Water Resources Management (2006) 20: 661–680
DOI: 10.1007/s11269-005-9001-3

Honey-Bees Mating Optimization (HBMO) Algorithm: A New Heuristic Approach for Water Resources Optimization

OMID BOZORG HADDAD[1]*, ABBAS AFSHAR[1], and MIGUEL A. MARIÑO[2]
[1]*Dept. of Civil Engineering, Iran University of Science and Technology (IUST), Tehran, Iran;*
[2]*Hydrology Program and Dept. of Civil and Environmental Engineering, University of California, Davis, CA 95616*
(**author for correspondence, e-mail: haddad@iust.ac.ir)*

(Received: 7 December 2004; in final form: 21 September 2005)

Abstract. Over the last decade, evolutionary and meta-heuristic algorithms have been extensively used as search and optimization tools in various problem domains, including science, commerce, and engineering. Their broad applicability, ease of use, and global perspective may be considered as the primary reason for their success. The honey-bees mating process may also be considered as a typical swarm-based approach to optimization, in which the search algorithm is inspired by the process of real honey-bees mating. In this paper, the honey-bees mating optimization algorithm (HBMO) is presented and tested with few benchmark examples consisting of highly non-linear constrained and/or unconstrained real-valued mathematical models. The performance of the algorithm is quite comparable with the results of the well-developed genetic algorithm. The HBMO algorithm is also applied to the operation of a single reservoir with 60 periods with the objective of minimizing the total square deviation from target demands. Results obtained are promising and compare well with the results of other well-known heuristic approaches.

Key words: honey-bees mating optimization, genetic algorithm, heuristic search, non-linear optimization, single-reservoir operation

Introduction

Traditional optimization search methods may be classified into two distinct groups: direct-search and gradient-based search methods. In direct-search methods, only the objective function and constraint values are used to guide the search strategy, whereas gradient-based methods use the first and/or second-order derivatives of the objective function and/or constraints to guide the search process. Since derivative information is not used, direct-search methods usually require many function evaluations for convergence. For the same reason, they can also be applied to a variety of problems without a major change in the algorithm. In contrast, gradient-based methods often quickly converge to an optimal solution, but are not efficient in non-differentiable or discontinuous problems. In addition, there are some common difficulties with most of the traditional direct and gradient-based techniques, such as: (1) the convergence to a suboptimal solution, with pre-mature convergence; (2) an algorithm efficiency varies depending on the particular problem; (3) algorithms are not efficient in handling problems having discrete variables; and (4) algorithms cannot be efficiently used on a parallel machine, should they be deemed useful.

In most engineering problems, some variables may be restricted to take discrete values only. A usual practice to deal with such problems is to assure that all variables are continuous during the optimization process, choosing an available size closer to the obtained solution. In this case, the optimization algorithm must spend enormous time in computing infeasible solutions, causing an inefficient search effort. In addition, post-optimization calculations on a large number of discrete variables, and few other problems can be eliminated if only feasible values of the variables are allowed during the optimization process. Thus, for one reason or another, traditional search methods may not be good candidates as efficient optimization algorithms for a broad range of engineering design and operation problems. Over the last decade, evolutionary and meta-heuristic algorithms have been extensively developed and used as search and optimization tools in various problem domains. Among them, genetic algorithms (GAs) have been extensively employed as search and optimization methods in various problem domains, including science, commerce, biology, and engineering (Esat and Hall, 1994; Gen and Cheng, 1997; Wardlaw and Sharif, 1999). Particularly, codes are available for solving multimodal problems (Goldberg *et al.*, 1992), multi-objective problems (Jaszkiewicz, 2001), scheduling problems, as well as Neuro-Fuzzy-GA implementation (Brasil *et al.*, 1998).

Modeling the behavior of social insects, such as ants and bees, and using these models for search and problem-solving are the context of the emerging area of swarm intelligence. Ant colony is a typical successful swarm-based approach to optimization, where the search algorithm is inspired by the behavior of real ants. Ant colony algorithms as evolutionary optimization algorithms were first proposed by Dorigo (1992) and Dorigo *et al.* (1996) as a multi-agent approach to different combinatorial optimization problems like the traveling salesman problem and the quadratic assignment problem. Later, Dorigo and Di Caro (1999) introduced a general ant colony optimization algorithm (ACOA) namely ant colony meta-heuristic, which enables them to be applicable to other engineering problems (Dorigo *et al.*, 2000). Successful application of ACO to some water resources design and operation problems have been reported (Abbaspour *et al.*, 2001; Simpson *et al.*, 2001; Jalali *et al.*, 2006).

Honey-bees mating may also be considered as a typical swarm-based approach to optimization, in which the search algorithm is inspired by the process of mating in real honey-bees. The behavior of honey-bees is the interaction of their (1) genetic potentiality. (2) ecological and physiological environments, and (3) the social conditions of the colony, as well as various prior and ongoing interactions between these three parameters (Rinderer and Collins, 1986). Each bee undertakes sequences of actions which unfold according to genetic, ecological, and social conditions of the colony. Honey-bees are also used to model agent-based systems (Perez-Uribe and Hirsbrunner, 2000). In a recent work, Abbass (2001a, b), developed an optimization algorithm based on the honey-bees mating process.

In this paper, a honey-bees mating-based optimization algorithm is developed and its performance is tested using three well defined and highly nonlinear benchmark mathematical functions, as well as developing an optimum operation policy for a single reservoir.

Honey-Bee Colony Structure

A honey-bee colony typically consists of a single egg laying long-lived queen, anywhere from zero to several thousand drones (depending on the season) and usually 10,000 to 60,000 workers (Moritz and Southwick, 1992). The colony can be founded in two different ways (Dietz, 1986). In "independent founding" the colony starts with one or more reproductive females that construct the nest, lay the eggs, and feed the larvas. The first group of broods is reared alone until they take over the work of the colony. Subsequently, division of labor takes place and the queen specializes in egg laying and the workers in brood care (Dietz, 1986). Another founding method is called "swarming" in which a new colony is founded by a single queen or more, along with a group of workers from the original colony.

A colony of bees is a large family of bees living in one bee-hive. A bee hive is like a big city with many "sections of the town". The queen is the most important member of the hive because she is the one that keeps the hive going by producing new queen and worker bees. With the help of approximately 18 males (drones), the queen bee will mate with multiple drones one time in her life over several days. The sperm from each drone is planted inside a pouch in her body. She uses the stored sperms to fertilize the eggs. Whether a honeybee will become a queen, a drone, or a worker, depends on whether the queen fertilizes an egg. Since she is the only bee in the colony that has fully developed ovaries, the queen is the only bee that can fertilize the egg. Queens and workers come from fertilized eggs and drones from unfertilized eggs.

Only the queen bee is fed "royal jelly," which is a milky-white colored jelly-like substance. "Nurse bees" secrete this nourishing food from their glands, and feed it to their queen. The diet of royal jelly makes the queen bee bigger than any other bees in the hive. A queen bee may live up to 5 or 6 years, whereas worker bees and drones never live more than 6 months. There are usually several hundred drones that live with the queen and worker bees. Mother nature has given the drones just one task which is to give the queen some sperm. After the mating process, the drones die. As the nights turn colder and winter knocks the door, the drones still in the hive are forced out of the hive by worker bees. It is a sad thing, but the hive will not have enough food if the drones stay.

