Induction motor parameter estimation using disrupted black hole artificial bee colony algorithm

Fani Bhushan Sharma* and
Shashi Raj Kapoor

Department of Electrical Engineering,
Rajasthan Technical University,
Kota, India
Email: fbsharma.kota@gmail.com
Email: srkapoor@rtu.ac.in
*Corresponding author

Abstract: The most widespread motors in industries are induction motors nowadays. The design, performance evaluation and control of induction motors are based on circuit parameters. The accurate measurement of electrical parameters, like resistance (or reactance), is a tedious job. So, researchers have found induction motor parameter estimation as a significant optimisation aspect. Though conventional techniques produce good results, the swarm intelligence-motivated techniques produce still better results for real-world optimisation problems nowadays. In this paper, swarm intelligence-motivated artificial bee colony (ABC) algorithm is modified with physics's phenomena. The proposed algorithm is named as disrupted black hole ABC (DBHABC) algorithm. Further, this proposed algorithm is applied for optimising induction motor parameter estimation. The obtained outcomes reveal that DBHABC may be a good choice for induction motor parameter estimation.

Keywords: black hole; disruption; induction motor; metaheuristics; real-world optimisation; swarm intelligence.

Reference to this paper should be made as follows: Sharma, F.B. and Kapoor, S.R. (2017) 'Induction motor parameter estimation using disrupted black hole artificial bee colony algorithm', *Int. J. Metaheuristics*, Vol. 6, Nos. 1/2, pp.85–106.

Biographical notes: Fani Bhushan Sharma is a Research Scholar at Rajasthan Technical University, Kota, India. He received BE and MTech from University of Rajasthan, India and Rajasthan Technical University, Kota, India, respectively. His research area include soft computing and electrical machines.

Shashi Raj Kapoor is a Professor at Rajasthan Technical University, Kota, India in Department of Electrical Engineering. He received BE and MTech from Rajasthan University, India and IIT Roorkee, India, respectively. He obtained PhD from RGPV, Bhopal, India. His primary area of interest is embedded system, machines and power drives.

1 Introduction

The electrical energy is also one of the bases of socio-economic development of any country (Bazmi and Zahedi, 2011). The three-fourth part of electrical energy is consumed by electrical motors in industrial sector (Dandil et al., 2013; Sakthivel, Bhuvaneswari and Subramanian, 2011). So, for energy conservation techniques, motor's energy consumption is a significant issue. Induction motors require low maintenance, less space and are easy to control (Lindenmeyer et al., 2001). Therefore, the induction motors are the most commonly used motors in industries. Eventually, enlightening the parameters of induction motor plays vital role in its designing, evaluating performance and application control. Though there are many methods for estimating induction motor parameters, iterative methods are most widespread and reasonable (Gupta, Wadhwani and Kapoor, 2011; Lindenmeyer et al., 2001; Pedra and Corcoles, 2004; Toliyat et al., 2003). The swarm intelligence (SI)-motivated algorithms are also iterative schemes. The significant SI algorithms are artificial bee colony (ABC) algorithm (Karaboga, 2005a), particle swarm optimisation (PSO) algorithm (Eberhart et al., 1995), bat swarm optimisation (BSO) algorithm (Yang, 2010), water cycle algorithm (WCA) (Eskandar et al., 2012) etc. The above metaheuristic techniques are a recent field of interest to solve real-world optimisation problems (Nesmachnow, 2014b, 2014a). The ABC algorithm carries out superior results than many other state-of-the-art methods (El-Abd, 2012; Karaboga and Akay, 2009; Karaboga and Basturk, 2008). This algorithm has already been applied on various real-world problems (Chandrasekaran et al., 2012; Haluk Gozden, 2011; Samanta and Chakraborty, 2011).

On the other hand, ABC algorithm may incline towards local optima stagnation and shows drawback in terms of sluggish convergence. To conquer these drawback, researchers have proposed a number of ABC variants (Akay and Karaboga, 2012; Banharnsakun, Achalakul and Sirinaovakul, 2011; Bansal et al., 2013; Bansal, Sharma and Jadon, 2013; Bansal et al., 2013; Sharma et al., 2015, 2016; Sharma, Bansal and Arya, 2014; Sharma et al., 2015; Zhu and Kwong, 2010). The WSO, BSO and PSO algorithms are also significantly modified, respectively, as chaotic WCA (Heidari, Abbaspour and Jordehi, 2015), Chaotic BSO (Jordehi, 2015a) and enhanced leader PSO (Jordehi, 2015b). In the same series of modifications, the authors of this paper presented a new variant of ABC algorithm. The proposed variant is based upon a amalgamation of two significant phenomena of physics and named as disruption black hole ABC (DHABC). The black hole ABC (BHABC) (Hatamlou, 2013; Sharma et al., 2015) increases exploitation, while disruption ABC (DiABC) (Sharma et al., 2016; Sarafrazi, Nezamabadi-Pour and Saryazdi, 2011) enhances both exploitation and exploration. In the proposed DBHABC, an optimum balance between exploitation and exploration is kept upon and positive aspect of both BHABC and DiABC is retained. Further, the proposed algorithm is applied on two test cases for estimating parameters of induction motor.

The remainder of this paper is organised as follows: Parameter estimation of induction motor is outlined in Section 2. Section 3 presents overview of ABC and the proposed algorithm is discussed. The results analysis, discussion and comparison are presented in Section 4. Finally, conclusion of the work is given in Section 5.

2 Parameter estimation of induction motor

The induction motor parameters can be calculated by any approximate or exact model. The input parameters are voltage, speed, starting torque, full-load torque and maximum torque, while the measured parameters are the rotor and stator resistances, reactances and magnetising reactances.

2.1 Approximate model

The approximate model is less accurate as the magnetising reactance and rotor reactance are neglected in the approximate model. The modelling is done using following objective function [11].

$$min(F) = f_1^2 + f_2^2 + f_3^2, \tag{1}$$

here;

$$f_1 = \frac{K_t \times R_2}{s[(R_1 + \frac{R_2}{s})^2 + X_1^2]} - T_{FL}(mf), \tag{2}$$

$$f_2 = \frac{K_t \times R_2}{(R_1 + R_2)^2 + X_1^2} - T_{STR}(mf), \tag{3}$$

$$f_3 = \frac{K_t}{2[R_1 + \sqrt{(R_1^2 + X_1^2)}]} - T_{MAX}(mf), \tag{4}$$

$$K_t = \frac{3V_{ph}^2}{\omega_s}, \tag{5}$$

Resistance, reactance, slip, manufacturer's data, synchronous speed and phase voltage are denoted with R, X, s, mf, ω_s and V_{ph}, respectively; subscript 1 is used for stator, while 2 is used for rotor; T_{FL}, T_{STR} and T_{MAX} are full load, starting and maximum torques, respectively.

