Quantum Henry Gas Solubility Optimization Algorithm for Global Optimization

Davood Mohammadi[a], Mohamed Abd Elaziz[b,*], Reza Moghdani[a], Emrah Demir[c], and Seyedali Mirjalili[d]

[a]CIIORG, Persian Gulf University, Bushehr, 75168, Iran

[b]Department of Mathematics, Faculty of Science, Zagazig University, Zagazig, Egypt

[c]PARC Institute of Manufacturing, Logistics and Inventory, Cardiff Business School, Cardiff University, Cardiff, United Kingdom

[d]Institute for Integrated and Intelligent Systems, Griffith University, Nathan, QLD 4111, Australia

Corresponding Author: Mohamed Abd Elaziz

dawoodmohammadi@mehr.pgu.ac.ir, abd_el_aziz_m@yahoo.com, Reza.moghdani@gmail.com, Demire@cardiff.ac.uk, ali.mirjalili@gmail.com

Abstract

This paper proposes an improvement on the recently introduced Henry Gas Solubility Optimization (HGSO) metaheuristic algorithm that simulates the Henry's gas law (i.e., the concentration of a gas sample in liquid solvent is proportional to the concentration of the sample in gas phase). As an improvement we apply quantum theory instead of standard procedure used in HSGO algorithm for updating solutions. The proposed algorithm is named as Quantum HGSO (QHGSO) algorithm in this paper. The suggested changes enhance the ability of HGSO to create a counterbalance between the exploitation and exploration for a better investigation of the solution space. For evaluating the capability of finding optimal solution of our proposed algorithm, a collection of forty-seven global optimization functions is solved. Moreover, three well-known engineering problems are studied to show the performance of the QHGSO algorithm in constrained optimization problems. Comparative results with other well-known metaheuristic algorithms have shown that the QHGSO algorithm outperforms others with higher computational performance.

***Keywords**:* Henry Gas Solubility Optimization Algorithm; Quantum Theory; Metaheuristics; Optimization; Physics-Based Algorithms.

1. Introduction

Global optimization problems have received a great deal of attention in recent years since they have been applied to different fields, including image processing [1], machine learning [2], Internet of Things (IoT) [3], and Transportation [4]. However, the complexity of solving these optimization problems has been brought into focus by not only researchers in the field of artificial intelligence, but also in other fields of mathematical sciences. As a result of this, a variety of methods and algorithms has been introduced to overcome these intricate problems.

Nature-inspired algorithms in the literature are generally created based on natural phenomena, such as biology and physics and are classified into four main groups. The first group (i.e., stochastic algorithms) utilizes randomness to explore the search space, including Stochastic Hill Climbing (SHC) [5], Greedy Randomized Adaptive Search Procedure (GRASP) [6], Local Search (LS) [7], Adaptive Random Search

[8], Tabu Search [9], Iterated Local Search (ILS) [10], and Variable Neighborhood Search (VNS) [11]. The second group is called as population-based algorithms. The most well-known metaheuristic in this group is Genetic Algorithm [12]. These algorithms include Evolution Strategies [13], Evolutionary Programming [14], Adaptive Differential Evolution [15], Differential Evolution [16], and Gene Expression Programming [17]. The third group is named as Physics-Based algorithm, which is inspired by a variety of physical systems that mix both local and global search methods. These algorithms include Water Wave Optimization (WWO) [18], Ions Motion Algorithm [19], Mine Blast Algorithm (MBA) [20], Bacteria Chemotaxis [21] and Teaching–Learning-Based Optimization [22]. The last group algorithms imitate the social and individual behavior of natural swarms. These algorithms include Particle Swarm Optimization (PSO) [23], Grey Wolf Optimizer (GWO) [24], Volleyball Premier League Algorithm [25], Dolphin Echolocation [26], Migrating Birds Optimization [27], Elephant Herding Optimization [28], Ant Colony Optimization [29], Cuckoo Search Algorithm [30], Bees Algorithm [31], Spider Optimization Algorithm (SOA) [32], Shuffled Frog Leaping [33], Cat Swarm Optimization [34], Firefly Algorithm [35], and Artificial Bee Colony [36].

The proposal of Richard Feynman [37] regards to quantum computing systems based on quantum mechanics in 1982 was an inspiration for physics-inspired optimization algorithms. This effort lighted a way for quantum computing and it was an inspiration for Narayanan and Moore to propose their Quantum-Inspired Genetic Algorithm (GA) in 1995 [38]. Later, in 2002, Han [39] improved the Evolutionary Algorithm (EA) using quantum theory based on [37] and [38].

