Applied Soft Computing 31 (2015) 153–171

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Artificial algae algorithm (AAA) for nonlinear global optimization

Sait Ali Uymaz [a], Gulay Tezel [a,*], Esra Yel [b]

[a] Department of Computer Engineering, Selçuk University, Konya, Turkey
[b] Department of Environmental Engineering, Selçuk University, Konya, Turkey

ARTICLE INFO

Article history:
Received 21 October 2014
Received in revised form 27 February 2015
Accepted 1 March 2015
Available online 10 March 2015

Keywords:
Artificial algae algorithm
Bio-inspired algorithm
Metaheuristic
Optimization

ABSTRACT

In this study, a novel bio-inspired metaheuristic optimization algorithm called artificial algae algorithm (AAA) inspired by the living behaviors of microalgae, photosynthetic species, is introduced. The algorithm is based on evolutionary process, adaptation process and the movement of microalgae. The performance of the algorithm has been verified on various benchmark functions and a real-world design optimization problem. The CEC'05 function set was employed as benchmark functions and the test results were compared with the algorithms of Artificial Bee Colony (ABC), Bee Algorithm (BA), Differential Evolution (DE), Ant Colony Optimization for continuous domain (ACO_R) and Harmony Search (HS_{POP}). The pressure vessel design optimization problem, which is one of the widely used optimization problems, was used as a sample real-world design optimization problem to test the algorithm. In order to compare the results on the mentioned problem, the methods including ABC and Standard PSO (SPSO2011) were used. Mean, best, standard deviation values and convergence curves were employed for the analyses of performance. Furthermore, mean square error (MSE), root mean square error (RMSE) and mean absolute percentage error (MAPE), which are computed as a result of using the errors of algorithms on functions, were used for the general performance comparison. AAA produced successful and balanced results over different dimensions of the benchmark functions. It is a consistent algorithm having balanced search qualifications. Because of the contribution of adaptation and evolutionary process, semi-random selection employed while choosing the source of light in order to avoid local minima, and balancing of helical movement methods each other. Moreover, in tested real-world application AAA produced consistent results and it is a stable algorithm.

1. Introduction

Optimization is the operation of finding the optimal solution (or solutions) to a given problem. The aim is to maximize profits by keeping resources minimum. Choosing the optimal method is important for this purpose. The complexity of real-world problems makes search of all possible solutions or combinations impossible. As resources are limited in real-world applications, mathematical modeling approaches seek solutions by simplifying problems or limiting them by making assumptions. As a result, an acceptable real solution for the original problem may considerably vary from solutions obtained from the modified model [1].

Optimization problems can be formulated in many ways. So far, the most widely used formulations is to write a nonlinear optimization problem as

$$\text{Minimize or Maximize } f_i(x), (\mathrm{i} = 1, 2, \ldots, M) \quad (1)$$

Subject to the following nonlinear constraints:

$$h_j(x) = 0, (j = 1, 2, \ldots, J) \quad (2)$$

$$g_k(x) <= 0, (k = 1, 2, \ldots, K) \quad (3)$$

where f_i, h_j, and g_k are in general nonlinear functions, or even integrals and/or differential equations. Here, the design vector $x = (x_1, x_2, \ldots, x_n)$ can be continuous, discrete, or mixed in d-dimensional space. The functions f_i are called objectives, or cost functions, and when $M > 1$, the optimization is multiobjective or multicriteria [2].

Heuristic is to produce acceptable solution for solving the problem in reasonable time. This solution may not be optimal or it may simply approximate the exact solution. The heuristic methods can be used to reduce the inconsistency between solutions obtained from a mathematical model approach, and to solve realistic problems because these methods allow the search within the entire solution space and present sub-optimal solutions with high quality [1]. The complexity of the problem makes it impossible for the search of all possible combinations or solutions. The aim is to find

* Corresponding author. Tel.: +90 3322233710.
E-mail addresses: aliuymaz@selcuk.edu.tr (S.A. Uymaz), gtezel@selcuk.edu.tr, g_tezel@hotmail.com (G. Tezel), etarlan@selcuk.edu.tr (E. Yel).

http://dx.doi.org/10.1016/j.asoc.2015.03.003
1568-4946/

a good solution in an acceptable time. There is no guarantee to find the best solution. The aim is to have an effective and practical algorithm that can produce high quality solutions [3,4]. Similar to heuristic, metaheuristic is a general algorithmic framework that can be applied at diverse group of optimization problems by adapting the algorithm to a particular problem by several modifications. The difference is that, metaheuristic is designed by combining one or more procedures (heuristic methods) with a higher-level strategy (that is why it is 'meta'). Each procedure of the metaheuristic is a black-box, which can be replaced with another procedure. These procedures can either be as simple as the manipulation of a representation or complex as another metaheuristic [4].

Researchers have developed a number of high performance heuristic optimization methods by taking inspiration from biology, physics, neurology and other disciplines since the mid-20th century [5]. Recently, there has been a growing interest to solve complex optimization problems inspiring from biology. Algorithms having these approaches are called bio-inspired metaheuristic algorithms, which were modeled by finding useful simulations, and by understanding and mimicking the behaviors in biological systems [4,5]. Generally, many studies in this area have focused on simulation of natural phenomena rather than the construction of mathematical and engineering tools. Successful bio-inspired metaheuristic methods are inspired by either Darwin's evolution rules or intelligent behaviors of swarms [6].

Evolutionary computation approaches include genetic algorithm, differential evolution, and harmony search algorithms. The main principle of these approaches is keeping the good individuals alive and being transferred them to the next generations. The first study in evolutionary computation approaches is genetic algorithm developed by the Holand et al. [7]. Basically, genetic algorithm is based on the principle that only the fittest survive during the transition of a population from one generation to the next. It offers multiple solutions to problems instead of a single solution. Each candidate solution is an individual of the population. According to the fitness value obtained from objective function in the optimization problem, new individuals, which are transferred to the next generations, are modified (recombined and mutated) and then reproduced. The number of individuals producing good solutions in new generations increases, whereas the number of individuals producing bad solutions decreases. The individuals in the new generations are mutated at a particular level in order to provide diversity and to avoid local optimums. It is observed that the best solution can be found by iterating this evolutionary process.

Differential evolution algorithm developed by Storn and Price in 1997 is another population-based evolution algorithm which uses real-valued parameters [8]. It is a typical evolutionary algorithm and individuals in the population, vectors, are formed in the beginning, and then they are subjected to mutation, crossover and selection operations. In the mutation operation, the difference between two arbitrarily selected vectors of solution space was added to a selected population vector to obtain mutated vectors. In the crossover operation, parameters of new candidate vector are selected from either the mutated individual or from the non-mutated individuals (target vector) of the population. Selection operation is the decision about the new vector of the next generation population, either parent vector, or the new generation, which one gives the fittest value to the solution. Another example is harmony search algorithm developed by Geem et al. [9]. In this algorithm, the harmony of musicians in an orchestra is inspired. Although harmony search algorithm resembles genetic algorithm, they differ in forming new individuals, i.e., in crossover operation, all individuals are used in the former. The algorithm starts with the formation of harmony memory composed of randomly chosen vectors. In order to create the new vector, the harmony memory is either chosen from one of the vectors in the memory according to the consideration rate of harmony memory designated in the beginning or it is chosen randomly. Each chosen vector parameter undergoes a change at a particular level according to the pitch adjustment rate. If the new harmony vector produces better results than the worst harmony, the worst harmony vector is removed from the memory and the new harmony vector replaces.

The food search approaches use swarm intelligence. The general principle of these approaches is modelling collective food search behaviors of some creatures communicating with each other in nature. The first study on swarm intelligence was The Ant Colony Algorithm presented by Dorigo in [10]. In this algorithm, the swarm intelligence of ants communicating with each other using pheromones is inspired. Ants leave pheromone to help other ants to find their ways. The movement of artificial ants relies on this base. Each ant leaves pheromone on its way inversely proportional to the distance it passes. Other ants in the population find their ways by the help of these pheromones. In each iteration, pheromone quantities on paths are updated and local optimums are avoided by partial evaporation of pheromone. Another algorithm inspired by the social behaviors of a bird flock and fish school is particle swarm optimization developed by American social psychologist Kennedy and engineer Eberhart [11]. In this algorithm, each individual qualified as agent mimics the behaviors of the best individual in the swarm or its own best former behavior. One of the most effective studies in this field is the Artificial Bee Colony Algorithm developed by Karaboğa [12]. This algorithm is based on bees' nectar source finding activity in which they communicate with their special dances. While employee bees collect nectar in the neighborhood of the sources, scout bees are trying to explore new nectar sources. Another study on bees is Honey Bee Algorithm presented by Pham et al. [13]. Basically, honey bee algorithm also mimics food collecting behavior of honey bees. Another swarm intelligence-based approach is the firefly algorithm developed by Yang in 2007 and is inspired by the communication between fireflies through the signaling system called "bioluminescence" [14].

As a result of increasing interest in this field, further population-based algorithms have been presented recently. For example, in 2009, Yang and Deb introduced a productive cuckoo search algorithm which was based on laying eggs, nest choosing behaviors and Levy flights of cuckoos [15]. In 2011, Rajabioun also inspired by cuckoo family and developed the evolutionary optimization algorithm, cuckoo optimization algorithm, based on private life styles, laying eggs and special reproduction characteristics of these birds [16]. Inspiring by the pollination process of flowers, Yang presented Flower Pollination Algorithm in 2012 [17].

Moreover, as a method modeling both swarm behavior and evolutionary processes of an organism, bacterial foraging algorithm, which is inspired by foraging behaviors of bacteria in groups, was developed by Passino [18]. Bacterial foraging algorithm is inspired by 'chemotaxis' behaviors of bacteria such as *Escherichia coli* and *Myxococcus xanthus* to perceive chemical changes in the environment and move away or come close (come close to an area having plenty of nutrients by avoiding harmful substances) with certain signals. Foraging process of the algorithm consists of 4 phases: 'chemotaxis', 'swarming', 'reproduction', and 'elimination and dispersal'. 'Chemotaxis' provides the tumbling and swimming movements of bacteria. 'Swarming' models signals of attraction and repulsion between the cells. Through 'Reproduction', cost-effective healthy bacteria are asexually reproduced and yield the most appropriate solution, while unhealthy, high-cost bacteria are eliminated at the same rate. Through 'Elimination and dispersal', some portion of the bacteria is eliminated from the environment and some portion is dispersed to provide diversity [18,19].

Studies in this field in the last 20 years have drawn considerable attention and many of the bio-inspired algorithms have gained popularity mainly because of the flexible and multi-directional structure of these algorithms. Despite the progress provided during the last 20 years, optimization is still a large research area as it is required in all fields of life, especially money, time and energy optimization in engineering and industrial applications. Researchers have been inspired by the most characteristic natural systems and creatures so far. There are many more systems that haven't been inspired yet to develop new optimization algorithms although all those natural systems have ability to optimize their lives themselves [2].

Despite the popularity and development of metaheuristic methods, there still remain many challenging issues. One of these has been the selection of convenient algorithm. Most of the users are confusing in algorithm selection. Selection of the correct optimization algorithm is an optimization problem. There is not a guide of matching the problem style and algorithm. An algorithm may result in worse optimization although it was the best in another similar problem in the literature. This is the nature of nonlinear global optimization, as most of such problems are NP-hard and sometimes there may not be a single convenient algorithm for a particular problem [2]. Therefore, experience and/or trial and error approach have been involved. Instead of implementing a single algorithm for a particular optimization problem, users generally involve several algorithms and select the best of the optimum results of algorithms. This need has been the starting point of establishment of many metaheuristic algorithms.

Algae are one of these natural systems which have not been studied in optimization models yet. In this study, a novel biology-inspired metaheuristic optimization algorithm, named as artificial algae algorithm (AAA) was developed and introduced. It has been developed as a result of inspiration taken from lifestyles of microalgae, which are photosynthetic species. AAA is based on algal reproduction, adaptation, and their swimming which emerges with the motion of being close to light as a photosynthetic organism. An alga grows in the liquid where it exists as long as there is light for photosynthesis and sufficient nutrient, and it reproduces by mitotic division. In case of insufficient light and nutrient conditions species either die or adapt to the changing conditions in the environment. Algae are good swimmers because they continuously swim and try to stay close to water surface to get adequate light. In AAA the global optimum of the objective function was defined as the point on which algae can receive optimum light for photosynthesis. The developed AAA was tested both on 24 benchmark functions and on a constrained design optimization problem, and its performances were compared with some of the other well-known algorithms.

2. Algae

The term 'algae' refers to a diverse group of photosynthetic eucaryotes (except blue-green algae–cyanobacteria) have a discrete nucleus, and chlorophyll, which enables them to synthesize their own food material from CO_2 and H_2O. Unicellular algae (microalgae) are called collectively the phytoplankton. Multicellular algae (macroalgae), the seaweeds, may be meters in length. Algae can survive in many types of environments such as marine, freshwater and terrestrial ecosystems. They can also adapt to extreme habitats such as hot springs and brine lakes [20].

Algae possess an internal green photosynthetic pigment called chlorophyll, held in chloroplasts and sometimes hidden or partially masked by other pigments, which in the presence of sunlight combine CO_2 and H_2O to form starch or related substance, and simultaneously release oxygen [21]. All eucaryotic algae contain chlorophyll *a*; they are quite similar in composition and in their biochemical mechanisms and thus the nutrients necessary for growth (C, H, O, P, N, and trace metals) are essentially the same for all genera [22]. It is known that a great variety of organic compounds can support algal growth in different systems. Numerous algae exhibit a mixed mode of nutrition; that is, photosynthesis in addition to osmotrophy and/or phagotrophy—an ability termed mixotrophy. The frequent occurrence of mixotrophy among algae suggests that it is highly adaptive in aquatic habitats and therefore needs to be considered when making assessments of algal abundance and function in natural systems [20].

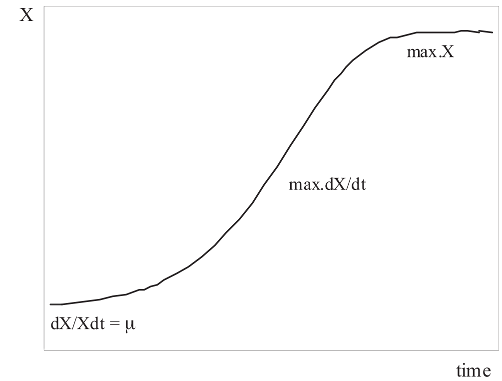

Fig. 1. Typical growth curve for algae [23].

Phytoplankton population dynamics is generally considered under two categories [20]. First is the growth processes including photosynthesis and nutrient uptake, and the second is the loss processes including competition, grazing, sedimentation (sinking to bottom), parasitism, washout and death. Phytoplankton may occupy distinct vertical layers or patches within the epilimnion where light is optimal for growth. Motile species migrate up to the top of the epilimnion for photosynthesis during the day and back to the darker lower levels to acquire nutrients at night. Forward movement of a swimming alga is resisted by the inertial resistance of the fluid moving with an alga during displacement, and the viscous drag that the moving organism exposed. The viscous drag is the rearward force exerted on the organism by the fluid molecules adhering to its surface when it moves within the viscous fluid.