2.2 Exact model

It is comparatively a high-accurate model, since both magnetising and rotor reactance are considered here with the parameters of approximate model.

$$min(F) = f_1^2 + f_2^2 + f_3^2 + f_4^2, \tag{6}$$

here;

$$f_1 = \frac{K_t \times R_2}{s[(R_{th} + \frac{R_2}{s})^2 + X_1^2]} - T_{FL}(mf), \tag{7}$$

$$f_2 = \frac{K_t \times R_2}{(R_{th} + R_2)^2 + X_1^2} - T_{STR}(mf), \tag{8}$$

$$f_3 = \frac{K_t}{2[R_1 + \sqrt{(R_{th}^2 + X_1^2)}]} - T_{MAX}(mf), \tag{9}$$

$$f_4 = \cos\left(tan^{-1}\left(\frac{X}{R_{th} + \frac{R_2}{s}}\right)\right) - PF(mf), \tag{10}$$

$$V_{th} = \frac{V_{ph} X_m}{X_1 + X_m}, \tag{11}$$

$$R_{th} = \frac{R_1 X_m}{X_1 + X_m}, \tag{12}$$

$$X_{th} = \frac{X_1 X_m}{X_1 + X_m}, \tag{13}$$

$$X = X_2 + X_{th}, \tag{14}$$

$$K_t = \frac{3V_{th}^2}{\omega_s}, \tag{15}$$

Resistance, reactance, slip, manufacturer's data, synchronous speed, phase voltage, thévenin voltage and power factor are denoted with R, X, s, mf, ω_s, V_{ph}, V_{th} and PF, respectively; subscript 1 is used for stator, 2 is used for rotor, while m is used for magnetising; T_{FL}, T_{STR} and T_{MAX} are full load, starting and maximum torques, respectively. In exact model objective function optimisation, values of obtained parameters and deviation amid estimated and manufacturer's values of torque and power factor should be within a specified range.

3 Disruption black hole artificial bee colony algorithm

3.1 Artificial bee colony algorithm

The ABC algorithm is an unique optimisation algorithm that comes under the category of SI. ABC is inspired by the collective intelligent foraging activities of the natural bees (Karaboga and Basturk, 2007).

In ABC algorithm, food source's position represents a possible solution for the optimisation problem and the nectar amount of a food source corresponds to the fitness of the solution (Karaboga and Akay, 2009). The colony of the artificial bees is partitioned into three groups, namely employed bees, onlooker bees and scout bees. The number of onlooker bees or employed bees is equal to the number of food sources. A bee waiting for employed bees for taking decision about how to pick the food source is known as onlooker bee (Abu-Mouti and El-Hawary, 2012; Karaboga, 2005b).The employed bees randomly search for the positions of the food source and share its experience with the onlooker bee which stays at hive. Scout bees search the new food sources randomly depending upon the internal motivation (Abu-Mouti and El-Hawary, 2012).

ABC is an iterative process alike other population-based metaheuristic algorithms. It requires cycles of the four phases, namely initialisation of the population phase, employed bee phase, onlooker bee phase and scout bee phase (Akay and Karaboga, 2012). The explanation of the phases is given below:

Initialisation of the population phase: Initially ABC generates an evenly scattered initial population of SN solutions, where each solution x_i ($i = 1, 2, \dots$; SN) is a D-dimensional vector. Here D is the number of variables in the optimisation problem, and x_i is the ith food source in the population. Generation of each food source is as follows:

$$x_{ij} = x_{minj} + rand[0, 1](x_{maxj} - x_{minj}), \tag{16}$$

where x_{minj} and x_{maxj} are bounds of x_i in jth direction and $rand[0, 1]$ is an evenly scattered random number in the range [0, 1].

Employed bee phase: During this phase, the current solution is modified based on the information provided by the experience of individual and the fitness value of the new solution, i.e. nectar amount. If the fitness value of the new solution is higher than that of the old solution, the bee updates its position with the new one and discards the old one (Akay and Karaboga, 2012). For ith candidate, the position update equation in this phase is

$$v_{ij} = \phi_{ij}(x_{ij} - x_{kj}), \tag{17}$$

where $k \in \{1, 2, \dots, SN\}$ and $j \in \{1, 2, \dots, D\}$ are randomly chosen indices. k must be different from i. ϕ_{ij} is a random number between $[-1, 1]$.

Onlooker bee phase: The information is shared by all the employed bees about the new fitness, i.e. nectar of the new solutions (food sources) and their position information with the onlooker bees in the hive. The available information is analysed by onlooker bees and they select a solution with a probability, related to its fitness. The probability p_i may be calculated using either following expression or there may be some other function, but must be a function of fitness:

$$p_i = \frac{fit_i}{\sum_{i=1}^{SN} fit_i}, \tag{18}$$

where fit_i is the fitness value of the solution i. Same as in the case of the employed bee, it generates a modification in the position in its memory and checks for the fitness of the candidate source. If the new fitness is higher than that of the previous one, the bee memorises the new generated position and forgets the old one.

Scout bee phase: The food source is assumed to be abandoned if the position of a food source is not updated up to a predetermined limit, i.e. number of cycles, and then scout bee phase starts. In this phase, food source is replaced by a randomly chosen food source within the specified region. Assume that the abandoned source is x_i and $j \in \{1, 2, \dots, D\}$ then the scout bee replaces this food source with x_i. This operation can be defined as follows:

$$x_i^j = x_{min}^j + rand[0, 1](x_{max}^j - x_{min}^j), \tag{19}$$

where x_{min}^j and x_{max}^j are bounds of x_i in jth direction.

The above discussion shows that there are three control parameters in ABC search process: first, the number of food sources SN (equal to number of onlooker or employed bees), second, the value of $limit$ and third, the maximum number of cycles MCN.

In the ABC algorithm, the exploitation process is carried out by employed bees and onlooker bees, while the exploration process is carried out by scout bees in the search space. The pseudocode of the ABC algorithm is as follows:

3.2 Disruption black hole artificial bee colony algorithm

In this paper, two significant phenomena of physics are combined, while keeping optimum balance in exploration and exploitation to modify the ABC algorithm.

3.2.1 Black hole phenomenon

The black hole (BH) phenomenon was invented by John Michell and Pierre Laplace (Montgomery, Orchiston and Whittingham, 2009). After the collapse of a massive star in the space, there is always a possibility for the BH to come in the picture (Doraghinejad et al., 2012). A lot of mass is scattered in the BH in such a way that the gravitational power of a BH becomes so high that any object which crosses its boundary will be swallowed by it and that object permanently dies out. Even the light cannot escape its gravitational pull. The boundary of the BH is called the event horizon and the radius of the event horizon is termed as the Schwarzschild radius (Doraghinejad et al., 2012; Hatamlou, 2013; Zhang et al., 2008) which is calculated using Eq. (20).

$$R_s = \frac{2GM}{C^2}, \tag{20}$$

where G is the gravity constant, M is the mass of the BH and C represents the velocity of the light. If any object moves close to the event horizon or crosses the Schwazschild radius, it will be absorbed into the BH and permanently fade away.

3.2.2 Disruption phenomenon in nature

When a group of gravitationally bounded particles (having total mass m) approaches very near to a massive object M, the group of particles tends to be torn apart. This is called disruption phenomenon in nature (Sarafrazi, Nezamabadi-Pour and Saryazdi, 2011). The same thing occurs in the solid body also. Disruption phenomenon in astrophysics is defined as *The sudden inward fall of a group of particles that are gravitationally bound under the effect of the force of gravity* (Ding, Liu and He, 2013; Sarafrazi, Nezamabadi-Pour and Saryazdi, 2011). It occurs when all the forces fail to supply a sufficient high pressure to counterbalance the gravity and keep the massive body in equilibrium (Liu, Ding and Sun, 2012).