As a promising alternative in the area of physics-based methodologies, the Henry Gas Solubility Optimization algorithm (HGSO) appears to be an useful tool for solving optimization problems [40]. This algorithm inspired by the rule of Henry who defined a law to describe solubility of gas in a fluid. More specifically, William Henry introduced a law for the utmost amount of solute which can be solved at a certain amount of solvent at a defined temperature or pressure in the late 1800s. This definition is known as solubility [40]. An example of the improved HGSO algorithm for solving the DNA motif discovery problem presented by Hashim et.al [41]. This problem is vitally important in terms of identifying the transcription factor binding sites which can assist in learning the mechanisms for regulation of gene expression. To solve the same problem, the HGSO is modified by adding a new phase which includes the main specifications of the motifs in DNA sequences.

It is a theoretical foundation of modern physics which tends to describe the essence and behavior of substance and energy at the levels of atomic and subatomic. On the other hand, modern physics shed a new light to describe the behavior of matter and energy at the layer of the atomic and subatomic levels by introducing quantum physics which also called as quantum mechanics. This theory presented by physicist Max Planck in 1900. Due to the aforementioned points, the wave function which borrowed from quantum theory is used to define the status of every gas instead of what is defined in the classical HGSO.

The intention of the proposed algorithm is to improve the performance of the original HGSO algorithm. This objective is done using a new scheme borrowed by the quantum theory for updating the position of each solution. The contributions of this study are the followings.

a) Improving the Henry Gas Solubility Optimization algorithm using Chaotic coefficient in quantum behavior instead of random numbers;
b) Evaluating the performance of the proposed QHGSO using a set of experiments (i.e., the forty-seven optimization test functions and three engineering problems);
c) Comparing the results of the proposed QHGSO with other well-known global optimization methods.

The rest of the paper is organized as follows. In section 2, a brief review on the related metaheuristics algorithms and its applications is presented. Section 3 presents a glimpse at the mechanism of the HGSO algorithm. Section 4 is dedicated to introducing the mechanism of the proposed algorithm. Section 5 presents the experimental analysis of the proposed algorithm on studied test functions and engineering problems. Finally, conclusions and future research directions are presented in section 6.

2. Literature Review

This section provides a brief literature review on the classification of metaheuristic algorithms.

Metaheuristic algorithms can be classified into four main groups as shown in Figure 1. Although there are numerous metaheuristic algorithms available, we only review the physic-based algorithms since our proposed algorithm is a variant in this category. The class of physic-based algorithms represent the methods which are inspired by laws and phenomena in the field of physic or chemistry and classified into five subdomains [42] as follows. These include quantum theory, electrostatics, electromagnetism, Newton's gravitational law, and the laws of motion. The algorithms related to the group of quantum theory are inspired by new laws and formulations on subatomic level that are entirely different from classical laws and formulas that were unable to explain the subatomic behavior. This theory is benefited from a probability distribution function to define the position of each particle instead of velocity and acceleration in classic physics.

Some of well-known algorithms in the quantum theory group are presented in the followings. Quantum-Inspired Bacterial Swarming Optimization (QBSO) [43] modifies bacterial foraging optimization algorithm using quantum bit for defining probabilistic solution representation for each bacteria position. This algorithm is introduced for solving discrete optimization problems as a new approach in evolutionary algorithms and called as Quantum-Inspired Evolutionary Algorithm (QEA) [39]. This scheme is benefited from a string of Q-bits to define the individual as a probabilistic solution representation for solving combinatorial optimization problems. Quantum-Inspired Genetic Algorithm (QGA) presented by Narayanan et. al. [38] is used the same approach to define quantum bits instead of binary ones in solution representation for solving traveling salesman problem. For enhancing the performance of immune clonal algorithm, a Quantum-Inspired Immune Clonal Algorithm (QICA) is presented by Jiao et. al. [44]. In this algorithm the antibody is proliferated and separated into a set of subgroups. The antibodies in a subgroup are indicated by multistate gene quantum bits. Moreover, for updating the antibody, the general quantum rotation gate strategy and the dynamic adjusting angle mechanism are implemented to speed up the convergence. Continuous quantum ant colony optimization (CQACO) is presented by Li et. Al. [45] to overcome the drawback of ACO algorithm which is for discrete optimization problems. ACO algorithm is benefited from quantum bit to determine the position of each ant to expands the algorithm for continuous optimization problems and boosts its convergence rate. Due to the quantum-behaved particle swarm optimization a novel parameter control method is introduced by Sun et al. [46] to boost the performance of the quantum-behaved particle swarm optimization (QPSO). An Improved Quantum Evolutionary Algorithm (IQEA) is introduced by Zhang et. al.[47]. In this study a novel scheme of adaptive computing rotation angle of quantum rotation gate is planned on the foundation of the probability domain ratio of the related positions and applied for solving 0/1 knapsack problems.