Queens represent the main reproductive individuals which are specialized in eggs laying (Laidlaw and Page, 1986). Drones are the fathers of the colony. They are haploid and act to amplify their mothers' genome without altering their genetic composition, except through mutation. Workers are specialized in brood care and sometimes lay eggs. Broods arise either from fertilized or unfertilized eggs. The former represent potential queens or workers, whereas the latter represent prospective drones.

The mating process occurs during mating-flights far from the nest. A mating-flight starts with a dance where the drones follow the queen and mate with her in the air. In a typical mating-flight, each queen mates with seven to twenty drones. In each mating, sperm reaches the spermatheca and accumulates there to form the genetic pool of the colony. Each time a queen lays fertilized eggs, she retrieves at random a mixture of the sperms accumulated in the spermatheca to fertilize the egg (Page, 1980). Insemination ends with the eventual death of the drone, and the queen receiving the "mating sign." The queen mates multiple times but the drone inevitably only once. These features make bees-mating the most spectacular mating among insects.

Honey-bees Modeling

The mating–flight may be considered as a set of transitions in a state-space (the environment) where the queen moves between the different states in some speed and mates with the drone encountered at each state probabilistically. At the start of the flight, the queen is initialized with some energy content and returns to her nest when her energy is within some threshold from zero or when her spermatheca is full.

In developing the algorithm, the functionality of workers is restricted to brood care (i.e., nurse bees), and therefore, each worker may be represented as a heuristic which acts to improve and/or take care of a set of broods (i.e., as feeding the future queen with royal jelly). A drone mates with a queen probabilistically using an annealing function as (Abbass, 2001a):

$$\text{Prob}\,(Q, D) = e^{-\frac{\Delta(f)}{S(t)}} \tag{1}$$

where Prob (Q, D) is the probability of adding the sperm of drone D to the spermatheca of queen Q (that is, the probability of a successful mating); $\Delta(f)$ is the absolute difference between the fitness of D (i.e., $f(D)$) and the fitness of Q (i.e., $f(Q)$); and $S(t)$ is the speed of the queen at time t. It is apparent that this function acts as an annealing function, where the probability of mating is high when either the queen is still in the start of her mating–flight and therefore her speed is high, or when the fitness of the drone is as good as the queen's. After each transition in space, the queen's speed, $S(t)$, and energy, $E(t)$, decay using the following equations:

$$S(t+1) = \alpha \times S(t) \tag{2}$$

$$E(t+1) = E(t) - \gamma \tag{3}$$

where α is a factor $\in [0, 1]$ and γ is the amount of energy reduction after each transition. Thus, an Honey-Bees Mating Optimization (HBMO) algorithm may be constructed with the following five main stages (Abbass, 2005a):

1. The algorithm starts with the mating–flight, where a queen (best solution) selects drones probabilistically to form the spermatheca (list of drones). A drone is then selected from the list at random for the creation of broods.
2. Creation of new broods (trial solutions) by crossoverring the drones' genotypes with the queen's.
3. Use of workers (heuristics) to conduct local search on broods (trial solutions).
4. Adaptation of workers' fitness based on the amount of improvement achieved on broods.
5. Replacement of weaker queens by fitter broods.

Solution Representation (Working Principle)

In the mathematical representation, a drone is represented by a genotype and a genotype marker. Realizing the fact that all drones are naturally haploid, a genotype marker may be employed to randomly mark half of the genes, leaving the other half unmarked. In this case, only the unmarked genes are those that form a sperm to be randomly used in the mating process.

Workers which are used to improve the brood's genotype, represent a set of different heuristics. The rate of improvement in brood's genotype, as a result of heuristic application to that brood, defines the heuristic fitness value. As an example, in one-point crossover heuristic, the crossover heuristic operator applies to the brood's genotype with that of a randomly generated genotype where the crossover point is also selected at random.

The queens play the most important role in the mating process in nature as well as in the HBMO algorithm. Each queen is characterized with a genotype, speed, energy, and a spermatheca with defined capacity. Spermatheca is defined as a repository of drones' sperm after the mating process with the queen. Thus, for a queen, spermatheca size is defined and kept constant during the mating flights. On the other hand, speed and energy are initialized before each mating flight, at random in the range of (0.5, 1). Since the drones' are assumed to be haploid, after successful mating, the drones' sperm is stored in queens' spermatheca. Later in breeding process, a brood is constructed by copying some of the drones' genes into the brood genotype and completing the rest of the genes from the queens' genome.

The fitness of the resulted genotype is determined by evaluating the value of the objective function of the brood genotype and/or its normalized value. It is important to note that a brood has only one genotype.

The algorithm starts with three user-defined parameters and one predefined parameter. The predefined parameter is the number of workers, representing the number of heuristics encoded in the program. However, the predefined parameter may be used as a user parameter to alter the number of active heuristics if required; that is, the user may choose the first heuristic, where the number of workers is less than or equal to the total number of heuristics encoded in the program. The three user-defined parameters are the number of queens, the queen's spermatheca size (representing the maximum number of matings per queen in a single mating-flight), and the number of broods that will be born by all queens.

Figure 1 shows a computational flowchart and translation of biological and natural processes in honey-bees mating into an algorithm. This figure clearly maps biological processes into a mathematical representation as well as identifying the steps taken in the optimization process.

A set of queens and their energy and speed at the start of each mating-flight is then initialized at random. A randomly selected heuristic is used to improve the genotype of each queen, assuming that a queen is usually a good bee. A number of mating-flights are then undertaken. In each mating-flight, all queens fly based on the energy and speed of each, where both energy and speed are generated at random for each queen before each mating flight commences. At the start of a mating-flight, a drone is generated at random and the queen is positioned over that drone. The transition made by the queen in space is based on her speed which represents the probability of flipping each gene in the drone's genome. At the start of a mating-flight, the speed may be higher and the queen may make very large steps in space. While the energy of the queen decreases, the speed decreases and as a result the neighborhood covered by the queen decreases. At each step in space, the queen mates with the drone encountered at that step using the probabilistic rule in Equation (1). If the mating is successful (i.e., the drone passes the probabilistic decision rule), the drone's sperm is stored in the queen's spermatheca. One may note that each time a drone is generated, half of his genes are marked at random, to make them inactive, since each drone is haploid by definition. Therefore, the genes that will be transmitted to the broods are fixed for each drone.

When all queens complete their mating-flight, they start breeding. For a required number of broods, a queen is selected in proportion to her fitness and mated with a randomly selected sperm from her spermatheca. A worker is chosen in proportion to its fitness to improve the resultant brood. After all broods are being generated, they are sorted according to their fitness. The best brood replaces the worst queen until there is no brood that is better than any of the queens. Remaining broods are then killed and a new mating-flight starts until all assigned mating-flights are completed or convergence criteria met. The main steps in a HBMO algorithm are presented in Figure 1.