3.3 Proposed DBHABC algorithm

In working of ABC algorithm, stagnation is a situation where the candidate solution stop exploring the new regions and work in a very narrow region. Due to the stagnation, algorithm lost its energy of finding better solution. Stagnation may cause algorithm to converge prematurely or stuck to the local optima. The stagnation phenomena occurs when all or most of the candidate solutions come very close to each other. In other words, when the sum of distances between any two candidate solutions become very small. In case of stagnation, to make ABC algorithm more explorative, concept of BH is introduced to basic version of ABC (Hatamlou, 2013).

For simulating the BH phenomenon, all candidate solutions are considered as stars, while the solution having best fitness among all candidate solutions is chosen as a BH. Further, for simulating the disruption phenomenon, a new phase called disruption phase is introduced within the ABC. In the proposed phase, the disruption operator initially explores the search space and as time passes it switches to the exploiting conditions. The proposed strategy is named as disruption black hole artificial bee colony algorithm (DBHABC). In DBHABC, it is assumed that the solution having best fitness value initially acts as BH and then is nominated as a star, and rest of the candidate solutions is scattered in the search space under the gravity force of the star solution (Sharma et al., 2015, 2016).

Initially all the solutions are randomly initialised in the given search space as shown in Eq. (21):

$$x_{ij} = x_{minj} + rand[0,1](x_{maxj} - x_{minj}), \tag{21}$$

where x_i represents the ith food source in the swarm, x_{minj} and x_{maxj} are bounds of x_i in jth dimension and $rand[0,1]$ is an uniformly distributed random number in the range [0, 1]. BH creation is not a randomised procedure. During the each iteration of the proposed algorithm, a new solution is generated near to the best solution and rest of the candidates is moved towards the black hole that depends upon the current position and a random number.

After the initialisation phase, the fitness values of all the candidate solutions are evaluated and the solution, having the best fitness value in the current swarm, is nominated as a black hole, and rest of the solutions form the ordinary stars. The BH is considered as an estimate of the actual optimal solution. After initialising the BH and stars, the stars are attracted by the BH, i.e. the solutions update their positions using the distance and direction of the BH (best solution found so far). The position update equation of the solutions is expressed by Eq. (22):

$$x_i(t+1) = x_i(t) + rand(x_{BH} - x_i(t)), \tag{22}$$

where $x_i(t)$ and $x_i(t+1)$ represent the position of the ith solution during iteration t and $t+1$, respectively. x_{BH} is the position of the BH (best solution) in the search region and $rand$ is an uniformly distributed random number specified in the interval [0, 1].

The BH search strategy is described as follows: While moving towards the BH, a star (solution) may reach to a position, which may be better than the position of the black hole. In such a case, the positions of the BH and the star are interchanged, i.e. the star is nominated as a new BH of the search space. In addition, when a star moves towards a BH, there is always possibility of crossing the event horizon of the BH. Therefore, the star which crosses the event horizon of the BH will be absorbed in the BH. When a star is sucked by the BH, it is died out and another star is born, i.e. a new solution is randomly generated in the search space.

The Euclidean distance R_E between the star and BH is calculated and compared with the radius of the event horizon. The radius of the event horizon of BH is calculated using Eq. (23):

$$R = \frac{f_{BH}}{\sum_{i=1}^{N} f_i}, \tag{23}$$

where f_{BH} represents the fitness value of the black hole (current best solution) and f_i is the fitness value of the ith star. N represents the total number of solutions in the search space.

The DBHABC algorithm is divided into four phases, namely employed bee phase, onlooker bee phase, scout bee phase and disruption phase. The BH phenomenon is applied in the employed bee phase of the algorithm, and a new phase namely disruption phase is applied after scout bee phase. The other phases are kept same as in the basic ABC. Based on the above explanation, the DBHABC algorithm is described in Algorithm 1.

Algorithm 1 : Disruption black hole artificial bee colony (DBHABC)

Initialize the parameters: MCN (Maximum number of cycles), D (Dimension of the problem), SN (Swarm Size), C_0, ρ;
Initialize the swarm having solutions, x_i where (i = 1, 2, ..., SN) by using Eq. (21);
cycle = 1;
while cycle $<>$ MCN **do**
  Employed Bee Phase: /* Explained as follows:*/
  for each solution **do**
    Evaluate the objective function;
    Select the best fitness solution as Black hole;
    Change the position of each star solution using Eq. (22); /* $fitness_{star}$ represents the fitness of the star solution and $fitness_{BH}$ represents the fitness of the black hole solution */
    if $fitness_{star} > fitness_{BH}$ **then**
      Interchange the position of star and black hole; /* R_{BH} represents the radius of the event horizon of black hole while R_E represents the Euclidean distance between the star solution and the black hole solution */
    end if
    if $R_{BH} > R_E$ **then**
      Generate a new solution x_i randomly in the search space;
    end if
  end for
  Onlooker Bee Phase;
  Scout Bee Phase;
  Step 4: Memorize the best food source found so far (Considered as star Solution);
  Step 5: Disruption Phase: /* Explained as follows:*/
  for each solution **do**
    Check the condition using Eq. (24) for all the candidate solutions except the star solution; here $R_{i,j}$ represents the Euclidean distance between the ith solution and its neighbour j, while $R_{i,best}$ is the Euclidean distance between the ith solution and the star solution; C is a threshold calculated using the Eq. (25);
    if ($\frac{R_{i,j}}{R_{i,best}} < C$) **then**
      D is calculated using the Eq. (27);
      Change the position of the solutions using Eq. (26);
    end if
  end for
  Memorize the best food source found so far;
  cycle=cycle+1;
end while
Output the best solution found so far.

The disruption phase is described as follows: for all the candidate solutions except the star solution (having best fitness value) following disruption condition is checked.

$$\frac{R_{i,j}}{R_{i,best}} < C^c, \tag{24}$$

where, $R_{i,j}$ and $R_{i,best}$ are the Euclidean distances between the ith and jth candidate solution and between ith and the best solution, respectively. Here j is the nearest neighbour of i. C^c is a threshold and it is defined as:

$$C^c = C_0 \left(1 - \frac{t}{T}\right) \tag{25}$$

Here, C_0 is a constant as described in Section 4.1. T and t represents the total number of iterations and current iteration, respectively. The solutions that satisfy the Eq. (24) are disrupted under the vicinity of the star (best) solution. The threshold C^c is a variable which is used to make the operator more meaningful. Initially when the solutions are not converged the value of C^c is kept to be large that leads to increase the exploration of the search space and as the solutions get closer to each other, C^c has to be small for exploitation of the search space. The position update equation for the candidate solutions that satisfies the Eq. (24) is defined as:

$$x_i(t+1) = x_i(t) \times D^d. \tag{26}$$

Here, $x_i(t)$ and $x_i(t+1)$ are the position of the ith candidate solution during the iteration t and $t+1$, respectively.

The value of D^d is defined as:

$$D^d = \begin{cases} R_{i,j} \times \text{U}(-0.5{,}0.5), & \text{if } R_{i,best} >= 1 \\ (1 + \rho \times \text{U}(-0.5{,}0.5), & \text{otherwise.} \end{cases} \tag{27}$$

In the above Eq. (27), U(-0.5,0.5) is an uniformly distributed random number from the interval [-0.5,0.5]. ρ is a small number used for the purpose of exploitation of the search space. Depending upon the value of D^d, the exploration and exploitation of search space are performed during the phase. When the value of $R_{i,best} >= 1$ means the solutions are not converged, then the value of D^d can be less than or greater than 1 [using Eq. (27)] and by multiplying this value with previous value of the ith solution, the dimension of the ith solution is changed randomly and it can be smaller or larger than the previous solution value. So this leads to explore the search space. On the other part, when $R_{i,best} <= 1$ means ith solution is close to the best solution, then the value of D^d is set to very small, and it will update the position of the candidate solution near to old position. This helps in exploitation of the search space. This implies that when the solutions are not converged, the disruption operator explores and as the solutions are converging and getting close to the star solution, the operator exploits the search space. C^c and D^d are the two new control parameters in the disruption phase. Fine tuning of both the parameters are required for proper implementation of the strategy.