The Particle Swarm Optimization is empowered by quantum bits to update the quantum angles spontaneously in a paper titled as Quantum Swarm Evolutionary Algorithm (QSE) [48]. After converting a test suite reduction problem to the standard optimization problem, a novel scheme to evolutionary algorithm using quantum bit in comparison with its original bit is presented in a paper titled as Reduced Quantum Genetic Algorithm (RQGA) [49]. To tackle some drawbacks of the quantum-inspired evolutionary algorithms and describing how the hitchhiking problem can slow down to find optimal solution and trapped in premature convergence, a Versatile Quantum-inspired Evolutionary Algorithm (VQEA) is introduced in Platel et.al [50]. In this algorithm, the attractor agents change their positions among the population via the search space and relocated at every generation without considering their fitness. Electrostatics-Based Algorithms and Electromagnetism-Based Algorithms are inspired by classical physics in the field of electrical phenomena consists of Charged System Search (CSS) [51] and Electromagnetism (EM) [52], respectively.

The algorithms related to Newton's gravitational law which is another group based on classic physics and inspired by gravity law including Big Bang-Big Crunch (BBBC) [53], Galaxy-Based Search Algorithm (GBSA) [54], Gravitational Interaction Optimization (GIO) [55], Gravitational Search Algorithm (GSA) [56], Artificial Physics Optimization (APO) [57], Central Force Optimization (CFO) [58], Black Hole (BH) algorithm [59], Ray Optimization (RO) algorithm [60], Small-World Optimization Algorithm (SWOA) [61]. The algorithms belong to the last group inspired by the phenomena in the field of chemistry are including Artificial Chemical Reaction Optimization Algorithm (ACROA) [62], Quantum Evolutionary Algorithm Hybridized with Enhanced Colliding Bodies (QEECB) [63], Plasma Generation Optimization (PGO) [64], Simulated Annealing (SA) [65], Gases Brownian Motion Optimization (GBMO) [66] and Henry Gas Solubility Optimization algorithm (HGSO) [40].