Populations take up space, show various levels of mobility, and are distributed in a variety of patterns in time and space. Populations are characterized by the fact that they grow, i.e., they change in size and density (number per unit area or volume). Growth rate (net change in biomass) represents the balance between additions due to reproduction and losses due to various sources of mortality and export [20].

2.1. Algal growth characteristics

Batch tests results are used to determine the specific growth rates of microbial biomass, i.e., the growth rate per unit of biomass, according to the following equation:

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \mu X \tag{4}$$

where $\mathrm{d}X/\mathrm{d}t$ is the change in biomass per unit time; X is biomass concentration, mg/L; t is time, d; μ is specific growth rate, $1/d$. A typical growth curve for a batch culture growing with an adequate supply of CO_2 and nutrient salts, at an appropriate temperature and under any specified illumination is indicated in Fig. 1.

At low cell concentrations as long as mutual shading is negligible, the increase in cell quantity is exponential and the specific growth rate is constant $\mathrm{d}X/X\mathrm{d}t = \mu$. For a high incident illuminance the exponential phase may continue in spite of mutual shading as long as all cells are maintained above light saturation. As the cell concentration increases, and the fraction of light absorbed approaches 100%, the approximately constant energy income of the culture is reflected in a linear increase in cell quantity, i.e., dX/dt is constant. Thereafter, any further increase in cell quantity merely increases the overhead demands of basal or endogenous metabolism. Whatever the incident illumination, the effective illumination will eventually approach a value, a compensation point for growth, and the quantity of cells in the culture will approach a maximum [23].

The biological growth process is commonly described by the Monod function which relates μ as a function of substrate concentration:

$$\mu = \frac{\mu_{\max} S}{K_s + S} \tag{5}$$

where $\mu_{\max}$ is the maximum specific growth rate (1/time), and K_s is the substrate saturation constant (mass/volume).

Under conditions, where adequate light and nitrogen are present, the growth rate of algae will be limited by the amount of inorganic carbon (total inorganic carbon or dissolved CO_2) available. Mixotrophic and heterotrophic growth of several types of microalgae were investigated by many researchers [24,25]. In mixotrophic cultures photosynthesis and oxidative metabolism of organic carbon functioned simultaneously. Some microalgae can also grow heterotrophically in the dark using organic substrates as carbon and energy sources [26]. The continuous changes of heterotrophic domination to photoauthotrophic domination were also observed in mixotrophic cultures.

2.2. Light and algae

Light is usually measured as irradiance expressed as the number of photons (μeinsteins/m^2 s; 1 Einstein ≈ 6 × 10^{23} photons), or the amount of energy, per unit area per unit of time (W/m^2) [27]. Algae, being photosynthetic, have large quantities of pigments, which also impede light penetration. Algal biomass production is generally limited by light and cultivation of microalgae is faced with problems of light attenuation and the light harvesting capacity. In well-mixed dense cultures of microalgae, where light attenuation takes place a few centimeters below the culture surface and light distribution within the reactor is not homogeneous, average light intensity must be considered, as the algae adapt [28]. As light penetrates the liquid surface, its intensity is reduced exponentially with depth. This exponential nature ensures that, even fairly substantial variations in the irradiance will make little difference to the depth of light penetration. Therefore, light- limited growth rate will be approximately constant per unit surface [22].

2.3. Changes in environmental conditions

Changes in the light intensities, temperature and liquid composition separately, or together in combinations have considerable effects on algal growth characteristics. Such changes cause changes in different metabolic activities. These changes may be photorespiration, photooxidation, photoheterotrophy, photoinhibition. Photorespiration dissipates energy which may prevent the cells from photooxidative damages produced in an O_2-rich atmosphere. Its formation is enhanced by high light together with reduced CO_2 supply or increased oxygen concentration [29]. In some mixotrophic cultures, using organic C sources, the photosynthetic mechanism and the oxidative assimilation of C source function independently and thus, both the photoassimilation of CO_2 and the oxidative assimilation proceed concomitantly [30]. Photoheterotrophy is a form of nutrition in which light is captured and used as an energy source by a pigmented alga, which, takes up dissolved organic compounds from the environment. Photoinhibition is the reduction of photosynthetic rates by high irradiance. It is reported to be a time dependent process that appears to involve a reversible or irreversible inactivation of photosynthetic activities [29].

Microalgae species are all have different metabolic characteristics under different light, temperature and carbon conditions. These characteristics, in turn, determine the dominant specie under certain conditions, e.g., for cold environments and/or for lower light intensities, diatoms were dominant, while for higher temperatures and/or higher light conditions green algae were dominant in the mixed culture [20]. This dominance balances the removal process under different conditions and indicate that mixed culture have ability to adjust itself for changing conditions, and by this property, it is possible to achieve removal under a variety of conditions.

3. Artificial algae algorithm (AAA)

Artificial algae correspond to each solution in the problem space by idealizing the characteristics of algae. Similar to the real algae, artificial algae can move toward the source of light to photosynthesize with helical swimming, and they can adapt to the environment, are able to change the dominant species and can reproduce by mitotic division. Thus, the algorithm was composed of 3 basic parts called “Evolutionary Process”, “Adaptation” and “Helical Movement”.

In the algorithm, algae are the main genera. This whole population was composed of algal colonies. An algal colony is a group of algal cells living together (Eqs. (6) and (7)). When a single algal cell is divided to produce two new algal cells, they live adjacently, and when these two are divided, the new four cells live together and so on. Algal colony behaves like a single cell, moves together, and cells in the colony may die under unsuitable life conditions. An external force like a shear force or some inappropriate conditions may distribute the colony, and each distributed part now become a new colony as life proceeds. The colony existing at the optimum point is named as the colony of optimums and it is composed of the optimum algal cells.

$$\text{Population of Algal colony} = \begin{bmatrix} x_1^1 & \cdots & x_1^D \\ \vdots & \ddots & \vdots \\ x_N^1 & \cdots & x_N^D \end{bmatrix} \tag{6}$$

$$i\text{th algal colony} = \left[x_i^1, x_i^2 \ldots x_i^D\right] \tag{7}$$

where x_i^j is algal cell in jth dimension of ith *algal colony*.

3.1. Evolutionary process

Under sufficient nutrient conditions, if the algal colony receives enough light, it grows and reproduces itself to generate two new algal cells in time t, similar to the real mitotic division. On the contrary, the algal colony not receiving enough light survives for a while but eventually dies. The growth kinetics of algal colony was computed with Monod model given in Eq. (5) where, μ is the specific growth rate, $\mu_{\max}$ is the maximum specific growth rate, S is the nutrient concentration, that is the fitness value ($f^t(x_i)$) in time t in the model and K is the substrate half saturation constant of the algal colony. $\mu_{\max}$ was assumed as 1 (since the maximum quantity converted to biomass should be equal to the quantity of substrate consumed in unit time according to the conservation of mass principle). K was computed as the growth rate at half nutrient conditions of algal colony in time t.

The size of ith algal colony in time $t+1$ in Monod equation is given in the following equation:

$$G_i^{t+1} = \mu_i^t G_i^t \quad i = 1, 2, \ldots N \tag{8}$$

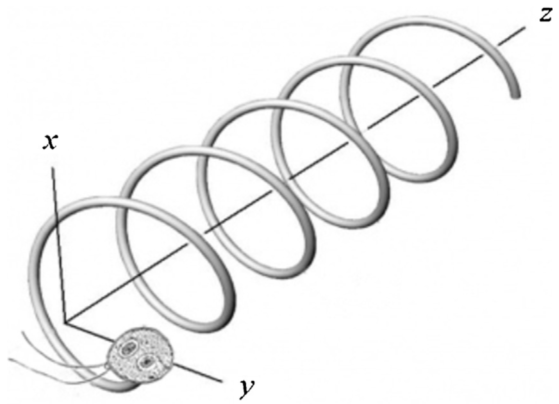

Fig. 2. Movement pattern.

where G_i^t is the size of ith algal colony in time t, N is the number of algal colonies in the system.

Algal colony providing good solutions (cost-efficient and the most appropriate solution) grow more as the amount of nutrient that they obtain is high. For each algal cell of the smallest algal colony dying in the evolutionary process, algal cell of the biggest algal colony is replicated (Eqs. (9)–(11)).

$$\text{biggest}^t = \max \quad G_i^t \quad i = 1, 2, \ldots N \tag{9}$$

$$\text{smallest}^t = \min \quad G_i^t \quad i = 1, 2, \ldots N \tag{10}$$

$$\text{smallest}_m^t = \text{biggest}_m^t, \quad m = 1, 2, \ldots D \tag{11}$$

where D is the problem dimension, biggest is the biggest algal colony and smallest is the smallest one.

In AAA, algal colonies are sorted according to their sizes in time t. In any randomly selected dimension, algal cell of the smallest algal colony dies and algal cell of the biggest colony replicates itself.

3.2. Adaptation

Algal colony, which cannot grow sufficiently in an environment, try to adapt itself to the environment and as a result the dominant species change. Adaptation is the process in which an insufficiently grown algal colony tries to resemble itself to the biggest algal colony in the environment. This process is ended up with the change in starvation level in the algorithm. The initial starvation value is zero for each artificial alga. Starvation value increases with time t, when algal cell receive insufficient light. The artificial alga having the highest starvation value (Eq. (12)) has adapted (Eq. (13)).

$$\text{starving}^t = \max \quad A_i^t, \quad i = 1, 2, \ldots N \tag{12}$$

$$\text{starving}^{t+1} = \text{starving}^t + \left(\text{biggest}^t - \text{starving}^t\right) \times \text{rand} \tag{13}$$

where A_i^t is the starvation value of ith algal colony in time t, starving^t is the algal colony with the highest starvation value in time t. The adaptation parameter (A_p) determines whether adaptation process would be applied in time t or not. A_p is constant on the interval [0, 1].

3.3. Helical movement

Algal cells and colonies generally swim and try to stay close to the water surface because of adequate light for survival is available there. They swim helically in the liquid with their flagella (Fig. 2) which provide forward movement that is restricted by gravity and the viscous drag.

Movements of algal cell differ. As friction surface of growing algal cell gets larger, the frequency of helical movements increases by increasing their local search ability. Each algal cell can move proportional to its energy. The energy of an algal cell at time t is directly proportional to the amount of nutrient uptake in that time. Therefore, the more an algal cell is closer to the surface, the more energy it has, and it finds more chance to move inside the liquid. On the contrary, friction surface is less; their movement distance in the liquid is longer. Therefore, their global search ability is more. However, they can move less in proportion to its energy.

The movement of an algal cell is helical as it is in the real life. In AAA, the gravity restricting the movement is displayed as 0 and viscous drag is displayed as shear force, which is proportional to the size of algal cell. It is in spherical shape and the size of it is its volume in the model. Therefore, friction surface becomes the surface area of the hemisphere (Eqs. (14) and (15)).

$$\tau(x_i) = 2\pi r^2 \tag{14}$$

$$\tau(x_i) = 2\pi \left(\sqrt[3]{\frac{3G_i}{4\pi}} \right)^2 \tag{15}$$

where $\tau(x_i)$ is the friction surface.

Three dimensions for the helical movement of the algal cell are determined randomly. One of these provides linear movement in Eq. (16) and other two dimensions provide angular movement in Eqs. (17) and (18). Eq. (16) is used for one-dimensional problems and algal cell/colony moves in single direction. In two-dimensional problems, algal movement is sinusoidal, and therefore Eqs. (16) and (18) are used. In case of three or more dimensions, algal movement is helical and Eqs. (16)–(18) are used. Friction surface and distance to the source of light determine the step size of the movement:

$$x_{im}^{t+1} = x_{im}^t + \left(x_{jm}^t - x_{im}^t\right)\left(\Delta - \tau^t(x_i)\right) p \tag{16}$$

$$x_{ik}^{t+1} = x_{ik}^t + \left(x_{jk}^t - x_{ik}^t\right)\left(\Delta - \tau^t(x_i)\right) \cos\alpha \tag{17}$$

$$x_{il}^{t+1} = x_{il}^t + \left(x_{jl}^t - x_{il}^t\right)\left(\Delta - \tau^t(x_i)\right) \sin\beta \tag{18}$$

where X_{ik}^t, X_{il}^t and X_{im}^t are x, y and z coordinates of ith algal cell at time t; $\alpha, \beta \in [0, 2\pi]$; $p \in [-1, 1]$; Δ is shear force; $\tau^t(x_i)$ is the friction surface area of ith algal cell.

3.4. Pseudo code and flowchart of AAA

Pseudo code and flowchart of AAA are given in Figs. 3 and 4, respectively.

4. Experimental studies

Experimental studies performed to verify the success of this new algorithm and to compare it with other methods. Comparisons were based upon the other similar algorithms, namely ABC, BA, ACO_R and SPSO (version 2011) which based on mimicking the foraging behavior of a living organism. Moreover, DE which based on simulating an evolutionary process in nature and population based HS_{POP} algorithms were involved in comparisons. Two different phases of verification will be presented here. The first phase was conducted on various benchmark functions with various dimensions. The results were compared to some well-known metaheuristic methods in the literature, ABC [31,32,12], BA [32,13], DE [32,8], as well as ACO_R [32,33], which is an approach developed to solve continuous optimization problems, and population-based harmony search (HS_{POP}) algorithm that was proposed by Mukhopadhyay et al. [34]. In the second phase, AAA was implemented on the pressure vessel problem, a well-known design optimization problem, and

Objective function f(x), $x=(x_1,x_2,\ldots,x_d)$

Initialize a population of n algal colonies with random solutions

Evaluate size (G) of n algal colonies

Define Algorithm parameters (shear force Δ*, loss of energy e and Adaptation parameter* A_p*)*

While *(t < MaxCalculation)*

Evaluate energy (E) and friction surface (τ) *of n algae*

For *i=1:n*

Starvation is true

While $(E(x_i) > 0)$

Choose j among all solution via tournament selection method

Choose randomly three dimensions to helical movement, k, l and m

$$x_{ik}^{t+1} = x_{ik}^t + \left(x_{jk}^t - x_{ik}^t\right)(\Delta - \tau_i)\cos\alpha$$

$$x_{il}^{t+1} = x_{il}^t + \left(x_{jl}^t - x_{il}^t\right)(\Delta - \tau_i)\sin\beta$$

$$x_{im}^{t+1} = x_{im}^t + \left(x_{jm}^t - x_{im}^t\right)(\Delta - \tau_i)p$$

α, β are random angles in the range [0,2π] and p is a random number in the range [-1, 1]

Evaluate new solution

$E(x_i) = E(x_i) - (\frac{e}{2})$ *energy loss caused by movement*

if *new solution is better, update algal colony i and starvation is false*

else $E(x_i) = E(x_i) - (\frac{e}{2})$ *energy loss caused by metabolism* **end if**

end While

if *starvation is true, increase starvation* $A(x_i)$ **end if**

end For

Evaluate size (G) of population

Choose one dimension to reproduction, r

$smallest_r^t = biggest_r^t$

if $rand<A_p$

$starving^{t+1} = starving^t + (biggest^t - starving^t) \times rand$

end if

Find the current best solution

end While

Fig. 3. Pseudo code of the proposed artificial algae algorithm (AAA).

the test results were compared with the algorithms ABC and SPSO [35,36].