4 Comparison and analysis of result

The performance of proposed algorithm DBHABC is evaluated on 20 different benchmark continuous optimisation functions (f_1 to f_{20}) having different degrees of complexity and multimodality as shown in Table 1. To check the competitiveness of DBHABC, it is compared with, ABC algorithm (Karaboga, 2005b), PSO-2011 (Clerc and Kennedy, 2011), differential evolution (DE) algorithms (Price, 1996) and three significant variants of ABC algorithm namely, Gbest-guided artificial bee colony (GABC) algorithm (Zhu and Kwong, 2010), Memetic artificial bee colony (MeABC) algorithm (Bansal et al., 2013) and Lévy flight artificial bee colony (LFABC) algorithm (Sharma et al., 2015). The experimental setting is given in Section 4.1.

4.1 Experimental setting

To prove the efficiency of proposed DBHABC algorithm, following experimental setting is adopted:

- The number of simulations/run =100
- Colony size NP = 50 and Number of food sources SN = NP/2
- $C_0 = 60$
- $\rho = 10^{-10}$
- $\phi_{ij} = rand[-1, 1]$ and limit = Dimension × Number of food sources = $D \times SN$ (Akay and Karaboga, 2012)
- Parameter setting for other considered algorithms are similar to their legitimate research papers (Banharnsakun, Achalakul and Sirinaovakul, 2011; Bansal et al., 2013; Kennedy and Eberhart, 1995; Storn and Price, 1995; Sharma et al., 2015; Zhu and Kwong, 2010)
- The data set for induction motors are used from Table 3.

4.2 Results comparison

The obtained results are shown in Table 2 expressed by success rate (SR), average number of function evaluations (AFE), mean error (ME) and standard deviation (SD).

The DBHABC is compared with ABC and its significant variants, it is also compared with DE and PSO. The results are shown in Table 2. The results reveal that DBHABC is a competitive algorithm and performs better for most of the optimisation benchmark functions irrespective of their nature. The boxplot analysis have also been carried out for all considered algorithms for comparison in terms of consolidated performance. In boxplot analysis tool (Williamson, Parker and Kendrick, 1989), graphical distribution of empirical data is efficiently represented. The boxplots for DBHABC and other considered algorithms are represented in Figure 1. It is clear from this figure that DBHABC performs better than the considered algorithms as interquartile range and median are quite low.

Table 1 Test problems. D, dimensions; C_h, characteristic; U, unimodal; M, multimodal; S, separable; N, non-separable; AE, acceptable error

Test problem	*Objective function*	*search Range*	*Optimum Value*	*D*	*AE*	C_h
Sphere	$f_1(x) = \sum_{i=1}^{D} x_i^2$	[−5.12, 5.12]	$f(\vec{0}) = 0$	30	$1.0E-05$	S, U
De Jong f4	$f_2(x) = \sum_{i=1}^{D} i \times (x_i)^4$	[−5.12, 5.12]	$f(\vec{0}) = 0$	30	$1.0E-05$	S, M
Cigar	$f_3(x) = {x_0}^2 + 100000 \sum_{i=1}^{D} {x_i}^2$	[−10, 10]	$f(\vec{0}) = 4$	30	$1.0E-05$	S, U
Brown3	$f_4(x) = \sum_{i=1}^{D-1} ({x_i}^{2(x_{i+1})^2+1} + {x_{i+1}}^{2{x_i}^2+1})$	[−1, 4]	$f(\vec{0}) = 0$	30	$1.0E-05$	U, N
Schewel	$f_5(x) = \sum_{i=1}^{D} \lvert x_i \rvert + \prod_{i=1}^{D} \lvert x_i \rvert$	[−10, 10]	$f(\vec{0}) = 0$	30	$1.0E-05$	N, U
Axis parallel hyper-ellipsoid	$f_6(x) = \sum_{i=1}^{D} i \times x_i^2$	[−5.12, 5.12]	$f(\vec{0}) = 0$	30	$1.0E-05$	U, S
Sum of different powers	$f_7(x) = \sum_{i=1}^{D} \lvert x_i \rvert^{i+1}$	[−1, 1]	$f(\vec{0}) = 0$	30	$1.0E-05$	S, M
Neumaier 3 Problem (NF3)	$f_8(x) = \sum_{i=1}^{D} (x_i - 1)^2 - \sum_{i=2}^{D} x_i x_{i-1}$	$[-D^2,\ D^2]$	$f_{min} = -\frac{(D(D+4)(D-1))}{6}$	10	$1.0E-01$	U,N
Rotated hyper-ellipsoid	$f_9(x) = \sum_{i=1}^{D} \sum_{j=1}^{i} x_j^2$	[−65.536, 65.536]	$f(\vec{0}) = 0$	30	$1.0E-05$	S, M
Ellipsoidal	$f_{10}(x) = \sum_{i=1}^{D} (x_i - i)^2$	[−D, D]	$f(1, 2, 3, \ldots, D) = 0$	30	$1.0E-05$	U, S
Beale function	$f_{11}(x) = [1.5 - x_1(1 - x_2)]^2 + [2.25 - x_1(1 - x_2^2)]^2 + [2.625 - x_1(1 - x_2^3)]^2$	[−4.5, 4.5]	$f(3, 0.5) = 0$	2	$1.0E-05$	N, M
Colville function	$f_{12}(x) = 100[x_2 - x_1^2]^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1)(x_4 - 1)$	[−10, 10]	$f(\vec{1}) = 0$	4	$1.0E-05$	N, M

Table 1 Test problems. D, dimensions; C_h, characteristic; U, unimodal; M, multimodal; S, separable; N, non-separable; AE, acceptable error (continued)