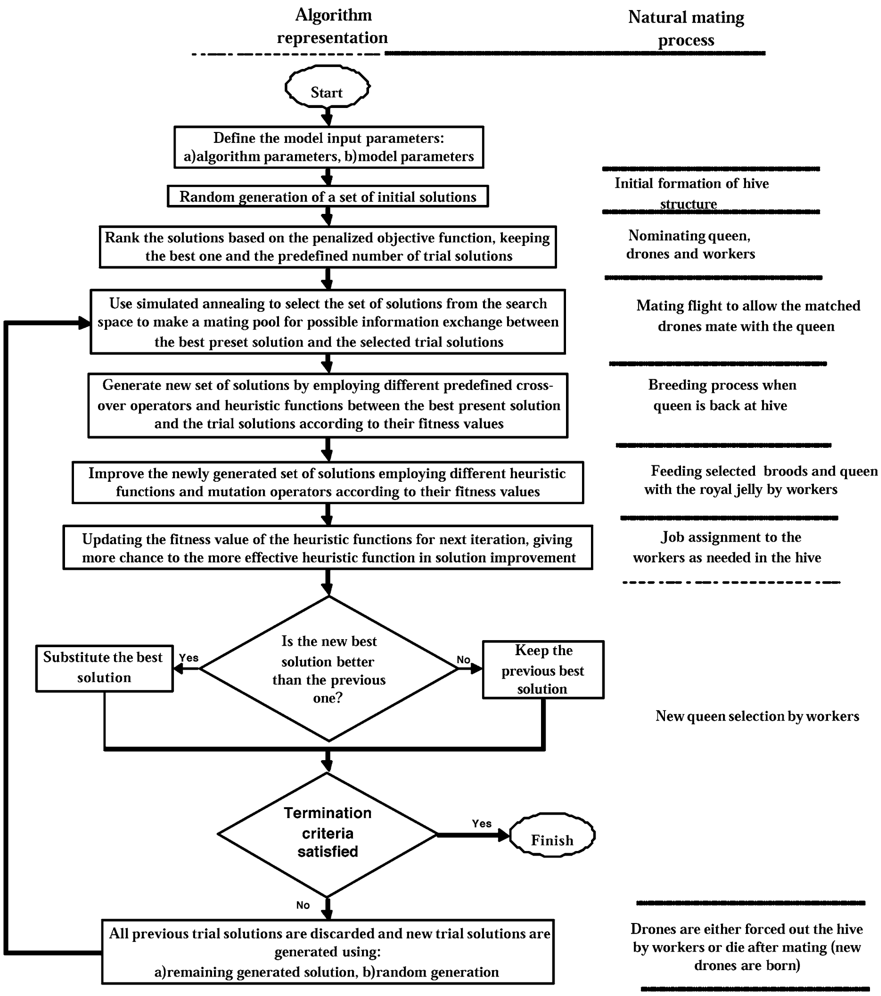


Figure 1. Algorithm and computational flowchart and algorithm translation of natural processes.

Algorithm Application

To test the performance of the proposed algorithm, it was applied to several benchmark constrained and unconstrained mathematical optimization functions. The first example of unconstrained optimization is Ackley's function, a continuous and multi-modal test function obtained by modulating an exponential function with a cosine wave of moderate amplitude. Its topology is characterized by an almost flat outer region and a central hole or peak where modulations by cosine wave become more and more influential. Ackley's function is:

$$\text{Min} f(x_1, x_2) = -c_1 \cdot \exp\left(-c_2 \sqrt{\frac{1}{2}\sum_{j=1}^{2} x_j^2}\right) - \exp\left[\frac{1}{2}\sum_{j=1}^{2}\cos(c_3 x_j)\right] + c_1 + e \tag{4}$$

$$-5 < x_j < 5 \quad j = 1, 2 \tag{5}$$

where $c_1 = 20, c_2 = 0.2, c_3 = 2\pi$, and e $= 2.71282$. This function causes moderate complications to the search, because a strictly local optimization algorithm that performs hill-climbing would surely get trapped in a local optimum (Figure 2). A search strategy that scans a slightly bigger neighborhood would be able to cross intervening valleys toward increasingly better optima. Therefore, Ackley's function provides one of the reasonable test cases for the honey-bees mating search algorithm. Employing the proposed HBMO algorithm, the fitness value is $f(x_1^*, x_2^*) = -0.005164$, obtained as an average of 10 runs. More detail is presented in Table I. Using GA, at the 1000th generation, the fitness value of $f(x_1^*, x_2^*) = -0.005456$ has been obtained (Gen and Cheng, 1997). The best, worst, and average rate of convergence for 10 runs is presented in Figure 3. The best run converges to the optimal solution with less than 200 mating flights. However, the worst run converges with 500 mating flights. Very low standard deviation of the solutions for 10 runs may be considered as a small discrepancy of the final solutions. Convergence to a near optimal solution as a function of number of function evaluations, employing the HBMO algorithm and a developed GA, is presented in Figure 4. In most of the cases, the HBMO algorithm converged to a near optimal solution much faster than GA, resulting in a slightly better final solution.

The second numerical example of unconstrained optimization is (Gen and Cheng, 1997):

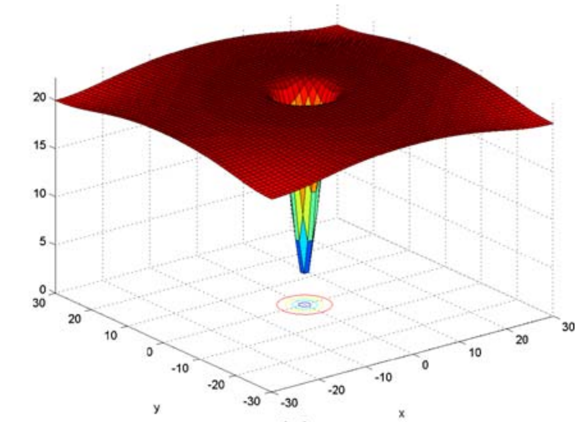
$$\text{Max} f(x_1, x_2) = 21.5 + x_1 \sin(4\pi x_1) + x_2 \sin(20\pi x_2) \tag{6}$$

$$-3.0 \le x_1 \le 12.1 \tag{7}$$

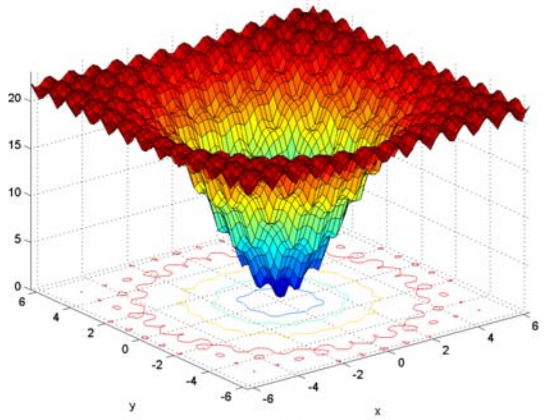
$$4.1 \le x_2 \le 5.8 \tag{8}$$

Table I. Results of three different problems, with their statistical measures

Example number	Spermatheca size	Max no. of mating flight	Best fitness value	Worst fitness value	Average over 10 runs	Standard deviation
1	300	500	−0.005390	−0.004654	−0.005164	0.000236
2	300	500	38.850300	38.850290	38.850294	0.000005
3	300	1000	13.590840	13.782440	13.628688	0.079275



(a)



(b)

Figure 2. Surface defined by Ackley's function for (a) $-30 \leq x_1, x_2 \leq 30$ and (b) $-6 \leq x_1, x_2 \leq 6$.