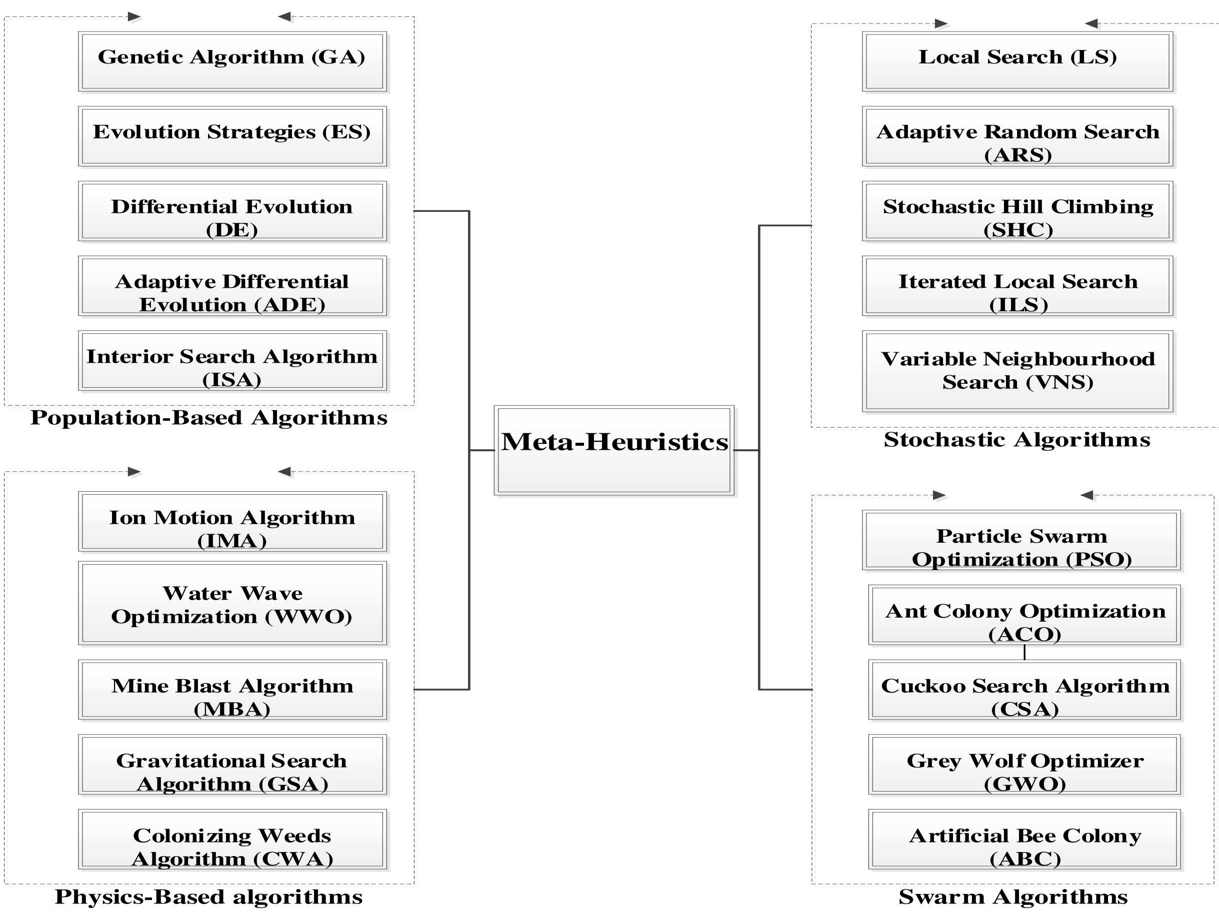

Figure 1: Four main groups of metaheuristic algorithms

3. Henry Gas Solubility Optimization Algorithm

Henry Gas Solubility Optimization algorithm is a newfound physics-inspired optimization method presented by Hashim et.al [40]. This new method inspired by Henry's gas law which describes the rules for solubility gas in liquids. Henry's law has ability to specify the solubility of low-solubility gases in fluids. Furthermore, two main factors which have direct effect on solubility are temperature and pressure. At temperature with the higher degrees, the solubility of solids increases, by contrast, gases have less chance to be soluble [67]. In terms of pressure, the capability of gas for solubility in liquids raises by increasing the amount of pressure [40]. Using these two important features, HGSO algorithm consists of eight steps as follows. The first step represents the initialization procedure which generates the number of gases (population), their positions and the value of Henry's constant for each group j ($H_j(t)$), partial pressure $P_{i,j}$ of gas *i* at each group *j*. The mathematical equation of this stage is as follows.

$$X_i(t+1) = X_{min} + r \times (X_{max} - X_{min}), \tag{1}$$

where X_i represents the location of the *ith* gas in population *N*, *r* defined as a chaotic number between 0 and 1, and X_{min}, X_{max} are the bounds of the problem and (t) is the iteration.

$$H_j(t) = l_1 \times rand\ (0,1), P_{i,j} = l_2 \times rand\ (0,1)\ , C_j = l_3 \times rand\ (0,1), \tag{2}$$

where l_1 to l_3 are constant numbers with value of 5E−02, 100 and 1E−02), respectively.

In the second step, called clustering, the population of gases is classified into based on the number of gas kinds. All gases at every group have the same H_j.

In the third step, called evaluation, to estimate the best gas in each cluster j which obtain the highest equilibrium position in comparison with others in its kind. To find the best gas in whole swarm, ranking is used.

An equation for updating Henry's coefficient is used in step four as follows:

$$H_j(t+1) = H_j(t) \times e^{(-C_j(\frac{1}{T(t)} - 1/T^{\theta}))}, \quad T(t) = e^{(-t/iter)}, \tag{3}$$

where H_j illustrates the coefficient of Henry gas rule in each group j, *T* represents the temperature, T^{θ} defined as a fixed quantity which its value is 298. Moreover, the *iter* represents the total number of iterations.

The solubility update equation as follows in which $S_{i,j}$ illustrates solvability of gas *i* at each group *j*, $P_{i,j}$ represents the parochial pressure on gas *i* at each group j and K used as a fixed value.

$$S_{i,j}(t) = K \times H_j(t+1) \times P_{i,j}(t). \tag{4}$$

Step six presents an equation for updating position of each gas in Eq.(5-6):

$$X_{i,j}(t+1) = X_{i,j}(t) + F \times r \times \gamma \times \left(X_{i,best}(t) - X_{i,j}(t)\right) + F \times r \times \alpha \times \left(S_{i,j}(t) \times X_{i,best}(t) - X_{i,j}(t)\right) \tag{5}$$

$$\gamma = \beta \times exp\left(-\frac{F_{best}(t) + \varepsilon}{F_{i,j}(t) + \varepsilon}\right), \qquad \varepsilon = 0.05, \tag{6}$$