Test problem	*Objective function*	*search Range*	*Optimum Value*	*D*	*AE*	C_h
Kowalik	$f_{13}(x) = \sum_{i=1}^{11}[a_i - \frac{x_1(b_i^2+b_ix_2)}{b_i^2+b_ix_3+x_4}]^2$	[−5, 5]	$f(0.192833, 0.190836, 0.123117, 0.135766) = 0.000307486$	4	$1.0E-05$	M, N
Shifted Sphere	$f_{14}(x) = \sum_{i=1}^{D} z_i^2 + f_{bias}$, $z = x - o$, $x = [x_1, x_2, \ldots x_D]$, $o = [o_1, o_2, \ldots o_D]$	[−100, 100]	$f(o) = f_{bias} = -450$	10	$1.0E-05$	S, M
Shifted Ackley	$f_{15}(x) = -20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} z_i^2}) - \exp(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi z_i)) + 20 + e + f_{bias}$, $z = (x - o)$, $x = (x_1, x_2, \ldots x_D)$, $o = (o_1, o_2, \ldots o_D)$	[−32, 32]	$f(o) = f_{bias} = -140$	10	$1.0E-05$	S, M
Easom's function	$f_{16}(x) = -\cos x_1 \cos x_2 e^{((-(x_1-\pi)^2-(x_2-\pi)^2))}$	[−10, 10]	$f(\pi, \pi) = -1$	2	$1.0E-13$	S, M
Dekkers and Aarts	$f_{17}(x) = 10^5x_1^2 + x_2^2 - (x_1^2 + x_2^2)^2 + 10^{-5}(x_1^2 + x_2^2)^4$	[−20, 20]	$f(0, 15) = f(0, -15) = -24777$	2	$5.0E-01$	N, M
Meyer and Roth Problem	$f_{18}(x) = \sum_{i=1}^{5}(\frac{x_1x_3t_i}{1+x_1t_i+x_2v_i} - y_i)^2$	[−10, 10]	$f(3.13, 15.16, 0.78) = 0.4E-04$	3	$1.0E-03$	U, N
Shubert	$f_{19}(x) = -\sum_{i=1}^{5} i\cos((i+1)x_1 + 1)\sum_{i=1}^{5} i\cos((i+1)x_2 + 1)$	[−10, 10]	$f(7.0835, 4.8580) = -186.7309$	2	$1.0E-05$	S, M
Moved axis parallel hyper-ellipsoid	$f_{20}(x) = \sum_{i=1}^{D} 5i \times x_i^2$	[−5.12, 5.12]	$f(x) = 0; x(i) = 5 \times i, i = 1 : D$	30	$1.0E-15$	U, S

Table 2 Comparison of the results of test problems

Test Function	*Measure*	*DBHABC*	*PSO*	*LFABC*	*MeABC*	*GABC*	*ABC*	*DE*
f_1	SD	**8.12E-07**	1.73E-06	7.87E-07	1.81E-06	1.98E-06	2.15E-06	2.02E-06
	ME	**7.88E-06**	8.39E-06	9.27E-06	8.11E-06	7.33E-07	7.59E-06	8.17E-06
	AFE	**10205.35**	16733.85	13626.8	14347.5	29341.69	30062	20409
	SR	100	100	100	100	100	100	100
f_2	SD	**1.36E-06**	3.02E-06	1.69E-06	2.72E-06	1.69E-06	3.12E-06	3.11E-06
	ME	8.04E-06	6.62E-06	8.26E-06	5.51E-06	4.59E-07	5.37E-06	4.90E-06
	AFE	8285.64	9556.12	5516.18	8388	24055.41	24523.5	9578.5
	SR	100	100	100	100	100	100	100
f_3	SD	**1.14E-06**	1.65E-06	8.48E-07	1.96E-06	2.15E-06	2.28E-06	2.20E-06
	ME	8.66E-06	8.84E-06	9.15E-06	8.10E-06	7.99E-06	7.58E-06	7.77E-06
	AFE	**19674.68**	24546.79	28443.32	23163	62825.35	62171.5	34887
	SR	100	100	100	100	100	100	100
f_4	SD	8.97E-07	1.58E-06	7.70E-07	1.88E-06	2.17E-06	2.15E-06	2.13E-06
	ME	8.51E-06	8.55E-06	9.21E-06	7.91E-06	7.50E-06	7.49E-06	7.84E-06
	AFE	**9982.85**	16111.3	13848.76	14072	28208.78	31418.5	20917
	SR	100	100	100	100	100	100	100
f_5	SD	7.70E-07	8.04E-07	3.75E-07	6.90E-07	8.71E-07	1.10E-06	1.03E-06
	ME	9.12E-06	9.34E-06	9.58E-06	9.30E-06	9.48E-06	9.19E-06	9.16E-06
	AFE	**21135.18**	30994.7	31451.16	27636	62256.74	52967	41646
	SR	100	100	100	100	100	100	100
f_6	SD	7.10E-07	1.70E-06	7.12E-07	1.98E-06	1.19E-06	2.18E-06	2.04E-06
	ME	7.92E-06	8.43E-06	9.20E-06	7.71E-06	9.90E-06	7.70E-06	7.96E-06
	AFE	**10691.31**	18093.08	16789.28	16084.5	52119.75	36751	22672
	SR	100	100	100	100	100	100	100
f_7	SD	2.86E-06	3.13E-06	2.87E-06	2.95E-06	2.95E-06	2.96E-06	2.65E-06
	ME	6.36E-06	5.86E-06	5.59E-06	5.62E-06	5.35E-06	6.19E-06	5.16E-06
	AFE	**3723.85**	7523.66	4527.08	9285	46248.07	14209	16229
	SR	100	100	100	100	100	100	100
f_8	SD	**8.89E-07**	6.84E-02	1.06E-02	1.04E+00	1.00E+00	6.78E+00	6.74E-01
	ME	**5.00E-06**	1.07E-01	8.88E-02	1.17E+00	7.56E-01	4.64E+00	9.22E-01
	AFE	**22110.61**	39650.81	22134.93	200014.61	201186.56	200024.84	200004.61
	SR	**100**	95	100	0	0	0	0
f_9	SD	**1.01E-06**	1.99E-06	9.59E-07	1.93E-06	2.98E-06	2.42E-06	2.14E-06
	ME	8.58E-06	8.51E-06	9.11E-06	7.85E-06	7.98E-06	7.77E-06	7.80E-06
	AFE	**16102.15**	21192.18	22617.84	19475.5	58781.89	49131	28065
	SR	100	100	100	100	100	100	100
f_{10}	SD	1.01E-06	1.97E-06	8.51E-07	1.94E-06	1.65E-06	2.47E-06	2.34E-06
	ME	8.10E-06	8.07E-06	9.14E-06	7.82E-06	8.53E-06	8.17E-06	7.44E-06
	AFE	**11892.78**	18653.59	17885.74	16683.5	64658.99	41449	24167
	SR	100	100	100	100	100	100	100
f_{11}	SD	2.83E-06	2.84E-06	2.96E-06	3.04E-06	3.31E-06	6.45E-05	2.25E-06
	ME	**4.03E-06**	7.52E-06	4.94E-06	5.61E-06	5.38E-06	6.26E-05	8.23E-06
	AFE	1758.93	3746.11	2573.53	9335.88	16158.98	49489.15	16098.58
	SR	100	100	100	100	100	93	100
f_{12}	SD	**3.53E-04**	1.29E-03	2.31E-03	1.61E-02	3.16E-02	3.68E-02	1.20E-01
	ME	**6.92E-04**	9.19E-03	6.99E-03	1.58E-02	2.53E-02	2.73E-02	1.74E-01
	AFE	**19125.84**	65107.64	29780.95	197731.18	199159.1	198577.54	200023.42
	SR	100	100	100	5	2	2	0

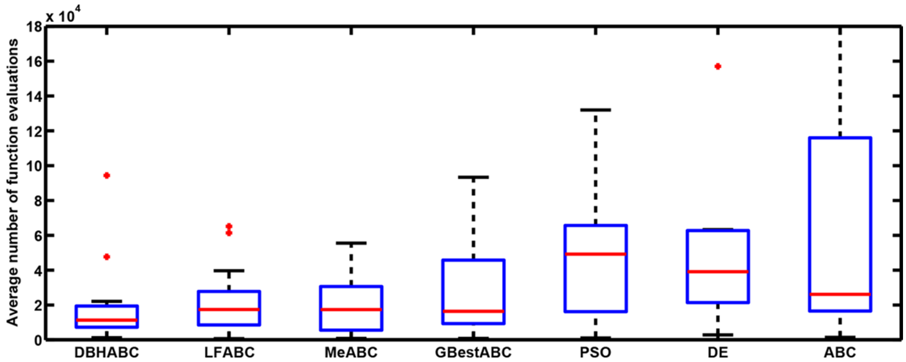
Table 2 Comparison of the results of test problems (continued)