4.1. Performance of AAA on benchmark functions

There are many benchmark functions in order to confirm the success of a new algorithm. There is not a unique agreed benchmark function set in the literature. CEC'05 [37] benchmark function set is commonly used for confirmation. This set provides a class of shifted and/or rotated functions, has three different function classes as unimodal (F1–F5), multi-modal (F6–F14) and hybrid (F15–F25). The functions in each class are as follows:

24 CEC'05 Test Functions

Unimodal Functions (5)

F1: Shifted Sphere Function (Bounds [−100,100], Bias = −450)

F2: Shifted Schwefel's Problem 1.2 (Bounds [−100,100], Bias = −450)

F3: Shifted Rotated High Conditioned Elliptic Function (Bounds [−100,100], Bias = −450)

F4: Shifted Schwefel's Problem 1.2 with Noise in Fitness (Bounds [−100,100], Bias = −450)

F5: Schwefel's Problem 2.6 with Global Optimum on Bounds (Bounds [−100,100], Bias = −310)

Multimodal Functions (20):

Basic Functions (7)

F6: Shifted Rosenbrock's Function (Bounds [−100,100], Bias = 390)

F7: Shifted Rotated Griewank's Function without Bounds (Initialization Range [0, 600], Bias = −180)

F8: Shifted Rotated Ackley's Function with Global Optimum on Bounds (Bounds [−32,32], Bias = −140)

F9: Shifted Rastrigin's Function (Bounds [−5,5], Bias = −330)

F10: Shifted Rotated Rastrigin's Function (Bounds [−5,5], Bias = −330)

F11: Shifted Rotated Weierstrass Function (Bounds [−0.5,0.5], Bias = 90)

F12: Schwefel's Problem 2.13 (Bounds [−100,100], Bias = −460)

Expanded Functions (2)

F13: Expanded Extended Griewank's plus Rosenbrock's Function (F8F2) (Bounds [−3,1], Bias = −130)

F14: Shifted Rotated Expanded Scaffer's F6 (Bounds [−100,100], Bias = −300)

Hybrid Composition Functions (11):

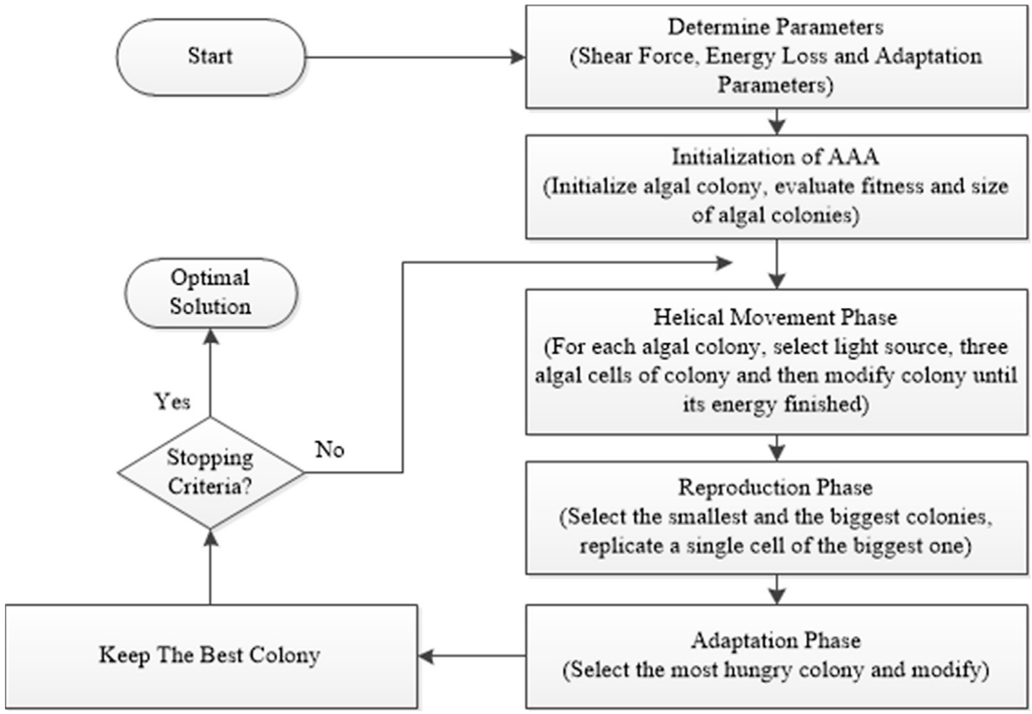

Fig. 4. General flowchart of the AAA.

F15: Hybrid Composition Function (Bounds [−5,5], Bias = 120)

F16: Rotated Hybrid Composition Function (Bounds [−5,5], Bias = 120)

F17: Rotated Hybrid Composition Function with Noise in Fitness (Bounds [−5,5], Bias = 120)

F18: Rotated Hybrid Composition Function (Bounds [−5,5], Bias = 10)

F19: Rotated Hybrid Composition Function with a Narrow Basin for the Global Optimum (Bounds [−5,5], Bias = 10)

F20: Rotated Hybrid Composition Function with the Global Optimum on the Bounds (Bounds [−5,5], Bias = 10)

F21: Rotated Hybrid Composition Function (Bounds [−5,5], Bias = 360)

F22: Rotated Hybrid Composition Function with High Condition Number Matrix (Bounds [−5,5], Bias = 360)

F24: Rotated Hybrid Composition Function (Bounds [−5,5], Bias = 260)

F25: Rotated Hybrid Composition Function without Bounds (Initialization Range [−2,5] Bias = 260)

4.1.1. Parameters used on benchmark functions

In this study, to compare the results correctly for each algorithm:

- The same parameter settings as El-Abd [32] were followed where population size was set to 40. The number of population was fixed as 40 in applying ABC, BA, ACO_R, DE and HS_{POP} algorithms for more comparable results.
- Number of function evaluation (cost of calculation) was used as the stopping criterion. Maximum number of function evaluations for 10, 30 and 50 dimensions were determined as 100,000, 300,000 and 500,000, respectively.
- Initial candidate solutions were randomly started between lower and upper limits of the function by uniform distribution.
- Metaheuristic algorithms may found slightly different optimums in each run. Therefore, in order to obtain more precise optimums, the average of results of 30 runs for each function was reported as the ultimate result.

Besides these general conditions, some model-specific conditions were as follows:

- For ABC algorithm, the recommendations in [31,32,12] were followed by setting the limit parameter to number of onlooker bees multiplied by Dimension.
- For BA, the same parameter settings as in El-Abd [32] were followed:
  The number of best selected patches, $m = 30$,
  The number of elite selected patches, $e = 10$,
  The number of recruited bees around best selected patches, $n1 = 15$,
  The number of recruited bees around elite selected patches, $n2 = 30$,
  The patch radius for neighborhood search, ngh = 1%.
- Parameter settings in DE algorithm were as follows [32]:
  F_weight step size = 0.9,
  Crossover probability constant, F_CR = 0.1.
- For HS_{POP} Algorithm the harmony memory considering rate (HMCR) was set as 0.99 and the pitch adjusting rate (PAR) was set to 0.5 [32].
- For ACO_R Algorithm, the number of ants used in an iteration $m = 2$, an archive size $k = 50$, the Locality of the search process, $q = 0.0001$ and the Speed of convergence, $\xi = 0.85$. These selections were also consistent with the settings suggested by El-Abd [32].
- For AAA, as a result of preliminary runs the parameters were set as follows: the loss of energy, $e = 0.3$, the shear force, $\Delta = 2$ and the adaptation probability constant, $A_p = 0.5$.

4.1.2. Comparison results on benchmark functions

Experimental results of the algorithms tested by CEC'05 functions are presented in Tables 1–3 for uni-modal, Tables 4–6 for multi-modal and Tables 7–9 for hybrid functions. In these tables, the mean and the best results and standard deviations of 30 runs of each function, are indicated for each tested algorithm. The best average value of each function is indicated in bold. Additionally, Figs. 5–7 indicate the convergence speeds of uni-modal,

Table 1
Mean, best and standard deviation values of solutions achieved for CEC05 uni-modal functions taken over 30 runs at 10 dimensions.

		AAA	ABC	BA	DE	ACO_R	HS_{POP}
F1	Best	0.000E+00	0.000E+00	2.997E+00	0.000E+00	0.000E+00	0.000E+00
	Mean	**0.000E+00**	1.895E−15	4.233E+03	**0.000E+00**	2.274E−14	**0.000E+00**
	Std.	0.000E+00	1.038E−14	3.812E+03	0.000E+00	3.533E−14	0.000E+00
F2	Best	1.541E−08	3.077E−01	4.257E+00	5.013E+00	0.000E+00	4.607E−03
	Mean	1.184E−06	3.322E+00	4.898E+03	1.403E+01	**1.478E−13**	2.774E+01
	Std.	1.757E−06	2.893E+00	3.547E+03	5.389E+00	1.263E−13	3.481E+01
F3	Best	5.199E+04	1.860E+05	3.169E+05	3.482E+05	8.126E+04	3.230E+04
	Mean	3.111E+05	7.224E+05	1.528E+07	1.753E+06	2.238E+06	**2.196E+05**
	Std.	2.441E+05	3.527E+05	1.959E+07	6.862E+05	2.689E+06	1.723E+05
F4	Best	4.191E−05	2.894E+02	1.597E+03	5.814E+01	5.684E−14	8.339E−02
	Mean	8.199E−03	1.327E+03	8.885E+03	1.483E+02	**3.752E−13**	5.989E+01
	Std.	9.187E−03	6.884E+02	4.665E+03	6.227E+01	3.221E−13	1.074E+02
F5	Best	0.000E+00	8.169E+00	8.165E+02	1.206E−01	3.456E+01	1.855E−10
	Mean	**1.601E−11**	7.750E+01	8.168E+03	3.384E+00	3.938E+02	5.555E+01
	Std.	3.433E−11	8.828E+01	3.946E+03	3.175E+00	4.884E+02	1.058E+02

Table 2
Mean, best and standard deviation values of solutions achieved for CEC05 uni-modal functions taken over 30 runs at 30 dimensions.

		AAA	ABC	BA	DE	ACO_R	HS_{POP}
F1	Best	5.684E−14	1.137E−13	2.412E+01	5.684E−14	5.684E−14	0.000E+00
	Mean	5.874E−14	1.213E−13	3.730E+04	5.684E−14	1.819E−13	**0.000E+00**
	Std.	1.038E−14	1.965E−14	2.362E+04	2.568E−29	9.605E−14	0.000E+00
F2	Best	1.536E+00	3.035E+02	2.145E+02	1.518E+03	2.028E−06	1.100E+01
	Mean	1.043E+01	2.256E+03	4.209E+04	2.966E+03	**3.866E−03**	2.030E+02
	Std.	7.792E+00	9.365E+02	2.298E+04	6.281E+02	7.647E−03	1.851E+02
F3	Best	1.486E+06	3.800E+06	1.506E+07	1.463E+07	5.022E+07	1.502E+06
	Mean	**3.059E+06**	7.514E+06	3.100E+08	2.782E+07	3.215E+08	3.319E+06
	Std.	9.623E+05	1.941E+06	3.166E+08	8.752E+06	1.833E+08	1.201E+06
F4	Best	2.319E+03	2.880E+04	3.091E+04	9.640E+03	1.216E+01	2.294E+01
	Mean	8.573E+03	4.251E+04	7.579E+04	1.474E+04	4.836E+02	**2.147E+02**
	Std.	3.766E+03	5.753E+03	1.638E+04	2.979E+03	7.762E+02	1.894E+02
F5	Best	2.699E+03	6.775E+03	9.901E+03	3.988E+03	1.600E+03	1.056E+03
	Mean	4.171E+03	9.508E+03	2.381E+04	5.516E+03	4.836E+03	**2.088E+03**
	Std.	9.825E+02	1.215E+03	7.818E+03	6.664E+02	2.130E+03	4.159E+02

Table 3
Mean, best and standard deviation values of solutions achieved for CEC05 uni-modal functions taken over 30 runs at 50 dimensions.

		AAA	ABC	BA	DE	ACO_R	HS_{POP}
F1	Best	1.137E−13	2.274E−13	5.949E+01	5.684E−14	1.137E−13	0.000E+00
	Mean	1.364E−13	2.785E−13	7.612E+04	**1.080E−13**	3.051E−13	1.426E−06
	Std.	3.202E−14	4.047E−14	5.591E+04	1.734E−14	1.526E−13	7.813E−06
F2	Best	1.211E+02	8.099E+03	9.133E+02	2.083E+04	2.710E+03	5.239E+01
	Mean	3.768E+02	1.551E+04	9.981E+04	2.765E+04	6.715E+04	**2.998E+02**
	Std.	1.769E+02	2.976E+03	7.231E+04	3.221E+03	6.757E+04	2.670E+02
F3	Best	2.438E+06	8.639E+06	1.259E+07	5.499E+07	3.137E+08	2.283E+06
	Mean	4.806E+06	1.439E+07	1.737E+09	1.015E+08	1.913E+09	**4.213E+06**
	Std.	1.217E+06	3.129E+06	1.538E+09	2.410E+07	1.167E+09	1.559E+06
F4	Best	2.288E+04	9.732E+04	1.055E+05	4.510E+04	3.137E+04	7.575E+02
	Mean	3.925E+04	1.194E+05	2.061E+05	6.063E+04	1.463E+05	**2.417E+03**
	Std.	1.045E+04	9.750E+03	4.876E+04	8.404E+03	9.059E+04	9.874E+02
F5	Best	6.491E+03	1.895E+04	1.227E+04	1.114E+04	5.097E+03	2.045E+03
	Mean	1.148E+04	2.365E+04	3.103E+04	1.328E+04	1.108E+04	**3.426E+03**
	Std.	1.717E+03	2.109E+03	9.531E+03	1.036E+03	3.916E+03	5.726E+02

multi-modal and hybrid functions, respectively, for all algorithms except BA which was not as successful as the others.

Uni-modal functions are the problems including exactly one local minimum or maximum and they have wide search space. These functions do have the problem of being attached to local minima. Therefore, HS_{POP} algorithm, which keeps the individuals producing good results in the harmony memory and adapting other individuals to the memory, achieved great success on such functions. AAA also obtained successful results on uni-modal functions (Tables 1–3). It provided the best values in F1 and F5 functions over 10 dimensions, in F3 function over 30 dimensions. It is also observed that AAA provided the closest results to the best results obtained in other functions. AAA equalizes the disadvantage of not using memory in choosing source of light, and slowly approaching to light through helical movement with its adaptation and evolution process. This balance is the reason of results that were the best or closest to the best.