where $X_{i,j}$ represents the status for each gas i at every group j, r and t a random value between 0 and 1 and the iteration time, $X_{i,best}$, shows the best gas i at each group j, while X_{best} presents the best gas among whole population. Moreover, γ represents the capability of each gas j in every group i interacted by the gases in its group, α shows the impact of the rest of the gases on gas i in group j and takes the value of 1 and a fixed number will be assigned to β. $F_{i,j}$ represents the fitness for each gas i at each group j, on the other hand F_{best} illustrates the fitness of the best gas in the whole population. An equation to avoiding local optimum situations as follows in step seven:

$$N_w = N * (rand((C_2 - C_1) + C_1), C_1 \ ; \ C_1 = 0.1 \ , \ C_2 = 0.2, \tag{7}$$

where N_w and N are worst agents and the number of search agents, respectively.

Finally, at stage eight, an equation for updating the position of worst agents as follows:

$$G_{i,j} = G_{Min(i,j)} + r \times (\ G_{Max(i,j)} - G_{Min(i,j)}), \tag{8}$$

where $G_{i,j}$ shows the status for each gas i in group j, r used as a number which is distributed on [0,1], $G_{Min(i,j)}$ and $G_{Max(i,j)}$ represents the bounds for the algorithm. Algorithm 1 presents the steps of HGSO algorithm.

Algorithm 1: Pseudocode of HGSO algorithm

Step 1: **Initialization:** $Xi(1 = 1, 2, \ldots N)$, number of gas types $i, Hj, Pi, j, Cj, l1, l2$ and $l3$.

Step 2: Divide the population agents into number of gas types (cluster) with the same Henry's constant value (Hj).

Step 3: Evaluate each cluster j.

Step 4: Get the best gas $Xi, best$ in each cluster, and the best search agent $Xbest$.

Step 5: **while** $t <$ maximum number of iterations **do**

Step 6: **for** each search agent **do**

Step 7: Update the positions of all search agents using Eqs. (5-6).

Step 8: **end for**

Step 9: Update Henry's coefficient of each gas type using Eq. (3).

Step 10: Update solubility of each gas using Eq. (4).

Step 11: Rank and select the number of worst agents using Eqs. (7).

Step 12: Update the position of the worst agents using Eq. (8).

Step 13: Update the best gas $Xi,best$, and the best search agent $Xbest$.

Step14: **end while**

Step 15: $t = t + 1$

Step 16: **return** $Xbest$

4. Advanced Quantum Henry Gas Solubility Optimization algorithm

This section introduces a framework of our proposed algorithm which we call it Quantum Henry Gas Solubility Optimization (QHGSO) algorithm.

In this research, quantum theory is firstly used for updating the positions of solutions. Moreover, chaotic coefficient is used in quantum formula to reach deeper exploitation of the search space. For avoiding any local optimal solution, the proposed algorithm is empowered using a local search. The reason behind this modification for updating position formula is that the quantum theory predicts the position of each particle (Gas) based on a probability function. This probability function determines the most likely position for each particle (Gas) due to its best position achieved so far. In other words, this mechanism investigates the positions of the search space with high chance for being the optimum which introduces the exploitation aspect of our proposed algorithm. Furthermore, using chaotic numbers instead of random numbers in quantum formula enhances the exploration performance due to its power of generating diversified numbers to explore the search space.

4.1. Initialization process

The following formula represents the initial population of gases (*N*) and their positions:

$$X_i(t+1) = X_{min} + r \times (X_{max} - X_{min}), \tag{9}$$

where X_i shows the position of i^{th} gas in the population N, r is defined as a random number which is distributed between 0 and 1, X_{max} and X_{min} are problem boundaries and *t* presents the iteration. Moreover, the following equations defining the Henry's constant for each gas *i* type $(H_j(t))$, sectorial pressure $P_{i,j}$ for every gas *i* in group *j* and $\nabla_{sol}E/R$ is a fixed value of type $j(C_i)$:

$$H_j(t) = l_1 \times rand\ (0,1), P_{i,j} = l_2 \times rand\ (0,1), C_j = l_3 \times rand\ (0,1) \tag{10}$$

where l_1 to l_3 are constant numbers with value of 5E−02, 100, and 1E−02, respectively.

In the following, the members of the crowd are categorized into same clusters as to the number of gas species. The population agents are divided into equal clusters equivalent to the number of gas types. Each cluster has similar gases and therefore has the same Henry's constant value (H_j).

4.2. Evaluation

In this phase, evaluating for each cluster *j* will be performed to finding the best gas which reaches the sublime equilibrium position in comparison with others in its kind. Afterward, sorting method is used to identify the optimum gas among the whole crowd.