As is clear from Figure 5, the search space is a highly non-linear and multi-modal surface. Again, by employing the proposed HBMO algorithm, the best fitness value was obtained as 38.850300 with an average over 10 runs of 38.850294, indicating a very small standard deviation (Table I). The best, worst, and average rate of convergence for 10 runs is presented in Figure 6. Solving the same problem with GA, the best run was terminated after 1,000 generations, obtaining the best chromosomes in the 419$^{\text{th}}$ generation as follows (Gen and Cheng, 1997):

$$\text{eval}(v^*) = f(11.631407, 5.724824) = 38.818208 \tag{9}$$

Results from the GA and HBMO algorithm converge well with minor improvement in the HBMO solution. All 10 runs have almost converged to the global optimal

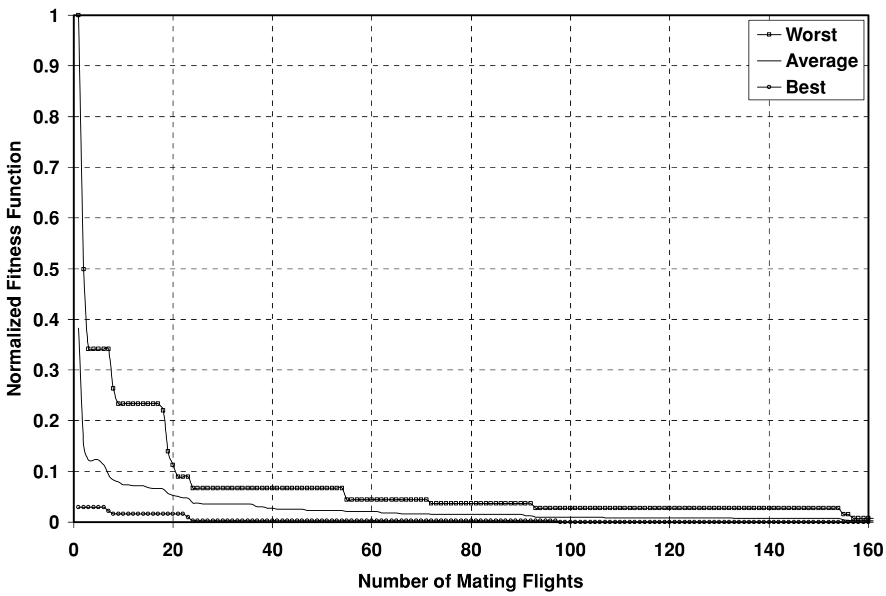


Figure 3. Rate of convergence of first example problem for the best, worst, and average over 10 runs.

solution within 200 to 500 mating flights. Standard deviation of the final results is practically zero (Table I). To compare the rate of convergence in the HBMO algorithm, a GA was also developed. Convergence of the objective function is presented in Figure 7. Once again for all ranges of number of function evaluations, the HBMO algorithm performed slightly better than the GA used for this purpose.

To test the performance of the proposed algorithm in handling constrained models, it was applied to a two-variable, two-constraint nonlinear programming problem as (Figure 8):

$$\text{Min} f_1(x_1, x_2) = \left(x_1^2 + x_2 - 11\right)^2 + \left(x_1 + x_2^2 - 7\right)^2 \tag{10}$$

s.t.:

$$g_1(x) \equiv 5.062 - x_1^2 - (x_2 - 2.5)^2 \geq 0 \tag{11}$$

$$g_2(x) \equiv (x_1 - 0.05)^2 + (x_2 - 2.5)^2 - 4.84 \geq 0 \tag{12}$$

$$0 \leq x_1 \leq 6, \quad 0 \leq x_2 \leq 6 \tag{13}$$

The unconstrained objective function $f_1(x_1, x_2)$ has a minimum solution at (3, 2) with a function value equal to zero. However, due to the presence of constraints, this solution is not feasible and the constrained optimal solution is $x^* = (2.2461, 2.38154)$ with a function value equal to $f_1^* = 13.61227$. The feasible region is a narrow crescent-shaped region (approximately 0.7% of the total search space) with the optimum solution lying on the second constraint.

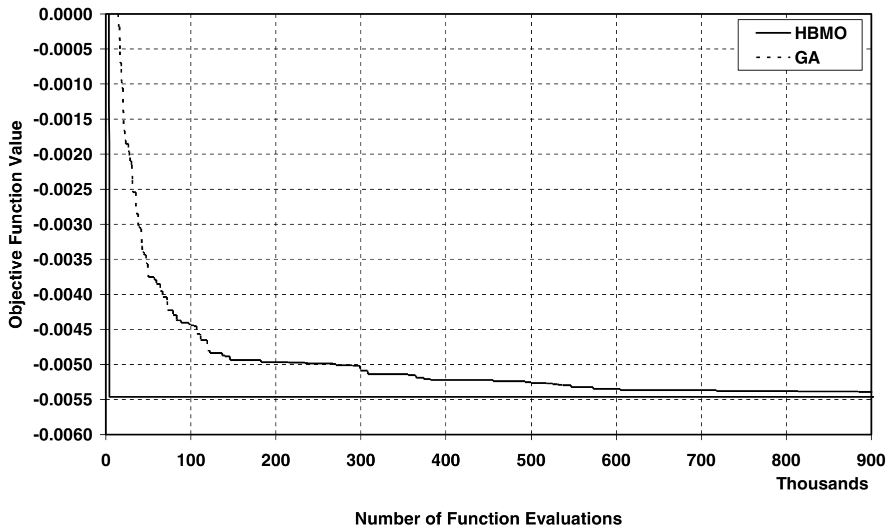


Figure 4. Convergence to a near optimal solution as a function of number of function evaluations for the first example (averaged over 10 runs).

Employing the same algorithm, the average fitness value over 10 runs was obtained as $f_1(x_1^*, x_2^*) = 13.628688$, with the best result as low as 13.590840. Details are provided in Table I. Figure 9 shows how the HBMO solutions converge to a narrow region of feasible solutions and finally to the true optimum solution for 10 different runs. Clearly, the best and the worst solutions reveal a very rapid rate of convergence to the near optimal solution. Again all ten runs show a very small discrepancy with the final result as indicated by a very small value of the standard deviation (Table I). The rate of convergence to a near-optimal solution for the proposed HBMO algorithm and GA is presented in Figure 10. Regardless of slight variations in the rate of convergence, after 6 million function evaluations, the HBMO algorithm ended up with a better performance in minimizing the defined constrained function.