Test Function	*Measure*	*DBHABC*	*PSO-2011*	*LFABC*	*MeABC*	*GABC*	*ABC*	*DE*
f_{13}	SD	8.35E-05	1.79E-04	1.63E-05	3.80E-05	6.95E-05	6.55E-05	7.97E-05
	ME	9.41E-05	1.37E-04	8.46E-05	9.10E-05	1.81E-04	1.39E-04	1.75E-04
	AFE	**47661.35**	61386.26	41583.08	93336.22	131931.67	156968.02	181667.37
	SR	97	95	100	90	67	49	20
f_{14}	SD	2.62E-06	2.36E-06	1.89E-06	1.94E-06	2.24E-06	2.43E-06	2.50E-06
	ME	7.00E-06	7.27E-06	7.64E-06	7.64E-06	7.14E-06	6.97E-06	6.85E-06
	AFE	6202.14	6203.32	5587.6	5546.5	16187.82	18225	9074.5
	SR	100	100	100	100	100	100	100
f_{15}	SD	**9.41E-07**	1.34E-06	1.49E-06	1.60E-06	1.57E-06	1.58E-06	1.97E-06
	ME	**7.48E-06**	8.66E-06	8.71E-06	8.28E-06	8.73E-06	8.83E-06	7.79E-06
	AFE	**8810.98**	10934.63	10002.68	9305.5	11656.6	32219	16842
	SR	100	100	100	100	100	100	100
f_{16}	SD	2.76E-14	3.28E-14	2.08E-11	1.50E-12	1.56E-11	2.97E-14	8.06E-05
	ME	4.58E-14	5.60E-14	2.15E-12	2.03E-13	1.65E-12	4.24E-14	2.71E-05
	AFE	14092.61	14065.55	55466.86	43142.05	66599.57	4666.62	181234.08
	SR	**100**	100	98	99	99	100	17
f_{17}	SD	5.25E-03	5.68E-03	5.37E-03	4.84E-03	5.22E-03	5.51E-03	5.41E-03
	ME	4.89E-01	4.91E-01	4.89E-01	4.89E-01	4.90E-01	4.92E-01	4.89E-01
	AFE	1203.94	687.8	783.58	785	1069.99	2824.11	1432.53
	SR	100	100	100	100	100	100	100
f_{18}	SD	**2.65E-06**	3.10E-06	2.87E-06	2.91E-06	2.96E-06	2.77E-06	2.85E-06
	ME	1.95E-03	1.95E-03	1.95E-03	1.95E-03	1.97E-03	1.94E-03	1.95E-03
	AFE	**1337.96**	3418.07	3886.93	4809.51	8872.9	16676.11	31742.53
	SR	100	100	100	100	100	100	100
f_{19}	SD	**9.41E-04**	1.67E-03	2.39E-03	2.36E-03	1.87E-03	2.19E-03	1.85E-03
	ME	**1.30E-04**	8.35E-03	7.30E-03	7.50E-03	6.57E-03	7.43E-03	7.88E-03
	AFE	17291.91	22030.31	37251.9	48341.78	5139.55	63281.46	50666.58
	SR	**100**	100	99	99	100	100	100
f_{20}	SD	**9.23E-03**	1.62E+00	1.65E+00	3.88E+00	5.21E+00	2.07E+01	1.01E+01
	ME	**5.73E-04**	1.03E+00	2.20E+00	6.12E+00	5.94E+00	2.66E+01	1.74E+01
	AFE	**94344.7**	199313.43	200053.04	200025.49	201228.24	200033.4	200023.08
	SR	**71**	3	0	0	0	0	0

Table 3 Data of the motors used

Specifications	*Motor 1*	*Motor 2*
Capacity (HP)	5	40
Voltage (V)	3000	3000
Frequency (Hz)	50	50
No. of poles	4	4
Full-load slip	0.07	0.07
Starting torque (Nm)	15	260
Maximum torque (Nm)	42	370
Full load torque (Nm)	25	190

Figure 1 Boxplots graphs for average number of function evaluation



4.3 Induction motor parameter estimation

In this work, penalty technique is used for converting multiobjective induction motor parameter estimation problem into single-objective parameter estimation problem. The estimated value of torque is obtained, and it is compared with manufacturer's value. The comparison between manufacturer's value and estimated parameter values for proposed and other considered algorithms is presented in Tables 4, 5, 6 and 7. The presented results reveal that DBHABC is a better choice for induction motor parameter estimation.

Table 4 Estimated parameters and percentage error using approximate model for motor 1

Algorithm	T_{MAX}(Nm)	T_{STR} (Nm)	T_{FL}(Nm)	T_{MAX}	T_{STR}	T_{FL}	R_1	R_2	X_1
Manufacture	42	15	25	(%error)	(%error)	(%error)	(ohms)	(ohms)	(ohms)
ABC (Abro and Saleh, 2014)	39.819	15.074	25.166	5.192	0.459	−0.663	0.001	7.702	35.971
GABC 1.5 (Abro and Saleh, 2014)	39.745	15.078	25.075	5.37	0.518	0.229	0.06	7.178	36.596
IABC 0.35 (Abro and Saleh, 2014)	39.72	15.059	25.072	5.428	0.396	0.286	0.001	7.101	36.06
MABC (Abro and Saleh, 2014)	39.735	15.04	25.116	5.393	0.263	0.466	0.001	7.085	36.047
ModABC (Abro and Saleh, 2014)	39.691	15.102	24.978	5.498	0.682	0.088	0.001	7.134	36.087
BSFABC (Abro and Saleh, 2014)	39.789	15.165	25.005	5.265	1.097	0.019	0.001	7.129	35.998
EABC (Abro and Saleh, 2014)	39.634	15.046	24.99	5.634	0.308	0.04	0.001	7.126	36.139
CH-EABC (Abro and Saleh, 2014)	39.682	15.034	25.063	5.519	0.226	0.252	0.001	7.102	36.095
DBHABC	**39.828**	**15.031**	25.0047	**5.17142**	0.2067	0.188	0.001	**7.082**	36.082

Table 5 Estimated parameters and percentage error using exact model for motor 1

Algorithm	T_{MAX}(Nm)	T_{STR} (Nm)	T_{FL}(Nm)	T_{MAX}	T_{STR}	T_{FL}	R_1	R_2	X_1	X_2	X_m
Manufacture	42	15	25	%error	%error	%error	ohms	ohms	ohms	ohms	ohms
ABC (Abro and Saleh, 2014)	39.683	15.068	25.017	5.518	0.45	0.067	0.001	6.37	20	13.38	350
GABC 0.5 (Abro and Saleh, 2014)	39.716	15.073	25.048	5.437	0.487	0.193	0.001	6.48	16.72	16.89	350
IABC 0.25 (Abro and Saleh, 2014)	39.698	15.046	25.065	5.48	0.307	0.261	0.001	6.26	20	13.01	304.93
MABC (Abro and Saleh, 2014)	39.729	14.992	25.176	5.408	0.055	0.706	0.001	6.36	18.78	14.65	350
ModABC (Abro and Saleh, 2014)	39.7	14.917	25.248	5.477	0.555	0.992	0.001	6.35	18.27	15.22	350
BSFABC (Abro and Saleh, 2014)	39.795	15.056	25.163	5.251	0.374	0.651	0.001	6.36	18.95	14.41	349.07
EABC (Abro and Saleh, 2014)	39.706	15.013	25.12	5.461	0.089	0.48	0.001	6.34	18.35	14.93	322.54
CH-EABC (Abro and Saleh, 2014)	39.616	14.988	25.052	5.675	0.083	0.207	0.001	5.93	19.8	12.09	209.05
DBHABC	**39.811**	15.011	25.042	**5.212**	0.0733	0.168	0.001	**5.91**	**16.66**	**12.01**	**208.72**