4.3. Updating the Henry' coefficient

The following formula explains the updating process of the Henry's coefficient.

$$H_j(t+1) = H_j(t) \times e^{(-C_j(\frac{1}{T(t)} - 1/T^{\theta}))} \quad , \quad T(t) = e^{(-t/iter)} \tag{11}$$

4.4. Updating solubility phase

The formula for updating solubility is described below:

$$S_{i,j}(t) = K \times H_j(t+1) \times P_{i,j}(t) \tag{12}$$

4.5. Updating the position based on quantum behavior

The dynamic behavior of a particle is entirely different in comparison with the particle in traditional swarm algorithms such as HGSO in which determining the exact values of *x* and v is not possible concurrently. There is just one thing is possible which is that calculating the probability of being a particle in position x from probability density function $|\Psi(x,t)|^2$. The density function shape is correlated to the potential field the particle lies in [68] and afterward the probability density function determines the probability distribution function of the particle's position. The updating position of each particle or gas is calculated in the following equation using Monte-Carlo method by [69]:

$$X_{i,j}(t+1) = \begin{cases} P_i - \beta * \left(M_{Best} - X_{i,j}(t)\right) * \ln\left(\frac{1}{u}\right), & if\ k \geq 0.5 \\ P_i + \beta * \left(M_{Best} - X_{i,j}(t)\right) * \ln\left(\frac{1}{u}\right), & if\ k \leq 0.5 \end{cases} \quad ((13)$$

$$P_i = \theta * pBest_i + (1-\theta) * gBest_i \quad ((14)$$

$$M_{best} = \frac{1}{N} \sum_{i=1}^{N} pBest_i \quad ((15)$$

Equations (13)-(15) are used for updating position of the gases in the HGSO. P_i presents the local attractor, $pBest_i$ illustrates the best position that the i^{th} gas has obtained up to now and $gBest_i$ represents the best position of all gases at each iteration. M_{best} introduces the average best status of the whole crowd, k is a random number between 0 and 1. Also, u and θ are chosen as chaotic numbers distributed on [0,1], since the chaotic function generates numbers with more diversity to help the exploration performance of the QHGSO. The parameter β in Eq. (14) indicates contraction expansion (CE) coefficient and utilized to control of the convergence rate. The value of parameter β starts from 1 and reduced permanently to 0.4 to seek the global optimum. The reason behind this is to empower the local search mechanism of the QHGSO. A formula for this parameter is introduced in Eq. (16) [70].

$$\beta = \beta_{max} - [\left\{\frac{\beta_{max}-\beta_{min}}{it_{max}}\right\} * it] \quad (16)$$

where β_{max} and β_{min} are the starting and ending points for contraction expansion factor, respectively. In addition, it indicates the contemporary iteration and $\boldsymbol{it_{max}}$ illustrates the last iteration number.

4.6. Escaping from a local optimum situation

In this step the following equations are used for avoiding in local optimum situations:

$$N_w = N * (rand((C_2 - C_1) + C_1), C_1 \;;\; C_1 = 0.1 \;\;,\; C_2 = 0.2 \quad (17)$$

$$X_{i,j}(t+1) = A_i * \left(X_{i,j}(t)\right) \quad (18)$$

4.7. Updating the position for each worst agent

We use the same equation in this step like the original ones as follows:

$$G_{i,j} = G_{Min(i,j)} + r \times (G_{Max(i,j)} - G_{Min(i,j)}) \quad (19)$$

The pseudocodes the QHGSO algorithm are shown in the following.

Algorithm 2: Pseudocode of QHGSO algorithm

Initialization: $Xi(1 = 1, 2, \ldots N)$, number of gas types i, Hj, Pi,j, Cj, $l1$, $l2$ and $l3$.

Divide the population agents into number of gas types (cluster) with the same Henry's constant value (Hj).

Evaluate each cluster j.

Get the best gas $Xi,best$ in each cluster, and the best search agent $Xbest$.

while $t <$ maximum number of iterations **do**

  for each search agent **do**

    Update the positions of all search agents using Eqs. (13-15).

  end for

  Update Henry's coefficient of each gas type using Eq. (11).

  Update solubility of each gas using Eq. (12).

  Rank and select the number of worst agents using Eqs. (17-18).

  Update the position of the worst agents using Eq. (19).

  Update the best gas $Xi,best$, and the best search agent $Xbest$.

end while

$t = t + 1$

return $Xbest$

From what has been discussed above, these modifications can make the HGSO faster and achieving better solutions in comparison with the state of the art of metaheuristics. This superiority will be discussed in forthcoming sections.