Single Reservoir Operation Optimization

To illustrate the model application and performance, operation of the Dez reservoir in southern Iran was selected as a case study. Monthly historical inflow to the reservoir along with monthly projected demand for a 5-year period is presented in Figure 11. Average annual inflow to the reservoir and annual demand are estimated as 5,900 $\times 10^6$ m^3 and 5,303 $\times 10^6$ m^3, respectively. The effective storage volume of the reservoir of 2,510 $\times 10^6$ m^3 was discretized uniformly into 14 discrete levels. The objective of the study is to minimize the total squared deviation *(TSD)* of

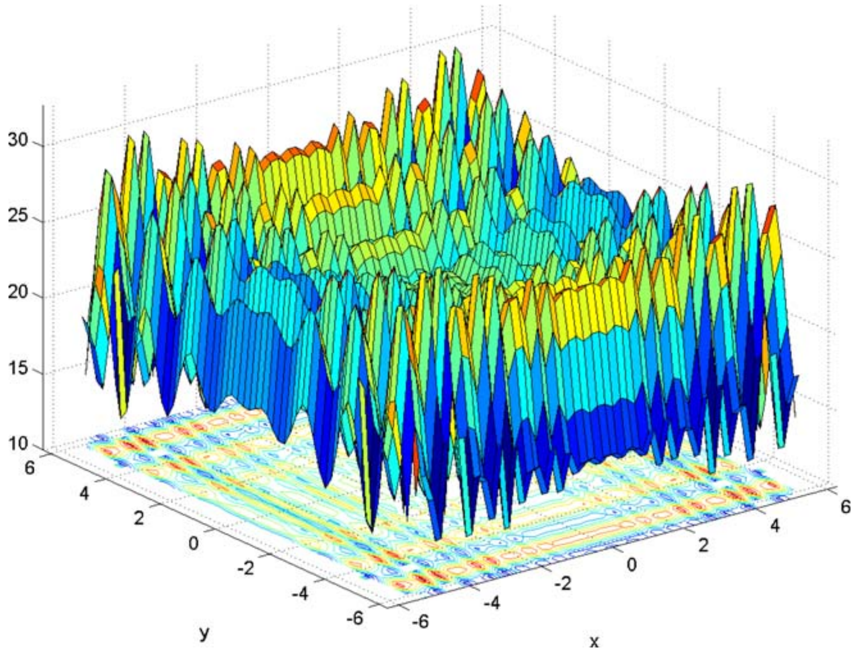


Figure 5. Surface defined by the second example.

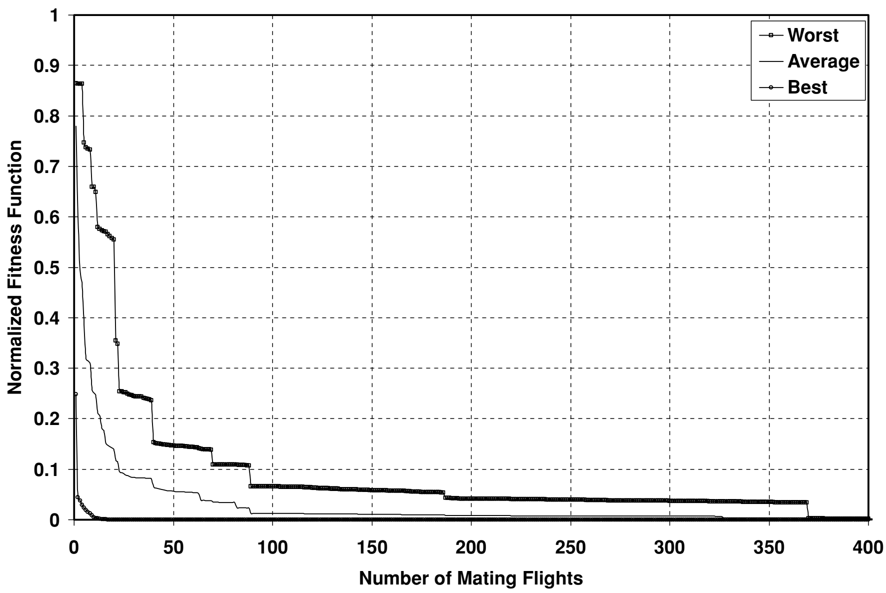


Figure 6. Rate of convergence of second example problem for the best, worst, and average over 10 runs.

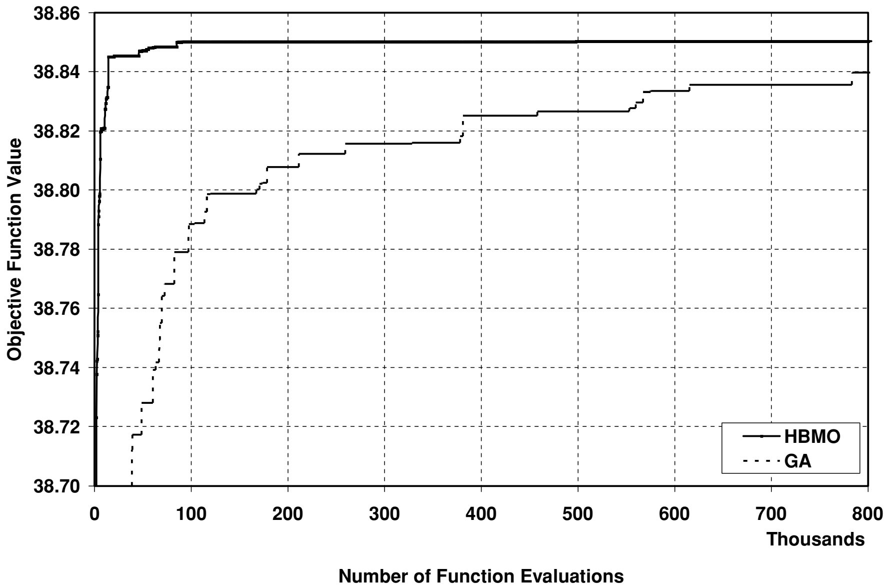


Figure 7. Convergence to a near optimal solution as a function of number of function evaluations for the second example (averaged over 10 runs).

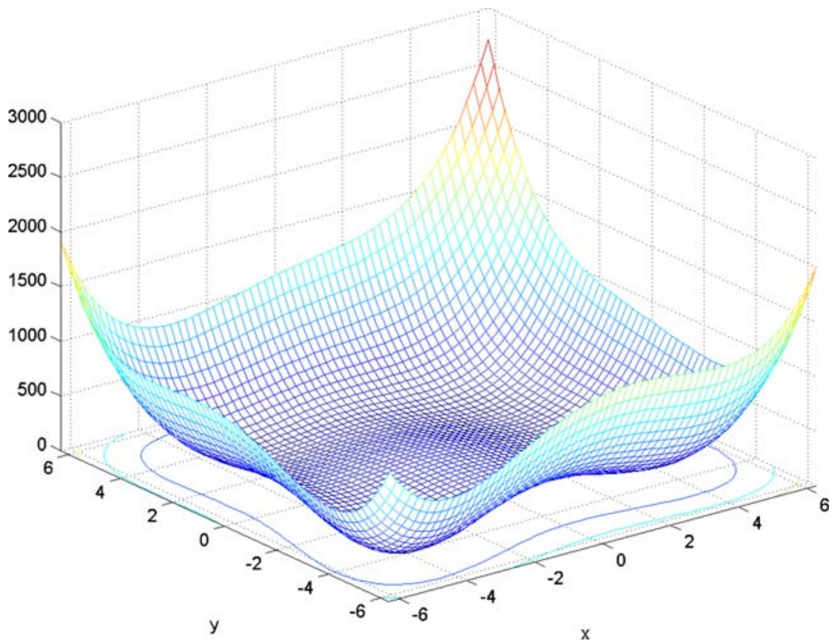


Figure 8. Surface defined by the third example.