In establishing convergence curves (Fig. 5), the functions providing similar results are selected. Tables 1–3 and Fig. 5 indicated that AAA resulted in good optimizations as well as good convergence speeds. Fig. 5 presents the convergence history of the average of 30 independent runs for all studied algorithms. Convergence curves indicate that AAA methodically approached global optimum by avoiding local optimums. When convergence rate of AAA is compared to other algorithms, it is seen that AAA can reach optimum solution with less calculation. As it is indicated in Fig. 5, AAA kept this success in different functions and different dimensions.

Table 4
Mean, best and standard deviation values of solutions achieved for CEC05 Multi-modal functions taken over 30 runs at 10 dimensions.

		AAA	ABC	BA	DE	ACO_R	HS_{POP}
F6	Best	1.623E − 06	5.465E − 02	7.899E + 02	3.020E − 01	3.661E − 06	5.533E − 01
	Mean	**1.219E + 00**	1.491E + 00	2.936E + 08	3.033E + 00	8.611E + 01	2.709E + 01
	Std.	1.491E + 00	2.487E + 00	4.188E + 08	2.213E + 00	4.412E + 02	2.839E + 01
F7	Best	1.267E + 03	1.267E + 03	1.560E + 02	2.363E − 01	3.161E − 01	1.267E + 03
	Mean	1.267E + 03	1.267E + 03	1.645E + 03	**4.106E − 01**	8.581E − 01	1.267E + 03
	Std.	3.288E − 02	2.313E − 13	7.414E + 02	9.131E − 02	2.913E − 01	1.094E − 02
F8	Best	2.002E + 01	2.019E + 01	2.017E + 01	2.021E + 01	2.014E + 01	2.018E + 01
	Mean	**2.016E + 01**	2.033E + 01	2.034E + 01	2.040E + 01	2.035E + 01	2.033E + 01
	Std.	8.695E − 02	7.863E − 02	7.941E − 02	6.267E − 02	8.296E − 02	5.667E − 02
F9	Best	0.000E + 00	0.000E + 00	1.766E + 01	0.000E + 00	2.985E + 00	0.000E + 00
	Mean	**0.000E + 00**	**0.000E + 00**	5.135E + 01	**0.000E + 00**	7.735E + 00	2.801E − 07
	Std.	0.000E + 00	0.000E + 00	1.820E + 01	0.000E + 00	3.603E + 00	1.534E − 06
F10	Best	4.975E + 00	1.008E + 01	3.757E + 01	1.144E + 01	7.091E + 00	1.751E + 01
	Mean	**1.501E + 01**	2.518E + 01	7.113E + 01	1.911E + 01	2.340E + 01	2.193E + 01
	Std.	5.593E + 00	7.635E + 00	2.512E + 01	3.599E + 00	7.573E + 00	2.371E + 00
F11	Best	1.499E + 00	4.175E + 00	6.003E + 00	4.875E + 00	4.981E + 00	8.113E + 00
	Mean	**3.667E + 00**	5.415E + 00	9.360E + 00	6.102E + 00	8.604E + 00	9.221E + 00
	Std.	9.292E − 01	7.297E − 01	1.518E + 00	6.214E − 01	9.727E − 01	4.890E − 01
F12	Best	2.439E + 00	8.504E + 01	6.907E + 03	1.715E + 02	1.349E + 04	4.466E + 01
	Mean	5.435E + 02	**3.070E + 02**	3.014E + 04	4.341E + 02	2.923E + 04	3.168E + 03
	Std.	7.766E + 02	1.634E + 02	8.858E + 03	1.850E + 02	6.722E + 03	3.135E + 03
F13	Best	1.228E − 01	3.125E − 02	4.722E + 00	7.506E − 02	8.360E − 01	2.048E − 01
	Mean	4.231E − 01	**2.241E − 01**	9.636E + 00	2.936E − 01	1.692E + 00	8.897E − 01
	Std.	1.329E − 01	8.985E − 02	3.494E + 00	1.176E − 01	5.347E − 01	4.433E − 01
F14	Best	2.698E + 00	2.992E + 00	3.256E + 00	3.174E + 00	3.219E + 00	1.170E + 00
	Mean	3.296E + 00	3.412E + 00	3.940E + 00	3.459E + 00	3.800E + 00	**2.485E + 00**
	Std.	2.823E − 01	1.441E − 01	2.307E − 01	1.299E − 01	2.861E − 01	6.736E − 01

Multi-modal functions are the functions including multiple local minima or maxima and a single global minimum. These functions are difficult problems as they possess many local minima. The performance of AAA appeared better in multi-modal functions. The algorithm displayed the best performance over 14 of 27 tests, on 10, 30 and 50 dimensions (Tables 4–6). It provided nearly best values over other tests with the exception of 30 and 50 dimensions of F7 and F9 functions. The success of the algorithm on average values becomes more evident with its standard deviation and speed of convergence indicated in Fig. 6. This success of AAA arises from its ability to balance local search and global search. Unlike uni-modal functions, in multi-modal functions the light source selection method and helical movement model of artificial algae becomes an advantage in such difficult problems having lots of local minima. Because updated information utilization in choosing light source helps avoid local minima. Furthermore, a better search can be carried out in the search space by means of helical movement. Each algal colony finds opportunity to catch global optimums on its course by means of its movement towards light.

AAA produced successful results on hybrid functions as well as uni-modal and multi-modal functions as compared to other algorithms. HS_{POP} algorithm, for instance, was successful on uni-modal

Table 5
Mean, best and standard deviation values of solutions achieved for CEC05 Multi-modal functions taken over 30 runs at 30 dimensions.

		AAA	ABC	BA	DE	ACO_R	HS_{POP}
F6	Best	8.076E − 04	5.616E − 02	1.864E + 04	2.240E + 01	8.643E − 05	2.718E + 01
	Mean	1.384E + 01	**5.506E + 00**	2.150E + 10	3.970E + 01	3.914E + 01	1.306E + 02
	Std.	2.177E + 01	5.613E + 00	1.786E + 10	1.961E + 01	7.112E + 01	1.218E + 02
F7	Best	4.696E + 03	4.696E + 03	8.685E + 02	9.916E − 02	1.535E − 12	4.696E + 03
	Mean	4.696E + 03	4.696E + 03	7.393E + 03	**1.922E − 01**	3.396E − 01	4.696E + 03
	Std.	0.000E + 00	0.000E + 00	2.896E + 03	4.537E − 02	3.252E − 01	0.000E + 00
F8	Best	2.009E + 01	2.080E + 01	2.081E + 01	2.074E + 01	2.083E + 01	2.080E + 01
	Mean	**2.018E + 01**	2.091E + 01	2.095E + 01	2.095E + 01	2.095E + 01	2.094E + 01
	Std.	7.323E − 02	4.794E − 02	4.464E − 02	6.506E − 02	4.525E − 02	5.480E − 02
F9	Best	5.684E − 14	5.684E − 14	1.709E + 02	5.684E − 14	2.288E + 01	0.000E + 00
	Mean	3.317E − 02	9.853E − 14	3.216E + 02	**5.684E − 14**	4.364E + 01	1.088E + 01
	Std.	1.817E − 01	3.636E − 14	8.604E + 01	2.568E − 29	2.285E + 01	1.745E + 01
F10	Best	6.954E + 01	2.021E + 02	3.084E + 02	1.294E + 02	1.803E + 02	1.487E + 02
	Mean	**1.535E + 02**	3.018E + 02	5.374E + 02	1.642E + 02	2.258E + 02	1.628E + 02
	Std.	3.947E + 01	4.591E + 01	1.017E + 02	1.593E + 01	2.190E + 01	7.746E + 00
F11	Best	1.412E + 01	2.451E + 01	2.309E + 01	2.468E + 01	3.716E + 01	3.597E + 01
	Mean	**2.241E + 01**	2.763E + 01	3.604E + 01	2.814E + 01	3.942E + 01	3.961E + 01
	Std.	2.976E + 00	1.408E + 00	5.183E + 00	1.613E + 00	9.357E − 01	1.088E + 00
F12	Best	4.578E + 02	1.022E + 04	8.780E + 05	1.955E + 04	8.973E + 05	3.525E + 05
	Mean	**1.357E + 04**	1.822E + 04	1.094E + 06	3.657E + 04	1.075E + 06	5.266E + 05
	Std.	1.171E + 04	4.421E + 03	1.093E + 05	8.603E + 03	7.674E + 04	6.509E + 04
F13	Best	1.083E + 00	1.008E + 00	3.175E + 01	1.948E + 00	8.707E + 00	7.379E + 00
	Mean	1.959E + 00	**1.264E + 00**	1.495E + 02	2.474E + 00	1.424E + 01	1.163E + 01
	Std.	5.224E − 01	1.314E − 01	1.491E + 02	2.275E − 01	2.461E + 00	1.507E + 00
F14	Best	1.128E + 01	1.229E + 01	1.295E + 01	1.252E + 01	1.334E + 01	1.286E + 01
	Mean	**1.269E + 01**	1.293E + 01	1.364E + 01	1.291E + 01	1.386E + 01	1.329E + 01
	Std.	4.711E − 01	2.703E − 01	2.936E − 01	1.868E − 01	1.873E − 01	1.552E − 01

Table 6
Mean, best and standard deviation values of solutions achieved for CEC05 Multi-modal functions taken over 30 runs at 50 dimensions.

		AAA	ABC	BA	DE	ACO_R	HS_{POP}
F6	Best	8.878E − 02	3.708E − 02	7.996E + 04	4.092E + 01	7.682E − 05	4.154E + 01
	Mean	3.144E + 01	**9.970E + 00**	5.817E + 10	6.137E + 01	7.218E + 02	1.421E + 02
	Std.	3.893E + 01	1.264E + 01	3.598E + 10	2.454E + 01	2.154E + 03	6.366E + 01
F7	Best	6.195E + 03	6.195E + 03	5.431E + 02	1.577E − 01	1.307E − 12	6.195E + 03
	Mean	6.195E + 03	6.195E + 03	1.203E + 04	2.302E − 01	**1.629E − 01**	6.195E + 03
	Std.	9.250E − 13	9.250E − 13	3.545E + 03	4.217E − 02	3.218E − 01	9.250E − 13
F8	Best	2.008E + 01	2.103E + 01	2.101E + 01	2.104E + 01	2.103E + 01	2.106E + 01
	Mean	**2.015E + 01**	2.111E + 01	2.112E + 01	2.114E + 01	2.113E + 01	2.115E + 01
	Std.	4.601E − 02	3.518E − 02	4.434E − 02	3.539E − 02	3.592E − 02	2.882E − 02
F9	Best	1.137E − 13	1.705E − 13	3.511E + 02	1.137E − 13	4.378E + 01	5.684E − 14
	Mean	5.313E − 01	3.676E − 13	6.144E + 02	**1.137E − 13**	1.217E + 02	8.765E + 01
	Std.	6.787E − 01	3.084E − 13	1.720E + 02	1.027E − 28	6.944E + 01	4.698E + 01
F10	Best	2.098E + 02	7.532E + 02	9.126E + 02	3.223E + 02	4.196E + 02	3.037E + 02
	Mean	3.692E + 02	9.341E + 02	1.132E + 03	4.070E + 02	4.594E + 02	**3.312E + 02**
	Std.	7.344E + 01	9.031E + 01	1.438E + 02	3.067E + 01	3.348E + 01	1.179E + 01
F11	Best	3.567E + 01	5.044E + 01	4.933E + 01	5.677E + 01	6.836E + 01	6.885E + 01
	Mean	**4.694E + 01**	5.507E + 01	6.544E + 01	6.181E + 01	7.265E + 01	7.237E + 01
	Std.	4.579E + 00	2.110E + 00	8.736E + 00	2.044E + 00	1.392E + 00	1.457E + 00
F12	Best	3.988E + 03	5.012E + 04	4.351E + 06	2.864E + 05	4.749E + 06	2.209E + 06
	Mean	**4.270E + 04**	8.320E + 04	5.396E + 06	3.284E + 05	5.428E + 06	2.544E + 06
	Std.	2.830E + 04	1.687E + 04	5.832E + 05	2.954E + 04	3.783E + 05	1.912E + 05
F13	Best	2.549E + 00	2.020E + 00	7.092E + 01	6.816E + 00	2.501E + 01	2.325E + 01
	Mean	3.513E + 00	**2.466E + 00**	7.074E + 02	8.111E + 00	3.326E + 01	2.674E + 01
	Std.	6.883E − 01	2.305E − 01	6.084E + 02	5.643E − 01	4.138E + 00	1.355E + 00
F14	Best	2.162E + 01	2.225E + 01	2.257E + 01	2.282E + 01	2.316E + 01	2.290E + 01
	Mean	**2.247E + 01**	2.267E + 01	2.343E + 01	2.316E + 01	2.376E + 01	2.315E + 01
	Std.	3.836E − 01	1.797E − 01	2.832E − 01	1.450E − 01	1.821E − 01	1.414E − 01

Table 7
Mean, best and standard deviation values of solutions achieved for CEC05 hybrid functions taken over 30 runs, 10 dimensions.