5. Numerical Experiments

This section provides the numerical results obtained using the proposed QHGSO algorithm.

5.1. Parameter settings

In terms of evaluating the QHSPO algorithm, we compare the performance of nine well-known metaheuristic algorithms, including Henry Gas Solubility Optimization Algorithm (HGSO) [40], Salp Swarm Algorithm (SSA) [71], Grasshopper Optimization Algorithm [72], Whale Optimization Algorithm (WOA) [73], Dragonfly Optimization Algorithm (DA) [74], Sine Cosine Algorithm (SCA) [75], Moth Flame Optimization (MFO) [76], Ant Lion Optimizer (ALO) [77], Grey Wolf Optimizer (GWO) [24]. We note that the parameters used in each method were used as mentioned in the original reference. In addition, for a fair comparison between these algorithms and the proposed algorithm, the common parameters are set to the same value for all these algorithms. For example, the population size is set to 30, the maximum number of iterations is set to 200 and for providing a statistical analysis each method was run 30 times. All algorithms are implemented using Matlab R2018b that installed on Windows 10 64bit with the system of 2.60 GHz processor with 4GB RAM.

5.2. Analysis on benchmark test functions

5.2.1. Description of test functions

In this section the performance of the proposed algorithm has been analyzed based on 47 standard test function (CEC'05) in comparison with nine well-known metaheuristics algorithms which are introduced recently. These functions are classified into three main groups: (1) unimodal, (2) multimodal, and (3) fixed dimension multimodal functions. The unimodal test functions (F1-F11) consist of just one global optimum point and used as a metric to reflect the exploitation power. While the multimodal test functions (F12-F47) consider a variety of local optimum spots to evaluate exploration performance; accordingly, these two types of benchmark functions can mirror the performance of the proposed algorithm in terms of exploitation and exploration capability. The definition of each function is described in Table 1 and Table 2 in Appendix. It is worth noting that the dimension of the function is shown by Dim, R and f_{min}, are the boundaries of the search space problem, and the value of fitness function, respectively. *Table 1 : Standard benchmark functions*

Type of functio	No.	Objective Function	Name	*Dim*	R	f_{min}
Unimodal functions	1	$f_1 = (\sum_{i=1}^{n} x_i^2)^2$	Chung Reynolds	30	[-100,100]	0
	2	$f_2(x) = \sum_{i=1}^{n} x_i^2$	Sphere	30	[-5.12,5.12]	0
	3	$f_3(x) = \sum_{i=1}^{D/4} (x_{4i-3} - 10x_{4i-2})^2 + 5\,(x_{4i-1} - x_{4i})^2 + (x_{4i-2} - x_{4i-1})^4 + 10\,(x_{4i-3} - x_{4i})^4$	Powell Singular 1	30	[-4,5]	0
	4	$f_4(x) = \sum_{i=2}^{D-2} (x_{i-1} - 10x_i)^2 + 5\,(x_{i+1} - x_{i+2})^2 + (x_i - 2x_{i+1})^4 + 10\,(x_{i-1} - x_{i+2})^4$	Powell Singular 2	30	[-4,5]	0
	5	$f_5(x) = \sum_{i=1}^{n} \lvert x_i \rvert^{i+1}$	Powell Sum	30	[-1,1]	0
	6	$f_6(x) = -\sum^{n} \lvert x_i \rvert$	Schwefel 2.20	30	[-100,100]	0
	7	$f_7(x) = Max_{1\le n\le n} \lvert x_i \rvert$	Schwefel 2.21	30	[-100,100]	0
	8	$f_8(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	Schwefel 2.22	30	[-100,100]	0
	9	$f_9(x) = \sum_{i=1}^{n} x_i^{10}$	Schwefel 2.23	30	[-10,10]	0
	10	$f_{10}(x) = \sum_{i=1}^{n} (\llbracket x_i \rrbracket)$	Step 1	30	[-100,100]	0
	11	$f_{11}(x) = \sum_{i=1}^{n} i * x_i^2$	Sum Squares	30	[-10,10]	0