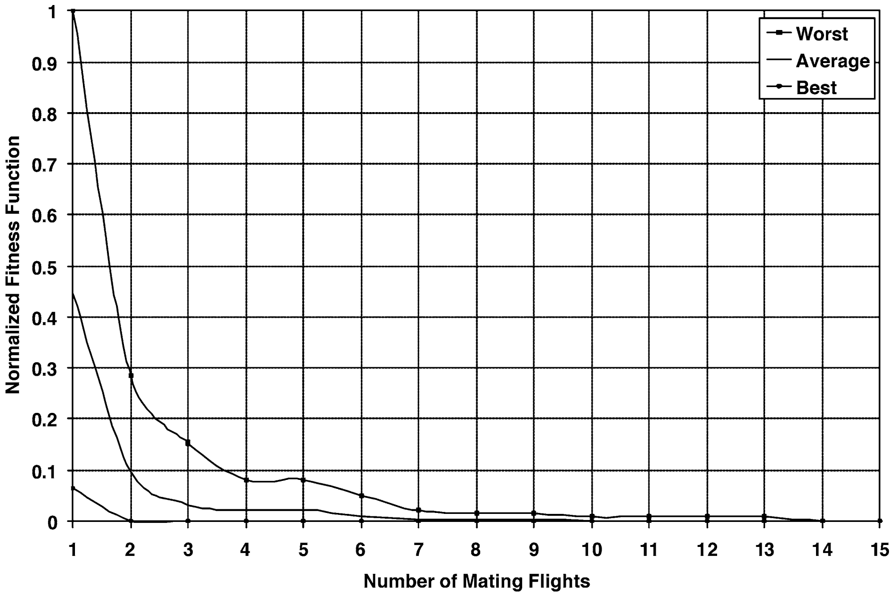


Figure 9. Rate of convergence of third example problem for the best, worst, and average over 10 runs.

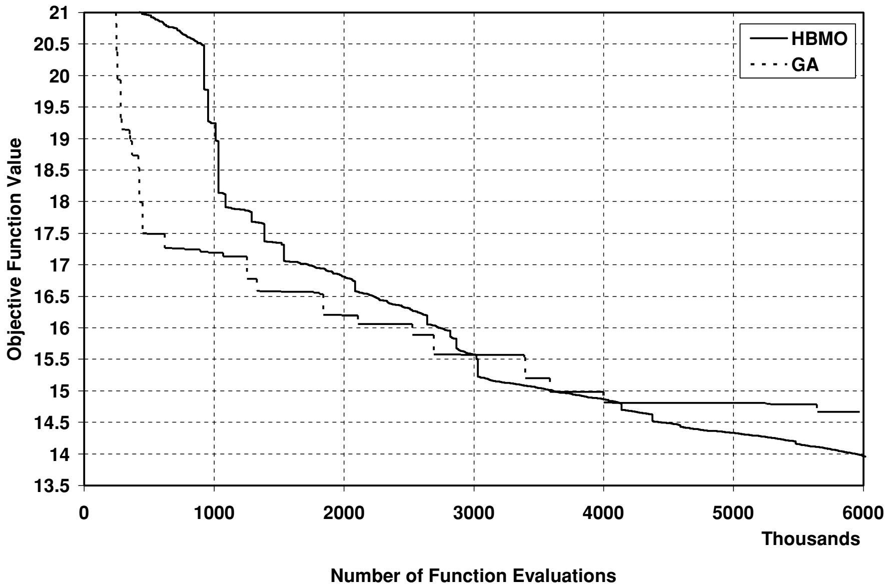


Figure 10. Variation of the objective function with number of function evaluations using HBMO and CPGA for the third example (averaged over 10 runs).

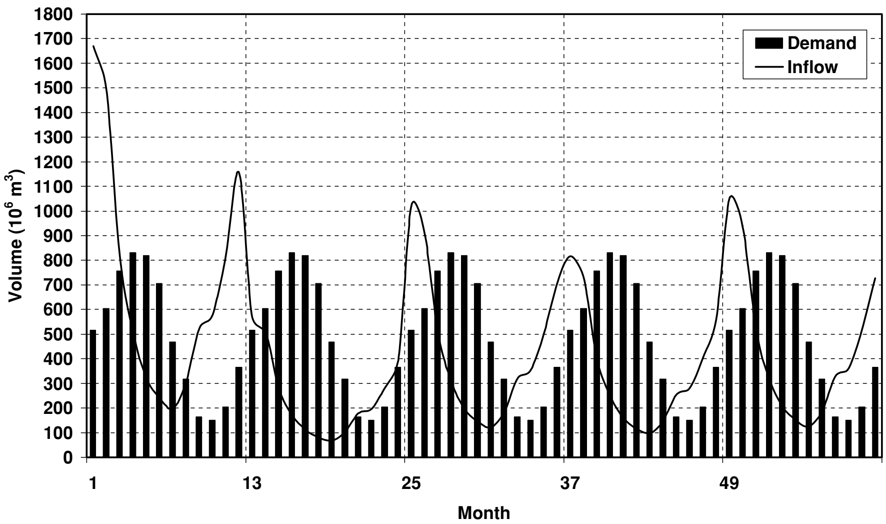


Figure 11. Monthly inflow to the reservoir along with monthly demand.

releases (R_t) from the target demands (D_t).

$$\text{Min}\, TSD = \sum_{t=1}^{nt} \left(\left(R_{(t)} - D_{(t)} \right) / D_{\max} \right)^2 \tag{14}$$

s.t.:

$$S(t) = S(t+1) - Q(t) + R(t); \quad \forall t \tag{15}$$

$$R_{\min}(t) \leq R(t) \leq R_{\max}(t); \quad \forall t \tag{16}$$

$$S_{\min}(t) \leq S(t) \leq \text{Cap}; \quad \forall t \tag{17}$$

$$S(1) = S_{\min} \tag{18}$$

In this problem, one queen with 160 drones were employed in each mating flight (or iteration), with the total number of mating flights and queen's spermatheca capacity limited to 50 and 30, respectively. Results of the model for storage volume at the end of each period, for the best run, are presented in Figure 12. For the same problem, along with the global optimum, monthly releases resulting from the HBMO model with 50 mating flights (or iterations) is presented in Figure 13. Monthly demand and the global optimum results are presented in the same figure. In order to have a notion of the rate of convergence of the model, Figure 14 is presented. Very rapid convergence, as well as comparable *TSD* from the target demands makes the approach and algorithm quite promising for further development and application in the field of water resources planning and management. To be specific, results of 10 different runs, with their statistical measures are presented in Table II. One may

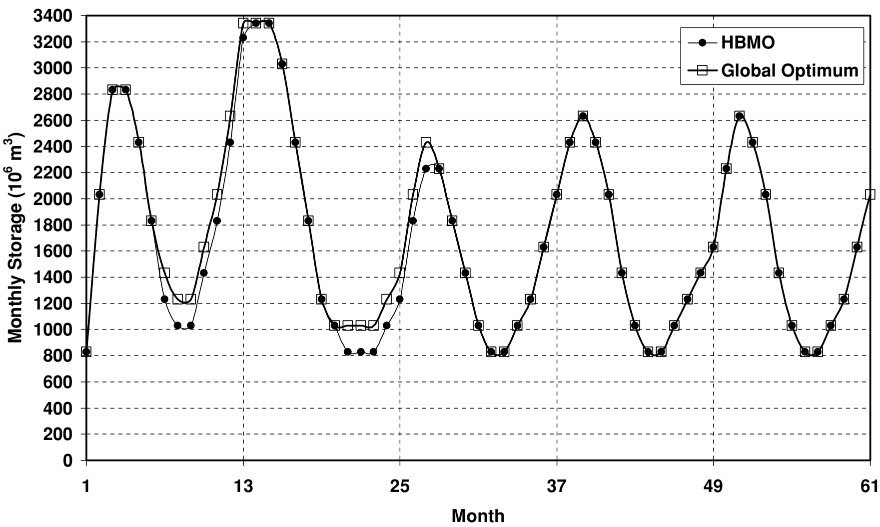


Figure 12. Storage volume at the end of each month.