Table 6 Estimated parameters and percentage error using approximate model for motor 2

Algorithm	T_{MAX}(Nm)	T_{STR}(Nm)	T_{FL}(Nm)	T_{MAX}	T_{STR}	T_{FL}	R_1	R_2	X_1
Manufacturer	370	260	190	%error	%error	%error	ohms	ohms	ohms
ABC (Abro and Saleh, 2014)	360.576	260.656	194.774	2.547	−0.252	−2.513	1.693	0.759	1.526
GABC 1.5 (Abro and Saleh, 2014)	369.302	260.765	189.371	0.189	−0.294	0.331	1.424	0.824	1.999
IABC 0.35 (Abro and Saleh, 2014)	361.081	260.793	189.477	2.411	−0.305	0.275	1.606	0.799	1.731
MABC (Abro and Saleh, 2014)	374.994	262.557	193.542	−1.350	−0.983	−1.864	1.391	0.805	1.991
ModABC (Abro and Saleh, 2014)	365.1242	251.5106	193.1943	1.3178	3.263	−1.681	1.443	0.7982	2.017
BSFABC (Abro and Saleh, 2014)	343.0155	264.3033	181.6591	7.2931	1.655	4.39	1.968	0.7984	1
EABC (Abro and Saleh, 2014)	369.85	259.483	189.927	0.041	0.199	0.039	1.403	0.823	2.033
CH-EABC (Abro and Saleh, 2014)	368.945	261.01	190.272	0.285	−0.389	−1.864	1.449	0.816	1.956
DBHABC	**369.857**	260.556	189.953	**0.0386**	0.2138	0.0247	**1.382**	**0.751**	1.000

Table 7 Estimated parameters and percentage error using exact model for motor 2

Algorithm	T_{MAX}(Nm)	T_{STR}(Nm)	T_{FL}(Nm)	T_{MAX}	T_{STR}	T_{FL}	R_1	R_2	X_1	X_2	X_m
Manufacturer	370	260	190	%error	%error	%error	ohms	ohms	ohms	ohms	ohms
ABC (Abro and Saleh, 2014)	353.752	261.725	185.846	4.391	−0.664	2.186	1.732	0.803	0.1	1.44	400
GABC 2.5 (Abro and Saleh, 2014)	372.785	259.467	188.002	−0.753	0.205	1.052	1.311	0.846	0.1	2.06	400
IABC 0.15 (Abro and Saleh, 2014)	368.416	263.277	192.661	0.428	−1.26	−1.401	1.526	0.792	0.1	1.7	315
MABC (Abro and Saleh, 2014)	367.383	262.135	194.746	0.707	−0.821	−2.498	1.563	0.776	0.17	1.56	400
ModABC (Abro and Saleh, 2014)	373.518	259.82	184.503	−0.951	0.069	2.893	1.241	0.867	0.88	1.37	200
BSFABC (Abro and Saleh, 2014)	358.528	266.194	174.979	3.101	−2.382	7.906	1.508	0.892	1.84	0.12	400
EABC (Abro and Saleh, 2014)	370.075	260.013	189.752	−0.02	−0.005	0.131	1.403	0.824	0.1	1.93	400
CH-EABC (Abro and Saleh, 2014)	369.601	260.091	190.734	0.108	−0.035	−0.386	1.429	0.815	0.1	1.89	400
DBHABC	**369.081**	259.991	190.734	0.248	**0.0034**	0.374	1.217	**0.773**	**0.10**	**0.10**	400

5 Conclusion

In this paper, an efficient variant of ABC algorithm, namely DBHABC is proposed. The proposed variant is based on amalgamation of two significant phenomena of physics. One is black hole and another is disruption. The DBHABC enhances exploration and exploitation as well as maintains optimum balance between these two. Further, the proposed variant is applied for estimation of induction motor parameters. On comparing with the other existing methods, it is found that DBHABC is a better choice for induction motor parameter estimation optimisation problem.

References

Abro, A.G. and Saleh, J.M. (2014) 'Multiple-global-best guided artificial bee colony algorithm for induction motor parameter estimation', *Turkish Journal of Electrical Engineering and Computer Sciences*, Vol. 22, No. 3, pp.620–636.

Abu-Mouti, F.S. and El-Hawary, M.E. (2012) 'Overview of artificial bee colony (ABC) algorithm and its applications', *Systems Conference (SysCon), 2012 IEEE International*, Canada, pp.1–6.

Akay, B. and Karaboga, D. (2012) 'A modified artificial bee colony algorithm for real-parameter optimization', *Information Sciences*, Vol. 192, pp.120–142.

Banharnsakun, A., Achalakul, T. and Sirinaovakul, B. (2011) 'The best-so-far selection in artificial bee colony algorithm', *Applied Soft Computing*, Vol. 11, No. 2, pp.2888–2901.

Bansal, J.C., Sharma, H., Arya, K. and Nagar, A. (2013) 'Memetic search in artificial bee colony algorithm', *Soft Computing*, Vol. 17, No. 10, pp.1911–1928.

Bansal, J.C., Sharma, H. and Jadon, S.S. (2013) 'Artificial bee colony algorithm: a survey', *International Journal of Advanced Intelligence Paradigms*, Vol. 5, No. 1, pp.123–159.

Bansal, J.C., Sharma, H., Nagar, A. and Arya, K. (2013) 'Balanced artificial bee colony algorithm', *International Journal of Artificial Intelligence and Soft Computing*, Vol. 3, No. 3, pp.222–243.

Bazmi, A.A. and Zahedi, G. (2011) 'Sustainable energy systems: role of optimization modeling techniques in power generation and supply - a review', *Renewable and Sustainable Energy Reviews*, Vol. 15, No. 8, pp.3480–3500.

Chandrasekaran, K., Hemamalini, S., Simon, S.P. and Padhy, N.P. (2012) 'Thermal unit commitment using binary/real coded artificial bee colony algorithm', *Electric Power Systems Research*, Vol. 84, No. 1, pp.109–119.

Clerc, M. and Kennedy, J. (2011) *Standard pso 2011*, Particle Swarm Central Site [online], 1 January, 2016, http://www.particleswarm.info.

Dandil, E. *et al.* (2013) 'Artificial immunity-based induction motor bearing fault diagnosis', *Turkish Journal of Electrical Engineering and Computer Science*, Vol. 21, No. 1, pp.1–25.

Ding, G.Y., Liu, H. and He, X.Q. (2013) 'A novel disruption operator in particle swarm optimization', *Applied Mechanics and Materials*, Vol. 380, pp.1216–1220.