		AAA	ABC	BA	DE	ACO_R	HS_{POP}
F15	Best	0.000E + 00	0.000E + 00	4.100E + 02	4.867E − 01	1.026E + 02	0.000E + 00
	Mean	3.311E + 01	**7.329E − 02**	5.921E + 02	1.653E + 01	4.355E + 02	2.782E + 02
	Std.	3.778E + 01	3.227E − 01	9.820E + 01	1.813E + 01	1.876E + 02	1.786E + 02
F16	Best	9.222E + 01	1.238E + 02	1.385E + 02	1.107E + 02	1.066E + 02	1.159E + 02
	Mean	**1.293E + 02**	1.476E + 02	3.212E + 02	1.443E + 02	2.055E + 02	1.380E + 02
	Std.	1.596E + 01	1.384E + 01	7.830E + 01	1.475E + 01	1.178E + 02	1.030E + 01
F17	Best	9.984E + 01	1.404E + 02	1.659E + 02	1.384E + 02	1.215E + 02	1.285E + 02
	Mean	**1.353E + 02**	1.694E + 02	3.442E + 02	1.730E + 02	1.890E + 02	1.507E + 02
	Std.	1.825E + 01	1.529E + 01	8.931E + 01	1.422E + 01	5.524E + 01	1.146E + 01
F18	Best	3.000E + 02	4.035E + 02	9.685E + 02	5.114E + 02	7.794E + 02	6.269E + 02
	Mean	**4.864E + 02**	5.086E + 02	1.104E + 03	7.459E + 02	9.317E + 02	8.541E + 02
	Std.	2.242E + 02	7.031E + 01	5.856E + 01	1.012E + 02	9.863E + 01	1.013E + 02
F19	Best	3.000E + 02	4.349E + 02	9.810E + 02	3.740E + 02	8.001E + 02	3.002E + 02
	Mean	**4.529E + 02**	5.213E + 02	1.111E + 03	6.980E + 02	9.552E + 02	8.265E + 02
	Std.	1.993E + 02	7.592E + 01	6.753E + 01	1.363E + 02	6.679E + 01	1.545E + 02
F20	Best	3.000E + 02	5.000E + 02	8.019E + 02	5.007E + 02	7.244E + 02	5.375E + 02
	Mean	**4.842E + 02**	5.430E + 02	1.097E + 03	7.536E + 02	9.396E + 02	8.674E + 02
	Std.	2.135E + 02	1.075E + 02	8.884E + 01	1.110E + 02	9.376E + 01	1.019E + 02
F21	Best	3.000E + 02	2.035E + 02	5.037E + 02	2.270E + 02	5.000E + 02	5.000E + 02
	Mean	5.233E + 02	**3.459E + 02**	1.266E + 03	4.642E + 02	1.088E + 03	1.038E + 03
	Std.	1.006E + 02	9.939E + 01	1.574E + 02	9.040E + 01	2.130E + 02	1.990E + 02
F22	Best	3.000E + 02	2.008E + 02	9.150E + 02	7.760E + 02	5.318E + 02	7.574E + 02
	Mean	7.347E + 02	**7.084E + 02**	1.025E + 03	7.941E + 02	8.315E + 02	7.873E + 02
	Std.	1.196E + 02	1.905E + 02	6.621E + 01	7.699E + 00	1.010E + 02	2.429E + 01
F24	Best	2.000E + 02	2.000E + 02	1.110E + 03	2.000E + 02	3.744E + 02	2.000E + 02
	Mean	**2.000E + 02**	**2.000E + 02**	1.264E + 03	**2.000E + 02**	6.038E + 02	2.400E + 02
	Std.	0.000E + 00	0.000E + 00	5.919E + 01	1.201E − 02	2.904E + 02	1.038E + 02
F25	Best	8.120E + 02	6.176E + 02	1.303E + 03	8.183E + 02	3.727E + 02	2.000E + 02
	Mean	8.170E + 02	7.689E + 02	1.390E + 03	8.272E + 02	5.708E + 02	**3.196E + 02**
	Std.	2.464E + 00	9.427E + 01	4.070E + 01	3.619E + 00	2.967E + 02	1.165E + 02

functions (Tables 1–3), however, on multi-modal (Tables 4–6) and hybrid functions (Tables 7–9) it was not so much successful. Similarly, ACO_R algorithm was successful on hybrid function but it did not produce good results for multi-modal functions. In addition, ABC produced results similar to AAA on multi-modal and hybrid functions but it could not produce successful results for uni-modal functions.

Single-problem analysis can be used for pairwise comparison of the problem-solving success of computational intelligence experiments. The analysis uses results of global minimum values obtained from several runs of the algorithm over a given problem. In this study the Wilcoxon Signed-Rank Test was used for pairwise comparisons (Tables 10–12 for 10, 30 and 50 dimensions, respectively). The test was performed by using the global minimum values obtained as a result of 30 runs for problem-based pairwise comparison of the algorithms. Hypothesis test can be used to make inferences about two algorithms from the given result. For this purpose, H0 (Null hypothesis) was: 'There is no

Table 8
Mean, best and standard deviation values of solutions achieved for CEC05 hybrid functions taken over 30 runs, 30 dimensions.

		AAA	ABC	BA	DE	ACO_R	HS_{POP}
F15	Best	0.000E+00	1.791E−11	5.032E+02	1.278E+01	2.494E+02	2.000E+02
	Mean	1.844E+02	**9.010E+00**	8.024E+02	6.952E+01	5.771E+02	3.000E+02
	Std.	1.646E+02	1.130E+01	1.972E+02	4.279E+01	1.718E+02	2.626E+01
F16	Best	1.222E+02	2.160E+02	3.896E+02	1.646E+02	2.183E+02	1.671E+02
	Mean	**1.948E+02**	3.221E+02	6.639E+02	2.114E+02	4.591E+02	2.635E+02
	Std.	6.580E+01	4.742E+01	1.208E+02	2.868E+01	1.870E+02	1.201E+02
F17	Best	1.099E+02	2.660E+02	5.002E+02	2.135E+02	2.094E+02	1.763E+02
	Mean	**1.873E+02**	4.057E+02	6.900E+02	2.726E+02	4.768E+02	2.933E+02
	Std.	6.640E+01	4.909E+01	1.010E+02	2.945E+01	1.079E+02	1.188E+02
F18	Best	9.054E+02	8.002E+02	9.999E+02	8.858E+02	8.257E+02	9.049E+02
	Mean	9.088E+02	9.139E+02	1.192E+03	9.084E+02	**8.408E+02**	9.099E+02
	Std.	1.927E+00	2.174E+01	7.621E+01	4.454E+00	4.592E+01	2.300E+00
F19	Best	9.058E+02	8.002E+02	9.953E+02	9.066E+02	8.252E+02	9.053E+02
	Mean	9.086E+02	9.085E+02	1.181E+03	9.091E+02	**8.295E+02**	9.099E+02
	Std.	1.779E+00	2.953E+01	8.459E+01	1.092E+00	3.450E+00	2.295E+00
F20	Best	9.060E+02	4.037E+02	9.710E+02	9.065E+02	8.238E+02	9.047E+02
	Mean	9.081E+02	8.924E+02	1.194E+03	9.093E+02	**8.357E+02**	9.099E+02
	Std.	1.627E+00	9.714E+01	9.034E+01	1.225E+00	3.126E+01	2.815E+00
F21	Best	5.000E+02	5.000E+02	1.169E+03	5.000E+02	5.000E+02	5.000E+02
	Mean	**5.000E+02**	**5.000E+02**	1.346E+03	**5.000E+02**	8.737E+02	**5.000E+02**
	Std.	2.313E−13	2.313E−13	7.248E+01	2.313E−13	9.199E+01	1.826E−06
F22	Best	8.707E+02	9.540E+02	1.124E+03	9.046E+02	5.101E+02	8.588E+02
	Mean	9.222E+02	1.062E+03	1.432E+03	9.421E+02	**5.332E+02**	8.894E+02
	Std.	2.742E+01	3.896E+01	9.213E+01	1.650E+01	1.985E+01	1.352E+01
F24	Best	2.000E+02	2.007E+02	1.315E+03	2.000E+02	2.132E+02	2.000E+02
	Mean	**2.000E+02**	8.118E+02	1.423E+03	2.034E+02	2.230E+02	**2.000E+02**
	Std.	0.000E+00	4.408E+02	3.858E+01	6.146E+00	2.482E+01	0.000E+00
F25	Best	9.829E+02	1.274E+03	1.379E+03	9.875E+02	2.134E+02	2.000E+02
	Mean	9.959E+02	1.330E+03	1.441E+03	9.967E+02	**2.208E+02**	2.245E+02
	Std.	7.025E+00	2.077E+01	2.920E+01	4.759E+00	5.544E+00	1.764E+01

significant difference between the results of two algorithms.', and H1 (Alternative hypothesis) was the opposite. When applying a statistical procedure to reject a hypothesis, a level of significance α was used to determine at which level the hypothesis may be rejected. The level of significance, α was 0.05. The *p*-value is the estimated probability of rejecting the null hypothesis (H0) when that hypothesis is true. A small *p*-value indicates strong evidence against the null hypothesis. $R+$ is the sum of ranks for the problems in which the first algorithm outperformed the second, and $R-$ is the sum of ranks for the opposite. T is the smaller of the sums ($T = \min(R+, R-)$) [38]. T and *p*-values have been computed by using Matlab (Release R2010a). In Table 4, '+' indicates the cases in which the null hypothesis was rejected and AAA displayed a statistically superior performance in the problem-based statistical comparison

Table 9
Mean, best and standard deviation values of solutions achieved for CEC05 hybrid functions taken over 30 runs, 50 dimensions.

		AAA	ABC	BA	DE	ACO_R	HS_{POP}
F15	Best	7.032E+00	8.779E−11	4.261E+02	2.717E+01	4.000E+02	2.000E+02
	Mean	3.086E+02	**4.964E+00**	9.253E+02	1.157E+02	7.732E+02	2.700E+02
	Std.	1.395E+02	8.595E+00	2.077E+02	6.206E+01	1.157E+02	7.022E+01
F16	Best	1.756E+02	3.951E+02	6.637E+02	2.586E+02	2.837E+02	2.209E+02
	Mean	3.362E+02	3.998E+02	8.267E+02	2.892E+02	5.189E+02	**2.634E+02**
	Std.	8.228E+01	8.871E−01	1.168E+02	1.708E+01	1.448E+02	5.230E+01
F17	Best	1.442E+02	4.653E+02	7.278E+02	2.783E+02	3.275E+02	2.321E+02
	Mean	**2.735E+02**	5.498E+02	9.639E+02	3.536E+02	5.683E+02	2.820E+02
	Std.	8.468E+01	2.650E+01	1.149E+02	2.360E+01	1.971E+02	4.906E+01
F18	Best	9.213E+02	9.363E+02	1.159E+03	9.253E+02	8.419E+02	9.207E+02
	Mean	9.306E+02	9.563E+02	1.305E+03	9.295E+02	**8.464E+02**	9.267E+02
	Std.	6.554E+00	1.058E+01	6.896E+01	2.910E+00	2.857E+00	7.408E+00
F19	Best	9.193E+02	9.417E+02	1.132E+03	9.258E+02	8.407E+02	9.205E+02
	Mean	9.287E+02	9.574E+02	1.287E+03	9.297E+02	**8.621E+02**	9.256E+02
	Std.	6.390E+00	7.776E+00	7.297E+01	2.530E+00	4.668E+01	3.555E+00
F20	Best	9.189E+02	9.398E+02	1.088E+03	9.234E+02	8.428E+02	9.166E+02
	Mean	9.278E+02	9.574E+02	1.264E+03	9.298E+02	**8.602E+02**	9.254E+02
	Std.	4.843E+00	1.110E+01	9.812E+01	3.026E+00	4.343E+01	3.819E+00
F21	Best	5.000E+02	5.000E+02	1.195E+03	1.014E+03	7.237E+02	5.000E+02
	Mean	5.687E+02	**5.000E+02**	1.392E+03	1.018E+03	7.291E+02	9.559E+02
	Std.	1.783E+02	1.734E−13	8.987E+01	1.842E+00	3.379E+00	1.546E+02
F22	Best	9.241E+02	1.065E+03	1.419E+03	9.118E+02	5.032E+02	8.887E+02
	Mean	9.816E+02	1.170E+03	1.550E+03	9.466E+02	**5.303E+02**	9.118E+02
	Std.	2.702E+01	3.140E+01	6.914E+01	1.511E+01	3.927E+01	1.471E+01
F24	Best	2.000E+02	1.310E+03	1.443E+03	2.598E+02	2.290E+02	2.000E+02
	Mean	8.150E+02	1.355E+03	1.546E+03	8.445E+02	2.869E+02	**2.000E+02**
	Std.	4.373E+02	2.423E+01	3.157E+01	2.485E+02	1.756E+02	1.095E−05
F25	Best	1.259E+03	1.356E+03	1.461E+03	1.246E+03	2.338E+02	2.530E+02
	Mean	1.278E+03	1.403E+03	1.533E+03	1.259E+03	3.884E+02	**2.708E+02**
	Std.	1.201E+01	1.839E+01	3.069E+01	6.973E+00	3.241E+02	1.308E+01

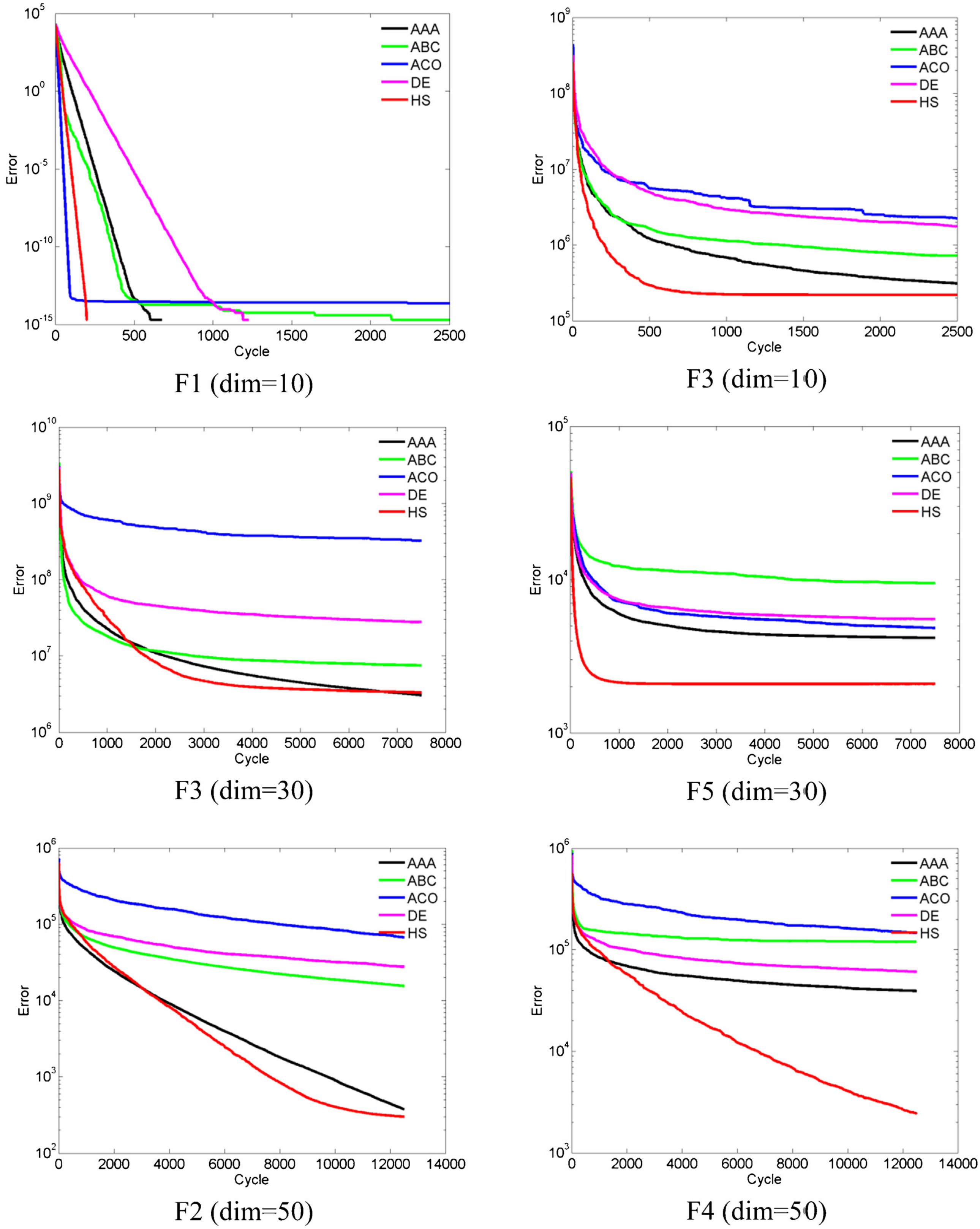

Fig. 5. Convergence curves for uni-modal functions.

tests at the 95% significance level ($\alpha = 0.05$); '–' indicates cases in which the null hypothesis was rejected and AAA displayed an inferior performance; and '=' indicates cases in which there was no statistical difference between the two algorithms success in solving the problems. The last row of Tables 10–12 show the total counts in the (+/=/–) format for the three statistical significance cases (marked with '+', '=' or '–') in the pairwise comparison. Tables 10–12 show that AAA can achieve statistically better results than the comparison algorithms, with a level of significance $\alpha = 0.05$.