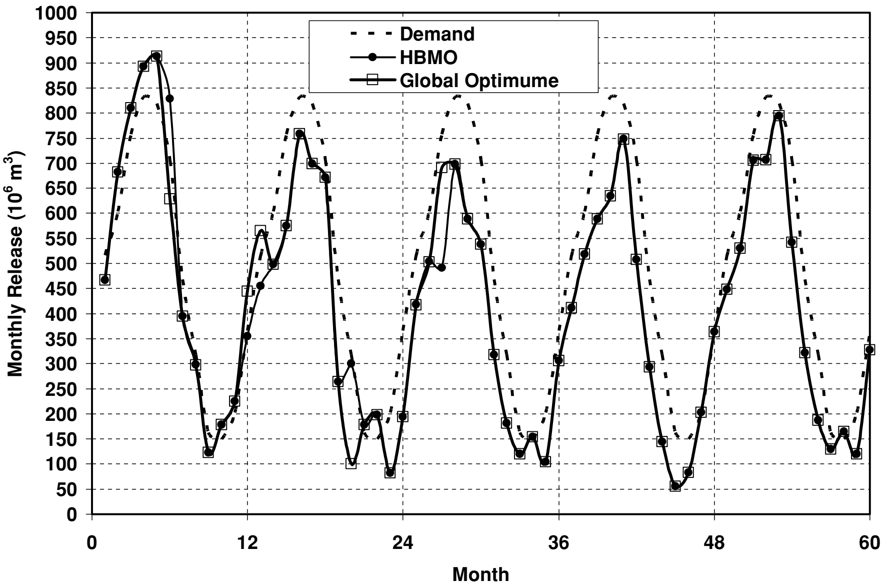


Figure 13. Monthly releases resulting from the HBMO model and the global optimum.

note that the global optimum *TSD* from target demands is 1.07, which is less than 3 percent from the best result of the HBMO algorithm.

To test the effect of the discretization scheme on the final solution, the entire search space was discretized into 3, 6, and 12 uniform grids. Results are depicted in Figure 15 for different number of function evaluations. For the finer discretized

Table II. Reservoir operation problem: Ten different runs with their statistical measures

Iteration number	1	2	3	4	5	6	7	8	9	10	Mean	Min.	Max.	Standard deviation	Coefficient of variation
Value of fitness function	1.32	1.34	1.14	1.27	1.28	1.14	1.43	1.10	1.24	1.34	1.26	1.10	1.43	0.11	0.084

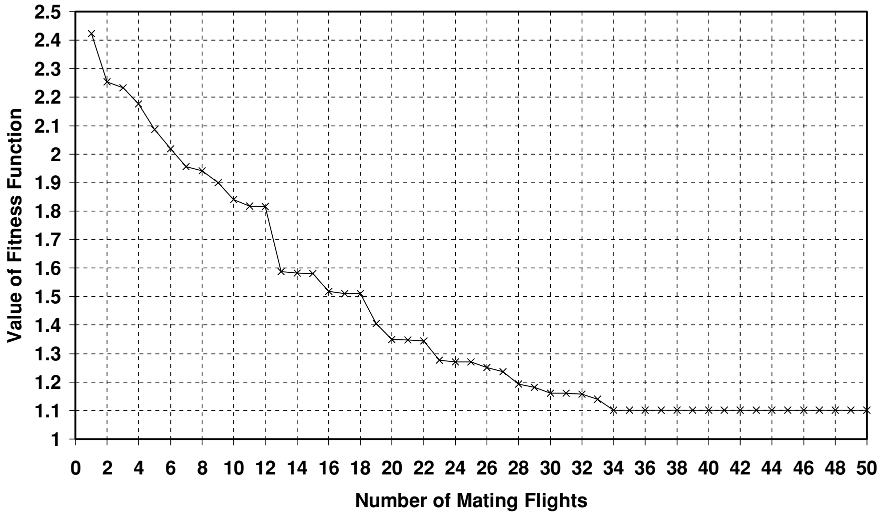


Figure 14. Rate of convergence of the model in reservoir operation problem to a near optimal solution.

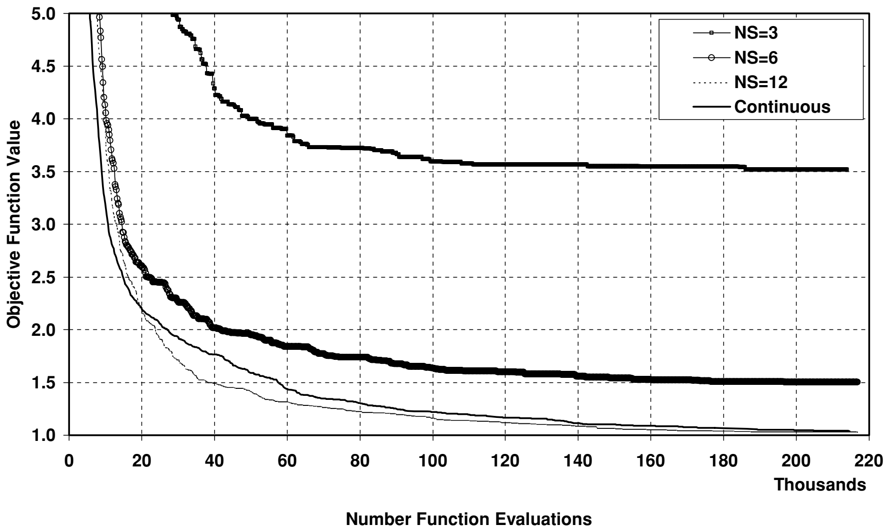


Figure 15. Effect of discretization on the results of reservoir operation problem.