Doraghinejad, M., Nezamabadi-pour, H., Hashempour Sadeghian, A. and Maghfoori, M. (2012) 'A hybrid algorithm based on gravitational search algorithm for unimodal optimization', *2012 2nd International eConference on Computer and Knowledge Engineering (ICCKE)*, Iran, pp.129–132.

Eberhart, R.C., Kennedy, J. *et al.* (1995) 'A new optimizer using particle swarm theory', *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, Vol. 1, Nagoya, Japan, pp.39–43.

El-Abd, M. (2012) 'Performance assessment of foraging algorithms vs. evolutionary algorithms', *Information Sciences*, Vol. 182, No. 1, pp.243–263.

Eskandar, H., Sadollah, A., Bahreininejad, A. and Hamdi, M. (2012) 'Water cycle algorithm - a novel metaheuristic optimization method for solving constrained engineering optimization problems', *Computers & Structures*, Vol. 110, pp.151–166.

Gupta, R., Wadhwani, A. and Kapoor, S. (2011) 'Early estimation of faults in induction motors using symbolic dynamic-based analysis of stator current samples', *IEEE Transactions on Energy Conversion*, Vol. 26, No. 1, pp.102–114.

Gozde, H., Taplamacioglu, M.C. and Kocaarslan, I. (2012) 'Comparative performance analysis of artificial bee colony algorithm in automatic generation control for interconnected reheat thermal power system', *International Journal of Electrical Power & Energy Systems*, Vol. 42, No. 1, pp.167–178.

Hatamlou, A. (2013) 'Black hole: a new heuristic optimization approach for data clustering', *Information Sciences*, Vol. 222, pp.175–184.

Heidari, A.A., Abbaspour, R.A. and Jordehi, A.R. (2015) 'An efficient chaotic water cycle algorithm for optimization tasks', *Neural Computing and Applications*, Springer, London, pp.1–29.

Jordehi, A.R. (2015a) 'Chaotic bat swarm optimisation (CBSO)', *Applied Soft Computing*, Vol. 26, pp.523–530.

Jordehi, A.R. (2015b) 'Enhanced leader PSO (ELPSO): a new PSO variant for solving global optimisation problems', *Applied Soft Computing*, Vol. 26, pp.401–417.

Karaboga, D. (2005a) *An Idea Based on Honey Bee Swarm for Numerical Optimization*, Technical Report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department.

Karaboga, D. (2005b) *An Idea Based on Honey Bee Swarm for Numerical Optimization*, Technical Report-TR06, Erciyes University Press, Erciyes.

Karaboga, D. and Akay, B. (2009) 'A comparative study of artificial bee colony algorithm', *Applied Mathematics and Computation*, Vol. 214, No. 1, pp.108–132.

Karaboga, D. and Basturk, B. (2007) 'A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm', *Journal of Global Optimization*, Vol. 39, No. 3, pp.459–471.

Karaboga, D. and Basturk, B. (2008) 'On the performance of artificial bee colony (ABC) algorithm', *Applied Soft Computing*, Vol. 8, No. 1, pp.687–697.

Kennedy, J. and Eberhart, R. (1995) 'Particle swarm optimization', *IEEE International Conference on Neural Networks, 1995. Proceedings*, Vol. 4, Nagoya, Japan, pp.1942–1948.

Lindenmeyer, D., Dommel, H., Moshref, A. and Kundur, P. (2001) 'An induction motor parameter estimation method', *International Journal of Electrical Power & Energy Systems*, Vol. 23, No. 4, pp.251–262.

Liu, H., Ding, G. and Sun, H. (2012) 'An improved opposition-based disruption operator in gravitational search algorithm', *2012 Fifth International Symposium on Computational Intelligence and Design (ISCID)*, Vol. 2, China, pp.123–126.

Montgomery, C., Orchiston, W. and Whittingham, I. (2009) 'Michell, laplace and the origin of the black hole concept', *Journal of Astronomical History and Heritage*, Vol. 12, pp.90–96.

Nesmachnow, S. (2014a) 'An overview of metaheuristics: accurate and efficient methods for optimisation', *International Journal of Metaheuristics*, Vol. 3, No. 4, pp.320–347.

Nesmachnow, S. (2014b) 'Metaheuristics as soft computing techniques for efficient optimization', in Khosrow-Pour, M. (Ed.): *Encyclopedia of Information Science and Technology*, IGI Global, USA, pp.1–10.

Pedra, J. and Corcoles, F. (2004) 'Estimation of induction motor double-cage model parameters from manufacturer data', *IEEE Transactions on Energy Conversion*, Vol. 19, No. 2, pp.310–317.

Price, K. (1996) 'Differential evolution: a fast and simple numerical optimizer', *Fuzzy Information Processing Society, 1996. NAFIPS. 1996 Biennial Conference of the North American*, USA, pp.524–527.

Sakthivel, V., Bhuvaneswari, R. and Subramanian, S. (2011) 'An accurate and economical approach for induction motor field efficiency estimation using bacterial foraging algorithm', *Measurement*, Vol. 44, No. 4, pp.674–684.

Samanta, S. and Chakraborty, S. (2011) 'Parametric optimization of some non-traditional machining processes using artificial bee colony algorithm', *Engineering Applications of Artificial Intelligence*, Vol. 24, No. 6, pp.946–957.

Sarafrazi, S., Nezamabadi-Pour, H. and Saryazdi, S. (2011) 'Disruption: a new operator in gravitational search algorithm', *Scientia Iranica*, Vol. 18, No. 3, pp.539–548.

Sharma, H., Bansal, J.C. and Arya, K. (2014) 'Power law-based local search in artificial bee colony', *International Journal of Artificial Intelligence and Soft Computing*, Vol. 4, Nos. 2/3, pp.164–194.

Sharma, H., Bansal, J.C., Arya, K. and Yang, X-S. (2015) 'Lévy flight artificial bee colony algorithm', *International Journal of Systems Science*, Vol. 3, pp.1–19.

Sharma, N., Sharma, H., Sharma, A. and Bansal, J.C. (2015) 'Black hole artificial bee colony algorithm', *In International conference on swarm, evolutionary and memetic computing*, India, pp.214–221.

Sharma, N., Sharma, H., Sharma, A. and Bansal, J.C. (2016) 'Modified artificial bee colony algorithm based on disruption operator', *Proceedings of Fifth International Conference on Soft Computing for Problem Solving*, India, pp.889–900.

Storn, R. and Price, K. (1995) 'Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces', *International Computer Science Institute - Publications - TR*, ICSI Berkeley, Vol. 3.

Toliyat, H., Levi, E., Raina, M. *et al.* (2003) 'A review of RFO induction motor parameter estimation techniques', *IEEE Transactions on Energy conversion*, Vol. 18, No. 2, pp.271–283.

Williamson, D.F., Parker, R.A. and Kendrick, J.S. (1989) 'The box plot: a simple visual method to interpret data', *Annals of Internal Medicine*, Vol. 110, No. 11, pp.916–921.

Yang, X-S. (2010) 'A new metaheuristic bat-inspired algorithm', *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, Springer, Spain, pp.65–74.

Zhang, J., Liu, K., Tan, Y. and He, X. (2008) 'Random black hole particle swarm optimization and its application', *International Conference on Neural Networks and Signal Processing, 2008*, China, pp.359–365.

Zhu, G. and Kwong, S. (2010) 'Gbest-guided artificial bee colony algorithm for numerical function optimization', *Applied Mathematics and Computation*, Vol. 217, No. 7, pp.3166–3173.