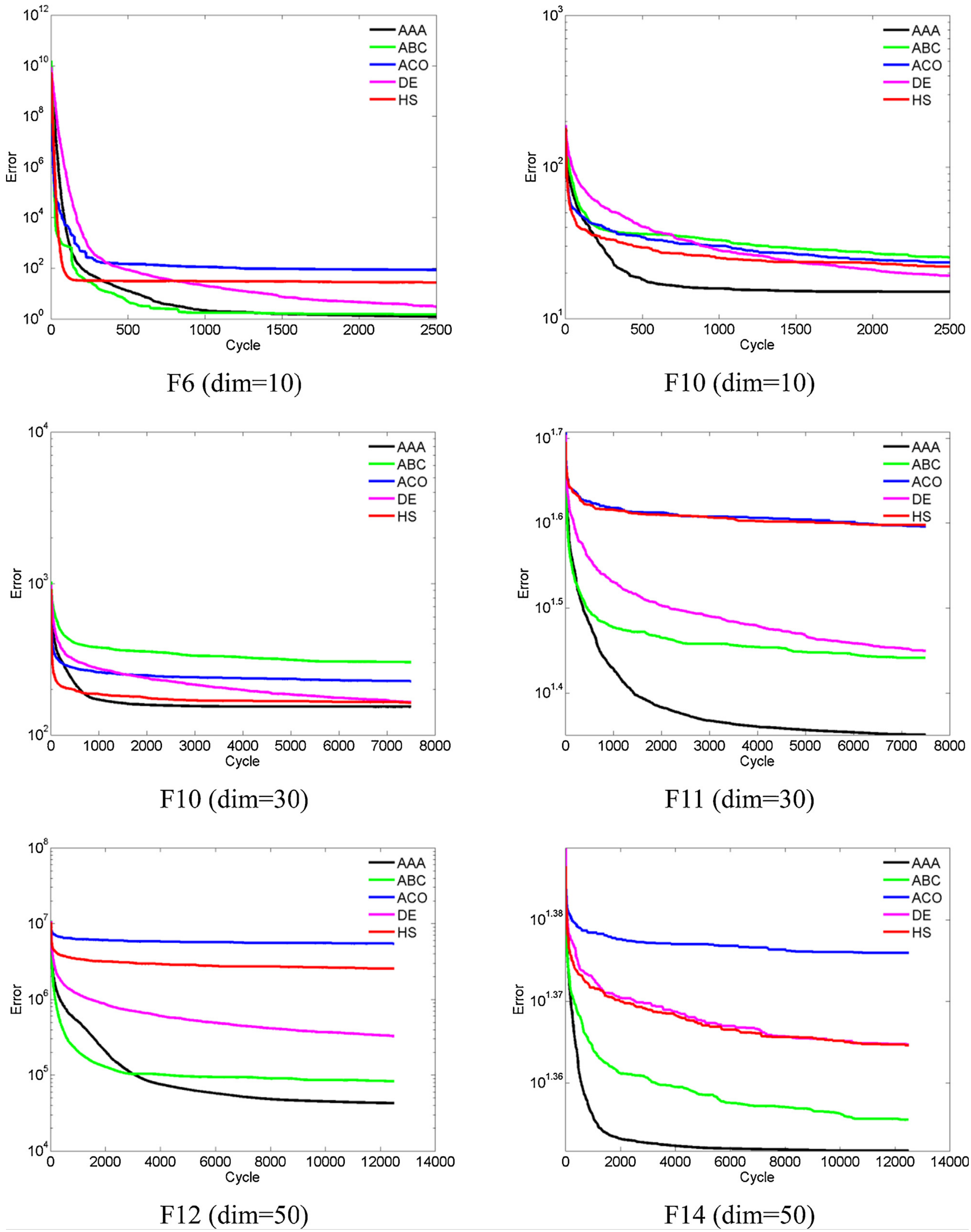

Fig. 6. Convergence curves of the algorithms on multi-modal functions.

4.1.3. Overall performance assessment

In this study, tests were performed on 24 benchmark functions and 3 different dimensions (10, 30 and 50) for each function. Totally 72 comparison results were obtained. Success rates of the algorithms by function groups are indicated in Fig. 8. In the figure, for each of the function group (uni-modal, multi-modal and Hybrid), the number of functions in which the algorithms were more successful than others is presented. When more than one

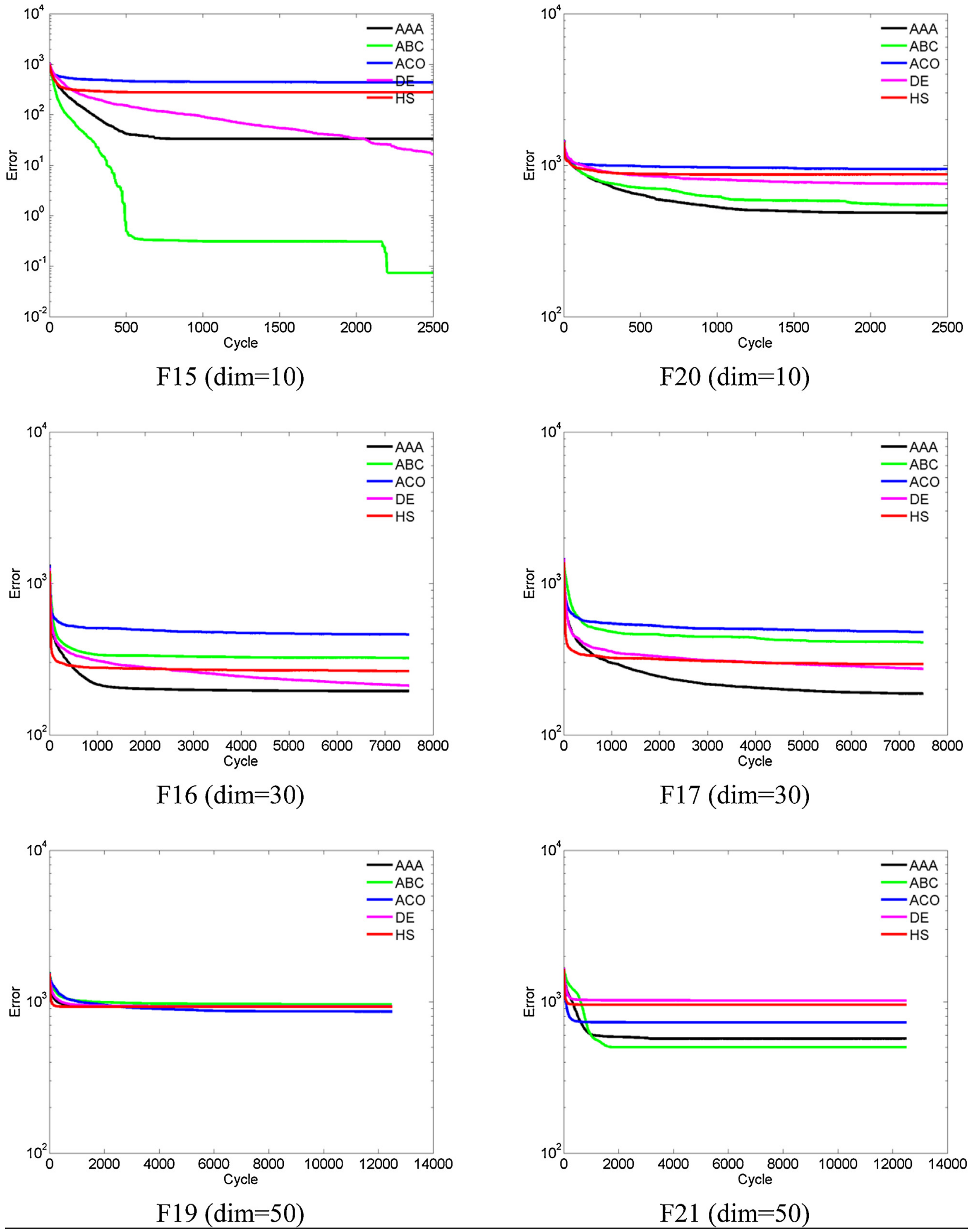

Fig. 7. Convergence curves of the algorithms on hybrid functions.

algorithm gave the same best result, all are counted separately in the histogram; therefore the total value in Fig. 8 was 82 instead of 72. AAA produced more successful results in comparison with other algorithms (Figs. 8 and 9).

The success rate of the function dimensions belonging to the algorithms are compared in Fig. 9. AAA produced successful and balanced results over different dimensions of the functions. These outcomes indicated that the AAA is a consistent algorithm having

Table 10
The results of two-sided Wilcoxon signed-rank test for 10 dimension ($\alpha = 0.05$).

P	AAA vs. ABC			AAA vs. ACO$_R$			AAA vs. BA			AAA vs. DE			AAA vs. HS$_{POP}$		
	p-Value	*T*	*W*	*p*-Value	*T*	*W*	*p*-Value	*T*	*W*	*p*-Value	*T*	*W*	*p*-Value	*T*	*W*
F1	1.00E+00	0	=	1.95E−03	0	+	1.73E−06	0	+	1.00E+00	0	=	1.00E+00	0	=
F2	1.73E−06	0	+	1.73E−06	0	−	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+
F3	8.92E−05	42	+	8.19E−05	41	+	1.92E−06	1	+	1.73E−06	0	+	2.06E−01	171	=
F4	1.73E−06	0	+	1.73E−06	0	−	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+
F5	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+
F6	9.92E−01	232	=	7.19E−01	215	=	1.73E−06	0	+	6.16E−04	66	+	1.92E−06	1	+
F7	1.00E+00	0	=	1.73E−06	0	−	1.57E−02	115	+	1.73E−06	0	−	3.42E−02	12	−
F8	1.73E−06	0	+	6.98E−06	14	+	6.34E−06	13	+	1.73E−06	0	+	2.35E−06	3	+
F9	1.00E+00	0	=	1.72E−06	0	+	1.73E−06	0	+	1.00E+00	0	=	1.00E+00	0	=
F10	7.51E−05	40	+	9.71E−05	43	+	1.73E−06	0	+	4.39E−03	94	+	1.49E−05	22	+
F11	3.18E−06	6	+	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+
F12	8.61E−01	224	=	1.73E−06	0	+	1.73E−06	0	+	9.75E−01	231	=	2.22E−04	53	+
F13	6.98E−06	14	−	1.73E−06	0	+	1.73E−06	0	+	1.59E−03	79	−	9.71E−05	43	+
F14	8.97E−02	150	=	3.18E−06	6	+	3.18E−06	6	+	9.27E−03	106	+	1.80E−05	24	−
F15	1.08E−02	91	−	1.73E−06	0	+	1.73E−06	0	+	1.06E−01	154	=	1.34E−05	21	+
F16	3.88E−04	60	+	1.25E−04	46	+	1.73E−06	0	+	5.29E−04	64	+	1.11E−02	109	+
F17	6.34E−06	13	+	7.51E−05	40	+	1.73E−06	0	+	2.88E−06	5	+	2.58E−03	86	+
F18	7.97E−01	220	=	2.88E−06	5	+	1.73E−06	0	+	8.92E−05	42	+	4.73E−06	10	+
F19	1.65E−01	165	=	1.73E−06	0	+	1.73E−06	0	+	2.84E−05	29	+	1.36E−05	21	+
F20	4.28E−01	194	=	1.73E−06	0	+	1.73E−06	0	+	5.79E−05	37	+	3.18E−06	6	+
F21	1.19E−05	15	−	5.61E−06	0	+	1.73E−06	0	+	4.65E−01	197	=	2.42E−06	0	+
F22	1.20E−01	157	=	6.16E−04	66	+	1.73E−06	0	+	4.73E−06	10	+	2.86E−05	24	+
F24	1.00E+00	0	=	1.73E−06	0	+	1.73E−06	0	+	6.10E−05	0	+	1.25E−01	0	=
F25	9.75E−01	231	=	4.90E−04	63	−	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	−
+/=/−	**9/12/3**			**19/1/4**			**24/0/0**			**17/5/2**			**17/4/3**		

Table 11
The results of two-sided Wilcoxon signed-rank test for 30 dimension ($\alpha = 0.05$).

P	AAA vs. ABC			AAA vs. ACO$_R$			AAA vs. BA			AAA vs. DE			AAA vs. HS$_{POP}$		
	p-Value	*T*	*W*	*p*-Value	*T*	*W*	*p*-Value	*T*	*W*	*p*-Value	*T*	*W*	*p*-Value	*T*	*W*
F1	3.24E−07	0	+	1.04E−05	0	+	1.73E−06	0	+	1.00E+00	0	=	6.80E−08	0	−
F2	1.73E−06	0	+	1.73E−06	0	−	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+
F3	2.13E−06	2	+	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+	3.71E−01	189	=
F4	1.73E−06	0	+	1.73E−06	0	−	1.73E−06	0	+	1.13E−05	19	+	1.73E−06	0	−
F5	1.73E−06	0	+	1.11E−01	155	=	1.73E−06	0	+	8.92E−05	42	+	1.92E−06	1	−
F6	4.65E−01	197	=	4.78E−01	198	=	1.73E−06	0	+	3.06E−04	57	+	2.13E−06	2	+
F7	1.00E+00	0	=	1.72E−06	0	−	1.15E−04	45	+	1.73E−06	0	−	1.00E+00	0	=
F8	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+
F9	3.17E−03	28	−	1.73E−06	0	+	1.73E−06	0	+	1.25E−01	0	=	2.43E−02	105	+
F10	1.73E−06	0	+	2.13E−06	2	+	1.73E−06	0	+	1.31E−01	159	=	2.06E−01	171	=
F11	2.60E−06	4	+	1.73E−06	0	+	1.92E−06	1	+	1.73E−06	0	+	1.73E−06	0	+
F12	1.66E−02	116	+	1.73E−06	0	+	1.73E−06	0	+	6.98E−06	14	+	1.73E−06	0	+
F13	5.22E−06	11	−	1.73E−06	0	+	1.73E−06	0	+	8.19E−05	41	+	1.73E−06	0	+
F14	4.28E−02	134	+	1.73E−06	0	+	1.92E−06	1	+	9.78E−02	152	=	2.13E−06	2	+
F15	3.41E−05	31	−	1.73E−06	0	+	1.73E−06	0	+	4.68E−03	95	−	1.30E−03	50	+
F16	3.18E−06	6	+	1.73E−06	0	+	1.73E−06	0	+	2.43E−02	123	+	8.22E−03	104	+
F17	1.92E−06	1	+	2.13E−06	2	+	1.73E−06	0	+	2.84E−05	29	+	3.88E−04	60	+
F18	3.11E−05	30	+	3.11E−05	30	−	1.73E−06	0	+	7.81E−01	219	=	5.71E−02	140	=
F19	4.20E−04	61	−	1.73E−06	0	−	1.73E−06	0	+	1.53E−01	163	=	1.48E−02	114	+
F20	1.48E−02	114	−	3.11E−05	30	−	1.73E−06	0	+	1.48E−03	78	+	8.22E−03	104	+
F21	1.00E+00	0	=	2.56E−06	0	+	1.73E−06	0	+	1.00E+00	0	=	1.00E+00	0	=
F22	1.73E−06	0	+	1.73E−06	0	−	1.73E−06	0	+	7.73E−03	103	+	2.16E−05	26	−
F24	1.73E−06	0	+	1.73E−06	0	+	1.73E−06	0	+	8.86E−05	0	+	1.00E+00	0	=
F25	1.73E−06	0	+	1.73E−06	0	−	1.73E−06	0	+	5.72E−01	205	=	1.73E−06	0	−
+/=/−	**16/3/5**			**14/2/8**			**24/0/0**			**14/8/2**			**13/6/5**		

balanced search qualifications. The first reason of this is the contribution of adaptation and evolutionary process, the second reason is semi-random selection employed while choosing the source of light in order to avoid local minima, and another reason is balancing of helical movement methods each other.