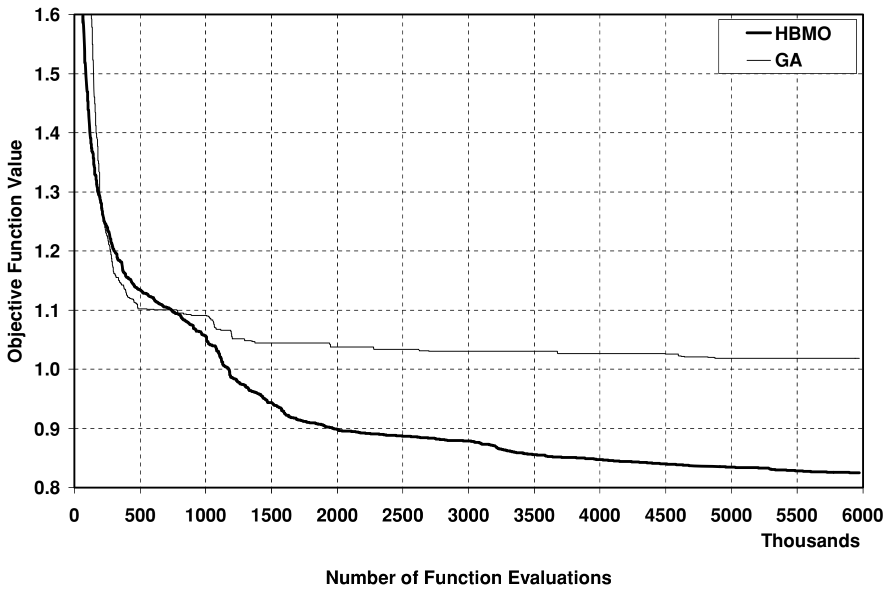


Figure 16. Convergence to a near optimal solution as a function of number of function evaluations in reservoir operation problem (averaged over 10 runs).

scheme, the final results approach the near-optimal solution from a real value coding for a continuous search space. Figure 16 shows the performance of the proposed HBMO algorithm compared with that of the GA. As is clear, for the best run, after 6 million function evaluations, the HBMO generated a significantly better solution. It is interesting to mention that the significantly better performance of the HBMO in the last 4.5 million function evaluations may mainly be attributed to the active contribution of heuristic functions employed in the breeding and queen's feeding process.

Concluding Remarks

HBMO as a search hybrid algorithm is inspired by the process of real honey-bees mating. A very limited attempt has been made to employ honey-bees' social behavior in real-world optimization. The modeling of honey-bees' mating process as an optimization algorithm and its application to several highly nonlinear constrained and unconstrained optimization problems, partially revealed the high potential of the proposed algorithm to solve nonlinear optimization problems. A mating flight is considered as a set of transition in a state-space environment in which the queen mates with the drones probabilistically. An annealing function defines the probability of mating drones with the queen where the number of predefined heuristic functions improves the generated solutions.

Results obtained are well comparable with those obtained by well developed GAs. The model performance in a real-world reservoir operation problem is promising. Test application of the algorithm revealed its capacity in conducting an extensive search in the entire search space. The algorithm performed quite well in problems with combination of discrete and real-valued decision variables.

References

Abbaspour, K. C. and Schulin, R., and van Genuchten, M. T., 2001, 'Estimating unsaturated soil hydraulic parameters using ant colony optimization', *Adv. Water Resour.* 24(8), 827–933.

Abbass, H. A., 2001a, 'Marriage in honeybees optimization (MBO): A haplometrosis polygynous swarming approach', *in The Congress on Evolutionary Computation, CEC2001, Seoul*, Korea, May 2001, pp. 207–214.

Abbass, H. A., 2001b, 'A monogenous MBO approach to satisfiability, in The International Conference on Computational Intelligence for Modelling', Control and Automation, CIMCA'2001, Las Vegas, NV, USA.

Brasil, L. M., de Azevdo, F. M., Barreto, J. M., and Noirhomme, M., 1998, 'Training algorithm for Neuro-Fuzzy-GA systems', in *Proc. 16th IASTED International Conference on Applied Informatics*, AI'98, Garmisch-Partenkirchen, Germany, pp. 45–47.

Dietz, A., 1986, 'Evolution', in T. E. Rinderer (ed.), *Bee Genetics and Breeding*, Academic Press Inc., N.Y. pp. 3–22.

Dorigo, M., 1992, 'Optimization, learning and natural algorithms', Ph.D. Thesis, *Politecnico di Milano*, Milan, Italy.

Dorigo, M., Bonabeau, E., and Theraulaz, G., 2000, 'Ant algorithms and stigmergy', *Future Generation Computer Systems* 16, 851–871.

Dorigo, M. and Di Caro, G., 1999, 'The ant colony optimization metaheuristic', in D. Corne, M. Dorigo, and F. Glover (eds.), New Ideas in Optimization, McGraw-Hill, Maidenhead, London, pp. 11–32.

Dorigo, M., Maniezzo, V., and Colorni, A., 1996, The ant system: Optimization by a colony of cooperating ants, *IEEE Trans. Syst. Man. Cybern.* 26, 29–42.

Esat, V. and Hall, M. J., 1994, 'Water resources system optimization using genetic algorithms', in Hydroinformatics' 94, *Proc., 1st Int. Conf. on Hydroinformatics*, Balkema, Rotterdam, The Netherlands, pp. 225–231.

Gen, M. and Cheng, R., 1997, '*Genetic Algorithm and Engineering Design*', John Wiley and Sons, N.Y.

Goldberg, D. E., Deb, K., and Horn, J., 1992, 'Massive multimodality, deception, and genetic algorithms', in R. Manner and B. Manderick (eds.), *Parallel Problem Solving from Nature*, Elsevier: Amsterdam, 2, pp. 37–46.

Jalali, M. R., Afshar, A., and Mariño, M. A., (2006). 'Reservoir operation by ant colony optimization algorithms.' *Iranian Journal of Science and Technology, Shiraz, Iran*, in press.

Jaszkiewicz, A., 2001, 'Multiple objective metaheuristic algorithms for combinatorial optimization, Habilitation Thesis', *Poznan University of Technology*, Poznan.

Laidlaw, H. H. and Page, R.E., 1986, 'Mating designs, in T. E. Rinderer (ed.), *Bee Genetics and Breeding*', Academic Press, Inc., pp. 323–341.

Moritz, R. F. A. and Southwick, E. E., 1992, *Bees as Superorganisms*, Springer Verlag, Berlin, Germany.

Page, R. E., 1980, 'The evolution of multiple mating behavior by honey bee queens (Apis mellifera L.)', *Journal of Genetics* 96, 263–273.

Perez-Uribe, A. and Hirsbrunner, B., 2000, 'Learning and foraging in robot-bees, in Meyer, Berthoz, Floreano, Roitblat and Wilson (eds.)', *SAB2000 Proceedings Supplement Book, Intermit. Soc. for Adaptive Behavior*, Honolulu, Hawaii, pp. 185–194.

Rinderer, T. E. and Collins, A. M., 1986, 'Behavioral genetics, in T. E. Rinderer (ed.), *Bee Genetics and Breeding*', Academic Press, Inc., pp. 155–176.

Simpson, A. R. Maier, H. R., Foong, W. K., Phang, K. Y., Seah, H. Y., and Tan, C. L., 2001, 'Selection of parameters for ant colony optimization applied to the optimal design of water distribution systems', in Proc., *Int. Congress on Modeling and Simulation*. Canberra, Australia, pp. 1931–1936.

Wardlaw, R. and Sharif, M., 1999, 'Evaluation of genetic algorithms for optimal reservoir system operation', *J. Water Resour. Plng. and Mgmt.* ASCE, 125(1), 25–33.