A haplometrosis polygynous swarming approach, *Proc. of the IEEE Congress on Evolutionary Computation*, pp.207-214, 2001.

- [16] H. A. Abbass and J. Teo, A true annealing approach to the marriage in honeybees optimization algorithm, *Proc. of the Inaugural Workshop on Artificial Life*, pp.1-14, 2001.
- [17] H. S. Chang, Converging marriage in honey-bees optimization and application to stochastic dynamic programming, *Journal of Global Optimization*, vol.35, pp.423-441, 2006.
- [18] R. F. A. Moritz and C. Brandes, Behavior genetics of honeybees, in *Neurobiology and Behavior of Honeybees*, R. Menzel and A. Mercer (eds.), Berlin, Germany, Springer-Verlag, 1987.
- [19] R. F. A. Moritz and E. E. Southwick, *Bees as Super-Organisms*, Springer, Berlin, Germany, 1992.
- [20] X. Hu, M. Huang and A. Z. Zeng, An intelligent solution system for a vehicle routing problem in urban distribution, *International Journal of Innovative Computing, Information and Control*, vol.3, no.1, pp.189-198, 2007.
- [21] M. Dorigo, Ant colonies for the traveling salesman problem, *TR/IRIDIA/3-1996*, Université Libre de Bruxelles, Belgium, 1996.
- [22] H. D. Sherali et al., A class of lifted path and flow-based formulations for the asymmetric traveling salesman problem with and without precedence constraints, *Discrete Optimization*, vol.3, pp.20-32, 2006.
- [23] *Asymmetric Traveling Salesman Problem Data*, <http://www2.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/atsp/>, 2011.