The global optimum of 72 test results used in this study is zero (0) and the goal is to get this value. The global optimum is obtained when each of the algorithms reaches the stopping criterion, and optimum is the calculated value as indicated in Tables 1–9.

Based on the errors between the expected and calculated values (i.e., real optimum and the model value), performance assessment of an algorithm might also be provided by calculating the values of mean squared error (MSE), root mean square error (RMSE) and mean absolute percentage error (MAPE) [39,40]. In this study, for both AAA and others (ABC, BA, DE, ACO$_R$ and HS$_{POP}$) error calculation and comparison were performed as another overall performance assessment and comparison approach. MSE, RMSE and MAPE values were calculated by the following equations:

$$\mathrm{MSE} = \sqrt{\frac{\sum_{f=1}^{n}(\mathrm{evaluated}\,(f) - \mathrm{prediction}\,(f))^2}{n}} \tag{19}$$

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{f=1}^{n}(\mathrm{evaluated}\,(f) - \mathrm{prediction}\,(f))^2}{n}} \tag{20}$$

Table 12
The results of two-sided Wilcoxon signed-rank test for 50 dimension ($\alpha = 0.05$).

P	AAA vs. ABC			AAA vs. ACO$_R$			AAA vs. BA			AAA vs. DE			AAA vs. HS$_{POP}$		
	p-Value	*T*	*W*	*p*-Value	*T*	*W*	*p*-Value	*T*	*W*	*p*-Value	*T*	*W*	*p*-Value	*T*	*W*
F1	1.40E − 06	0	+	8.01E − 06	16.5	+	1.73E − 06	0	+	2.44E − 04	0	–	2.48E − 05	30	+
F2	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+	2.07E − 02	120	–
F3	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+	1.02E − 01	153	=
F4	1.73E − 06	0	+	2.60E − 06	4	+	1.73E − 06	0	+	4.73E − 06	10	+	1.73E − 06	0	–
F5	1.73E − 06	0	+	4.05E − 01	192	=	1.92E − 06	1	+	1.60E − 04	49	+	1.73E − 06	0	–
F6	2.85E − 02	126	–	1.16E − 01	156	=	1.73E − 06	0	+	4.11E − 03	93	+	2.60E − 06	4	+
F7	1.00E + 00	0	=	1.68E − 06	0	–	6.34E − 06	13	+	1.73E − 06	0	–	1.00E + 00	0	=
F8	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+
F9	1.02E − 01	153	=	1.73E − 06	0	+	1.73E − 06	0	+	5.02E − 05	0	–	1.92E − 06	1	+
F10	1.73E − 06	0	+	2.84E − 05	29	+	1.73E − 06	0	+	5.32E − 03	97	+	1.66E − 02	116	–
F11	3.88E − 06	8	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+
F12	3.41E − 05	31	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+
F13	1.73E − 06	0	–	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+
F14	2.70E − 02	125	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+	1.73E − 06	0	+
F15	1.92E − 06	1	–	2.56E − 06	0	+	1.73E − 06	0	+	1.97E − 05	25	–	1.60E − 01	121	=
F16	9.88E − 05	10	+	1.49E − 05	22	+	1.73E − 06	0	+	2.96E − 03	88	–	2.58E − 03	86	–
F17	1.73E − 06	0	+	6.34E − 06	13	+	1.73E − 06	0	+	2.22E − 04	53	+	5.86E − 01	206	=
F18	1.92E − 06	1	+	1.73E − 06	0	–	1.73E − 06	0	+	5.86E − 01	206	=	1.32E − 02	112	–
F19	1.73E − 06	0	+	3.72E − 05	32	–	1.73E − 06	0	+	3.82E − 01	190	=	6.87E − 02	144	=
F20	1.73E − 06	0	+	3.72E − 05	32	–	1.73E − 06	0	+	8.59E − 02	149	=	5.98E − 02	141	=
F21	1.25E − 01	0	=	1.48E − 02	114	+	1.73E − 06	0	+	1.92E − 06	1	+	1.70E − 05	10	+
F22	1.73E − 06	0	+	1.73E − 06	0	–	1.73E − 06	0	+	4.45E − 05	34	–	1.73E − 06	0	–
F24	1.73E − 06	0	+	6.64E − 04	67	–	1.73E − 06	0	+	9.59E − 01	230	=	5.96E − 05	0	–
F25	1.73E − 06	0	+	1.73E − 06	0	–	1.73E − 06	0	+	2.13E − 06	2	–	1.73E − 06	0	–
+/=/–	**18/3/3**			**15/2/7**			**24/0/0**			**13/4/7**			**9/6/9**		

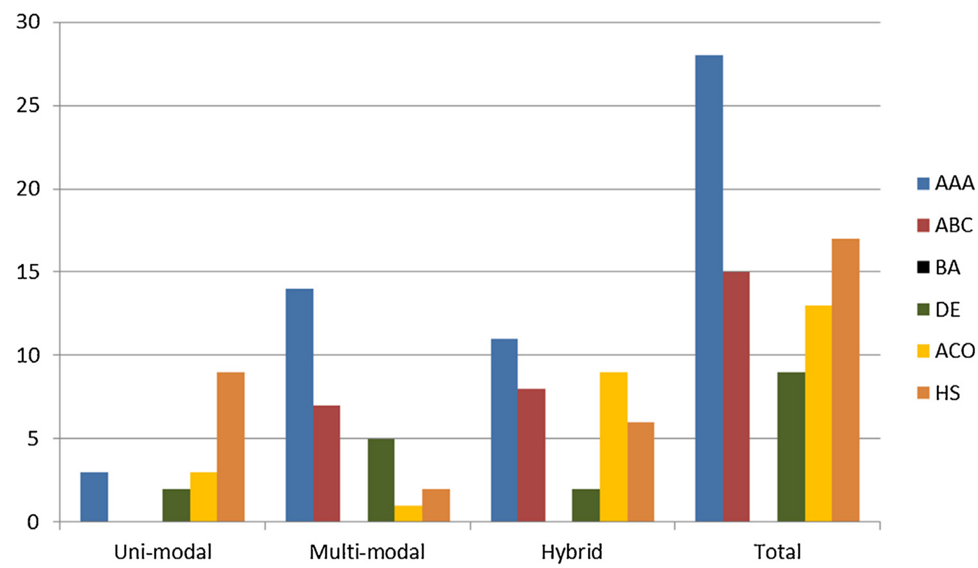

Fig. 8. The number of functions in which the algorithms are successful according to the function groups.

$$\text{MAPE} = \sqrt{\frac{\sum_{f=1}^{n} \left|\text{evaluated}(f) - \text{prediction}(f)/\text{evaluated}(f)\right|}{n}} \tag{21}$$

where evaluated(f) is the calculated optimum value when the algorithm reaches the stopping criterion; prediction(f) is the global optimum value expected from the algorithm; and n is the number of functions.

Calculated error values are given in Table 13. The success of AAA in overall error results supports the histogram results given in Figs. 8 and 9. The lowest error values were obtained by AAA, and the second successful algorithm was HS$_{POP}$.

Table 13
Overall errors of algorithms.

Algorithm	MSE	RMSE	MAPE
AAA	**4.522E + 11**	**6.724E + 05**	**1.157E + 05**
ABC	3.668E + 12	1.915E + 06	3.191E + 05
BA	5.346E + 19	7.312E + 09	1.139E + 09
DE	1.539E + 14	1.240E + 07	1.827E + 06
ACO$_R$	5.226E + 16	2.286E + 08	3.116E + 07
HS$_{POP}$	4.939E + 11	7.028E + 05	1.508E + 05

4.2. Performance of AAA on design optimization problems

The performance of AAA was tested on a popular design optimization problem known as Pressure Vessel Design Optimization Problem [41,42]. The results of the test were compared with SPSO (version 2011) and ABC.

4.2.1. Pressure vessel design problem

The objective in Pressure Vessel Design Problem is to find the most appropriate design to produce a pressure vessel having a volume of 750 ft^3 (21.24 m^3) and working pressure of 3000 psi

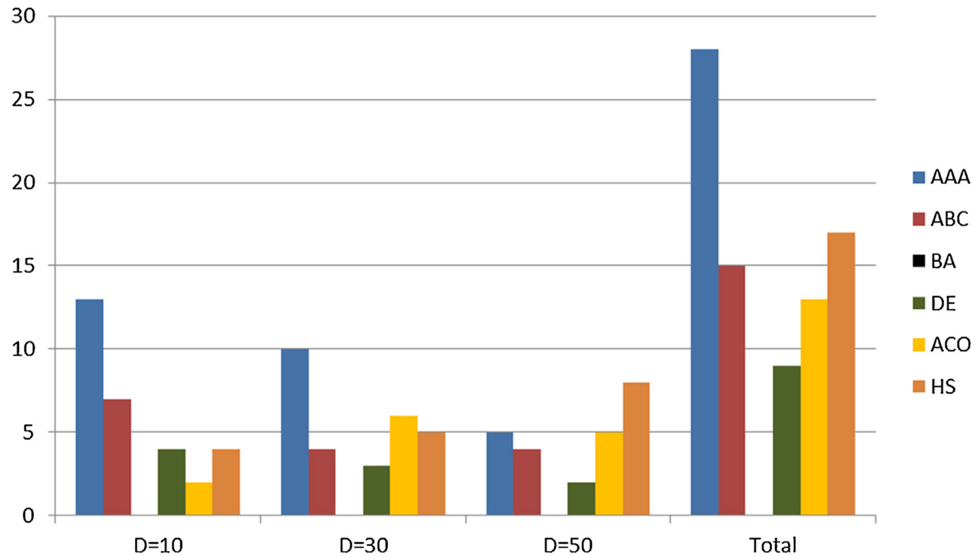

Fig. 9. Number of functions that algorithms were successful according to the dimensions.

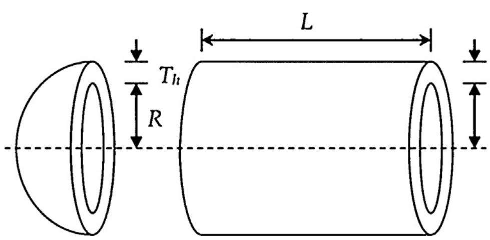

Fig. 10. Schematic diagram of pressure vessel problem: T_s (x_1): thickness of the shell; T_h (x_2): thickness of the head; R (x_3): inner Radius; and L (x_4): length of the cylindrical section of the vessel.

(20.68 MPa) with the lowest cost. As it is displayed in Fig. 10, both ends of the cylindrical pressure vessel are closed with hemispherical heads. The problem has four design variables indicated in Fig. 10. L and R are the variables receiving values continuously, T_h and T_s are separate variables receiving discrete values and they are expected to be the multiples of 0.0625 inch (0.16 cm) [42].

There are also studies conducted on some variations of this problem having different limits, constraints and variables not receiving discrete values [17,43]. In this study tests were performed on the problem whose details were given by Onwubolu and Babu [42], and Clerc [41]. The problem was formulated as follows:

The variables:

$$x_1 \in [1.125, 12.5]\,\textit{granularity}\,0.0625$$

$$x_2 \in [0.625, 12.5]\,\textit{granularity}\,0.0625$$

$$x_3 \in [0.240]$$

$$x_4 \in [0.240]$$

Subject to,

$$g_1(x) = 0.0193x_3 - x_1 \le 0$$

$$g_2(x) = 0.0095x_3 - x_2 \le 0$$

$$g_3(x) = 750.0 \times 1728.0 - \pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 \le 0$$

$$g_4(x) = x_4 - 240.0 \le 0$$

$$g_5(x) = 1.1 - x_1 \le 0$$

$$g_6(x) = 0.6 - x_2 \le 0$$

The objective function:

$$f(x) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1611x_1^2x_4 + 19.84x_1^2x_3$$

Optimum value and parameters:

$$f(x) = 7197.72893$$

$$X = (x_1, x_2, x_3, x_4) = (1.125, 0.625, 58.2901554, 43.6926562)$$

The objective function $f(x)$ represents the production cost of the pressure vessel.

4.2.2. Parameters used on design optimization problems

Common parameters of all algorithms used for this problem were as follows;

- Population size was set to 40 for AAA, SPSO and ABC [35,32]. Number of function evaluation (cost of calculation) was employed as the stopping criterion. Maximum number of function evaluation for three different runs were 10,000, 20,000 and 30,000.
- Initial candidate solutions were randomly started between lower and upper limits of the function by uniform distribution.
- The average of the results of 30 runs was reported as the final result of that function.

Particularly:

- For SPSO algorithm, the inertia weight, $w = 1/(2 \times \log(2))$ and the positive constant parameters were $c1 = c2 = 0.5 + \log(2)$.
- For AAA, as a result of preliminary runs, the parameters were set as follows: the loss of energy, $e = 30\%$; the shear force, $\Delta = 2$; and the adaptation probability constant, $A_p = 1$.

4.2.3. Comparison results on design optimization problems

The parameter values by which the algorithms gave the best results at the end of 30 runs with 30,000 function evaluations for Pressure Vessel Design optimization problem are given in Table 14. The values of y_1 and y_2 correspond to the discrete values of x_1 (T_s) and x_2 (T_h) parameters used in objective function based on

Table 14
Comparison of the best solutions (minimum cost) for the pressure vessel problem.

	AAA	SPSO	ABC
$x_1(T_s)$	1.13905470	1.13168647	1.12500000
$x_2(T_h)$	0.625000000	0.629557446	0.62500000
x_3 (R)	58.2901554	57.9589752	58.1960902
x_4 (L)	43.6926565	45.5276803	44.2348349
y_1	1.125	1.125	1.125
y_2	0.625	0.625	0.625
g_1	−7.79999842E−10	−6.39177864E−03	−1.81545914E−03
g_2	−6.89119175E−02	−7.20713766E−02	−6.98092995E−02
g_3	−4.15554037E−04	−2.07786023E+01	−2.54823623E+02
g_4	−1.96307344E+02	−1.94472320E+02	−1.95765165E+02
g_5	−2.50000000E−02	−2.50000000E−02	−2.50000000E−02
g_6	−2.50000000E−02	−2.50000000E−02	−2.50000000E−02
***f*(*x*)**	***7197.72893***	*7218.30817*	*7204.57448*

Table 15
Comparison of optimizers for the pressure vessel design problem (NFEs: number of function evaluations).

NFEs	Algorithm	Fit. mean	Fit. best	Fit. worst	Fit. std.	Time avg.	Time std.
10,000	AAA	**7199.64315**	**7197.81176**	**7204.81608**	**1.433E+00**	**3.642E−01**	1.372E−02
	SPSO	7787.81825	7307.80230	8703.37492	3.377E+02	2.495E+00	1.921E−01
	ABC	7467.07005	7216.87872	7903.67565	1.757E+02	1.669E+00	5.725E−02
20,000	AAA	**7197.75025**	**7197.73086**	**7197.88922**	**3.305E−02**	**7.245E−01**	1.939E−02
	SPSO	7755.77219	7215.65815	8903.28015	3.551E+02	4.404E+00	2.028E−01
	ABC	7700.73753	7294.37680	7903.67564	2.400E+02	3.405E+00	8.433E−02
30,000	AAA	**7197.72909**	**7197.72893**	**7197.73117**	**4.153E−04**	**1.119E+00**	3.959E−02
	SPSO	7709.40529	7218.30817	8470.96724	3.073E+02	7.230E+00	4.242E−01
	ABC	7463.02094	7204.57448	7903.67564	2.315E+02	5.134E+00	9.521E−02

the penalty method. The introduced algorithm, AAA produced the values providing the lowest cost among these three methods (Table 14).

In order to analyze the performance of the implemented methods better, the stopping criterion was considered as 10,000, 20,000 and 30,000 number of function evaluations. In addition to the best results of the algorithms, mean value, the worst value, standard deviation, average run time and standard deviation of the average run time values are presented in Table 15 where the best mean cost, the best cost and the best average time are written in bold. For all of the functional calculation numbers, AAA gave the best results. In addition to providing results close to global optimum, AAA algorithm is a consistent and fast algorithm as it provides lower standard deviation and shorter calculation duration. All results were obtained on a PC having Intel core i7-2670QM 2.20 GHz cpu, 8GB ram and Windows 7 operating system.

5. Discussion and conclusions

This paper introduced the basic structure of the developed artificial algae algorithm (AAA) which is a novel and a highly effective population-based evolutionary, metaheuristic optimization algorithm. AAA is inspired by the living behaviors of microalgae in nature. Basically, an artificial algae model has been created by using the algal tendency to be close to light, algal helical movement and features including their reproduction and adaptation to the environment. The artificial algal colony was defined as an individual looking for better solutions within the algorithm continuously.

AAA has three control parameters (energy loss, adaptation parameter and shear force). Energy loss parameter determines the number of new candidate solutions of algal colonies produced at each iteration. Each algal colony can produce new candidate solutions in direct proportion to its energy (the success achieved in the previous iteration). Loss of energy that occurs when each candidate solution produced is determined by the 'Energy Loss' parameter. The smaller the energy loss parameter the higher the local search capability of the algorithm. However, this may lead to get stuck in a local minimum. On the contrary, the higher the determined parameter, this time the global search ability so the diversity of the algorithm increases, but this may slow down of the rate of convergence of the algorithm. In the subsequent versions of this model, to increase the success, it is planned to use an adaptive energy loss parameter. Adaptation parameter determines the speed of the process in which algal colony, which could not find good solutions, tries to resemble itself to the biggest algal colony in the environment. If this parameter is determined as big the rate of convergence increases, the computation time and computational cost reduces. The shear force parameter determines the boundaries of the search space within which each algal colony generate a new candidate solution which is inversely proportional to the size of the colony.

AAA has two operators: adaptation and evolutionary process. In adaptation process, in each iteration, an insufficiently grown algal colony tries to resemble itself to the biggest algal colony in the environment. This process increases the rate of convergence. In evolutionary process, single algal cell of the smallest algal colony dyes and it is replaced by the replicated algal cell of the biggest algal colony this process achieves fine tuning to find the global optimum.

Helical movement is applied to produce a new candidate solution in AAA. In AAA, a step size which is (in terms of size of current solution) weighted difference of the current solution and a solution (selected with tournament selection method) is applied to only three algal cells of the current solution (algal colony) to produce a candidate solution. AAA employs a greedy selection process between the candidate and the current solutions. Thus, helical movement has three capabilities: first, it increases diversity by the tournament method, second, it increases sensitivity to provide changing in certain sizes, and third, it increases the convergence speed of search into a local minimum by greedy selection. These capabilities allow the algorithm to be stable and flexible.

The success of the model was proved with two different experimental phases: benchmark functions and a design optimization problem. In the first phase 24 benchmark functions were tested on 10, 30 and 50 dimensions. Other algorithms (ABC, ACO_R, BA, DE and HS_{POP}) that AAA was compared to, could not keep on their success in uni-modal, multi-modal and hybrid functions, whereas AAA was consistently successful. All test results indicated that in addition to its success, AAA is a balanced and consistent algorithm. The success of AAA was compared with the other algorithms (ABC, ACO_R, BA, DE and HS_{POP}) using the Wilcoxon signed-rank test widely used in the literature. The results of Wilcoxon sign-rank test show that AAA was statistically more successful than the comparison algorithms in solving numeric optimization problem. The second testing phase was performed on a well-known constrained design optimization problem: Pressure vessel design optimization. In these tests AAA was compared with other current algorithms (ABC and SPSO2011) obtaining successful results in the literature, and AAA produced good results in terms of mean, best, standard deviation and run time.

The results obtained from this study indicate that performance of the proposed algorithm AAA is more successful than compared algorithms. However, as a disadvantage, the number of parameters of AAA is higher than some algorithms (e.g. ABC). If inappropriate values are given to the parameters, this may lead to get stuck in a local minimum and premature convergence problem. AAA will be further developed in order to provide faster convergence and better optimum solutions with parameter tuning and hybrid studies. Furthermore, the algorithm can be adapted to developments provided on other methods to AAA will be performed. This was the paper that introduced the developed main AAA model.

Acknowledgement

This work was financially supported by the Coordinatorship of Scientific Research Projects of Selcuk University (P.N.: 13101019).

References

[1] R.A. Sarker, C.S. Newton, Optimization Modeling: A Practical Introduction, Taylor & Francis: CRC Press, Boca Raton, FL, 2008, ISBN 9781420043105.
[2] X.S. Yang, Z. Cui, Swarm Intelligence and Bio-inspired computation: Theory and Application, Elsevier Inc., Newness, London, 2013, ISBN 978-0-12-405163-8.
[3] X. Yang, Nature-Inspired Metaheuristic Algorithms Second Edition, Luniver Press, United Kingdom, 2010, pp. 1–9.
[4] J. Brownlee, Clever Algorithms: Nature-Inspired Programming Recipes, LuLu, Melbourne, VIC, 2011.
[5] S. Gao, Bio-Inspired Computational Algorithms and Their Applications, InTech, Rijeka, Croatia, 2012, ISBN 978-953-51-0214-4.
[6] S. Binitha, S. Siva Sathya, A survey of bio inspired optimization algorithms, Int. J. Soft Comput. Eng. (IJSCE) 2 (2) (2012), ISSN: 2231-2307.
[7] J. Holland, Adaptation in Natural and Artificial systems, University of Michigan Press, Ann Anbor, MI, 1975.
[8] R. Storn, K. Price, Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces, J. Glob. Optim. 11 (1997) 341–359.
[9] Z.W. Geem, J.H. Kim, G.V. Loganathan, A new heuristic optimization: Harmony search, Simulation 76 (2001) 60–68.
[10] M. Dorigo, Optimization, Learning and Natural Algorithms, Politecnico di Milano, Italy, 1992 (PhD thesis).
[11] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proc. of the IEEE Int. Conf. on Neural Networks, Piscataway, NJ, 1995, pp. 1942–1948.
[12] D. Karaboga, An idea based on honey bee swarm for numerical optimization, in: Technical Report, Erciyes University, Kayseri, Turkey, 2005.
[13] D.T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, M. Zaidi, The bees algorithm—a novel tool for complex optimization problems, in: Proceedings of Innovative Production Machines and Systems Virtual Conference, IPROMS, 2006, pp. 451–461.
[14] X.S. Yang, Firefly algorithms for multimodal optimization, in: O. Watanabe, T. Zeugmann (Eds.), Proc. 5th Symposium on Stochastic Algorithms, Foundations and Applications, SAGA 2009, 5792, 2009, pp. 169–178 (Lecture Notes in Computer Science).
[15] X.S. Yang, S. Deb, Cuckoo search via Lévy flights, in: World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), IEEE Publications, USA, 2009, pp. 210–214.
[16] R. Rajabioun, Cuckoo Optimization Algorithm, Appl. Soft Comput. 11 (8) (2011) 5508–5518.
[17] X.S. Yang, Flower pollination algorithm for global optimization, in: Unconventional Computation and Natural Computation 2012, Lect. Notes Comput. Sci. 7445 (2012) 240–249.
[18] K.M. Passino, Biomimicry of bacterial foraging for distributed optimization and control, IEEE Control Syst. Mag. 5 (2002) 52–67.
[19] J. Dang, A. Brabazon, M. O'Neill, D. Edelman, Option model calibration using a bacterial foraging optimization algorithm, Lect. Notes Comput. Sci. 4974 (2008) 113–122.
[20] L.F. Graham, L.W. Wilcox, Algae, Prentice-Hall, USA, 2000.
[21] S.J. Arceivala, J.S.S. Lokshminarayana, S.R. Alegersamy, C.A. Sastry, Mechanism of treatment and role of algae and bacteria, in: Waste Stabilisation ponds Design, Construction & Operation in India, Central Public Health Engineering Research Institute, Nagpur, India, 1970.
[22] H.O. Buhr, S.B. Miller, A dynamic model of the high-rate algal–bacterial wastewater treatment pond, Water Res. 17 (1983) 29–37.
[23] R.A. Lewin, Physiology and Biochemistry of Algae, third ed., Academic Press, New York, NY, 1970.
[24] F. Chen, H. Chen, X. Gong, Mixotrophic and heterotrophic growth of *Haematococcus lacustris* and rheological behaviour of the cell suspensions, Bioresour. Technol. 62 (1997) 19–24.
[25] I. Cohen, A.F. Post, The heterotrophic connection in a photoautotrophic *Chlorella vulgaris* dominant in wastewater oxidation ponds, Water Sci. Technol. 27 (7–8) (1993) 151–155.
[26] M. Kobayashi, T. Kakizono, K. Yamaguchi, N. Nishio, S. Nagai, Growth and astaxanthin formation of *Haematococcus pluvialis* in heterotrophic and mixotrophic conditions, J. Ferment. Bioeng. 74 (1) (1992) 17–20.
[27] T.P. Curtis, D.D. Mara, N.G.H. Dixo, S.A. Silva, Light penetration in waste stabilization ponds, Water Res. 28 (5) (1994) 1031–1038.
[28] E.M. Grima, F.G. Camacho, J.A.S. Pérez, J.M.F. Sevilla, F.G.A. Fernandez, A.C. Gomez, A mathematical model of microalgal growth in light-limited chemostat culture, J. Chem. Technol. Biotechnol. 61 (1994) 167–173.
[29] R.J. Geider, B.A. Osborne, Algal Photosynthesis—The Measurement of Algal Gas Exchange, Chapman and Hall Inc., New York, NY, 1992.
[30] G.E. Fogg, Algal Cultures and Phytoplankton Ecology, The Univ. of Wisconsin Press, Wisconsin, 1975.
[31] B. Akay, D. Karaboga, Parameter tuning for the artificial bee colony algorithm, in: Proceedings of 1st International Conference on Computational Collective Intelligence, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 608–619.
[32] M. El-Abd, Performance assessment of foraging algorithms vs. evolutionary algorithms, Inf. Sci. 182 (2012) 243–263.
[33] K. Socha, M. Dorigo, Ant colony optimization for continuous domains, Eur. J. Oper. Res. 185 (3) (2008) 1155–1173.
[34] A. Mukhopadhyay, A. Roy, S. Das, S. Das, A. Abraham, Population-variance and explorative power of harmony search: an analysis, in: Second National Conference on Mathematical Techniques: Emerging Paradigms for Electronics and IT Industries (MATEIT 2008), New Delhi, India, 2008.
[35] M. Clerc, Standard Particle Swarm Optimization from 2006 to 2011, 2014, ⟨http://clerc.maurice.free.fr/pso/⟩ [Online].
[36] M.G.H. Omran, M. Clerc, Particle Swarm Central (PSC), Standard PSO 2011 code, 2014, ⟨http://www.particleswarm.info⟩ [Online].
[37] P.N. Suganthan, N. Hansen, J.J. Liang, K. Deb, Y.P. Chen, A. Auger, S. Tiwari, Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization, Nanyang Technological University (NTU), Singapore, in: Technical Report, 2005, Available ⟨http://www.lri.fr/~hansen/Tech-Report-May-30-05.pdf⟩ [Online].
[38] J. Derrac, S. Garcia, D. Molina, F. Herrera, A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, Swarm Evol. Comput. 1 (2011) 3–18.
[39] I.B. Aydilek, A. Arslan, A hybrid method for imputation of missing values using optimized fuzzy c-means with support vector regression and a genetic algorithm, Inf. Sci. 233 (2013) 25–35.
[40] M.Y. Chen, A hybrid ANFIS model for business failure prediction utilizing particle swarm optimization and subtractive clustering, Inf. Sci. 220 (2013) 180–195.
[41] M. Clerc, A Method to improve Standard PSO, Open access archive HAL, France, 2009, MC2009-03-13, (hal-00394945).
[42] G.C. Onwubolu, B.V. Babu, New Optimization Techniques in Engineering, Springer, Berlin, Germany, 2004, ISBN 3-540-20167-X.
[43] H. Eskandar, A. Sadollah, A. Bahreininejad, M. Hamdi, Water cycle algorithm—a novel metaheuristic optimization method for solving constrained engineering optimization problems, Comput. Struct. 110–111 (2012) 151–166.