Multimodal functions	12	$f_{12}(x) = -20\exp\left(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$	Ackley	30	[-35,35]	0
	13	$f_{13}(x) = \sum_{i=1}^{d} \lvert x_i \sin(x_i) + 0.1\, x_i \rvert$	Alpine	30	[-10,10]	0
	14	$f_{14}(x) = \sum_{i=1}^{n-1} ((x_i^2)^{(x_{i+1}^2+1)}) + (x_{i+1}^2)^{(x_i^2+1)})$	Brown	30	[-1,4]	0
	15	$f_{15}(x) = x_1^2 + 10^6 \sum_{i=2}^{n} x_i^2$	Cigar	30	[-100,100]	0
	16	$f_{16}(x) = -\exp\left(-0.5\sum_{i=1}^{n} x_i^2\right)$	Exponential	30	[-1,1]	-1
	17	$f_{17}(x) = \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	Griewank	30	[-600,600]	0
	18	$f_{18}(x) = (1 - n - \sum_{i=1}^{n-1} x_i)^{n - \sum_{i=1}^{n-1} x_i}$	Mishra 1	30	[0,1]	2
	19	$f_{19}(x) = (1 - n - \sum_{i=1}^{n-1} 0.5(x_i + x_{i+1}))^{n - \sum_{i=1}^{n-1} 0.5(x_i + x_{i+1})}$	Mishra 2	30	[0,1]	2
	20	$f_{20}(x) = \left[\frac{1}{n}\sum_{i=1}^{n} \lvert x_i \rvert - (\prod_{i=1}^{n} \lvert x_i \rvert)^{\frac{1}{n}}\right]^2$	Mishra 11	30	[0,10]	0
	21	$f_{21}(x) = \sum_{i=1}^{n} i x_i^4 + random\,[0,1]$	Quartic	30	[-1.28,1.28]	0
	22	$f_{22}(x) = 10d + \sum_{i=1}^{d} [x_i^2 - 10\cos 2\pi x_i]$	Rastrigin	30	[-5.12,5.12]	0
	23	$f_{23}(x) = \sum_{i=2}^{n} \left[(x_i - 1)^2 + (x_1 - x_i^2)^2\right]$	Schwefel 2.25	30	[0,10]	0
	24	$f_{24}(x) = \left[\left(-\sum_{i=1}^{n} \lvert x_i \rvert\right) * \exp(-\sum_{i=1}^{n} \sin(x_i)^2)\right]$	Xin-She Yang 2	30	[-2 π, 2 π]	0
	25	$f_{25}(x) = \left[\exp\left(-\sum_{i=1}^{n}\left(\frac{x_i}{15}\right)^{10}\right) - 2\exp\left(-\sum_{i=1}^{n}(x_i)^2\right) * \prod_{i=1}^{n} cos^2(x_i)\right]$	Xin-She Yang 3	30	[-20,20]	0
	26	$f_{26}(x) = \sum_{i=1}^{n} x_i^2 + (\frac{1}{2}\sum_{i=1}^{n} i x_i^2)^2 + (\frac{1}{2}\sum_{i=1}^{n} i x_i^2)^4$	Zakharov	30	[-5,10]	0

Table 2 : Fixed-dimension multimodal functions description.

No.	Objective Function	Name	*Dim*	R	f_{min}

27	$f_{27}(x) = 200 * \exp(-0.2 * (\sqrt{x_1{}^2 + x_2{}^2}))$	Ackley 2	2	[-32,32]	0
28	$f_{28}(x) = \lvert x_1{}^2 + x_2{}^2 + x_1 * x_2 \rvert + \lvert\sin(x_1)\rvert + \lvert\cos(x_2)\rvert$	Bartels Conn	2	[-500,500]	0
29	$f_{29}(x) = x_1{}^2 + 2x_2{}^2 - 0.3\ \cos(3\pi x_1) - 0.4\ \cos(4\pi x_2) + 0.7$	Bohachevsky 1	2	[-100,100]	0
30	$f_{30}(x) = x_1{}^2 + 2x_2{}^2 - 0.3\ \cos(3\pi x_1) * 0.4\ \cos(4\pi x_2) + 0.3$	Bohachevsky 2	2	[-100,100]	0
31	$f_{31}(x) = x_1{}^2 + 2x_2{}^2 - 0.3\ \cos(3\pi x_1 + 4\pi x_2) + 0.3$	Bohachevsky 3	2	[-100,100]	0
32	$f_{32}(x) = 1.05x_1{}^4 + 2x_1{}^2 + \frac{1}{6} * x_1{}^6 + x_1 * x_2 + x_2{}^2$	Camel-Three Hump	2	[-5,5]	0
33	$f_{33}(x) = x_1{}^2 + 12x_1 + 11 + 10\ \cos(\pi x_1/2) + 8\sin(5\pi x_1/2) - \left(\frac{1}{5}\right)^{0.5} \exp\left(-0.5(x_2 - 0.5)^2\right)$	Chichinadze	2	[-30,30]	0
34	$f_{34}(x) = -.0001\left[\left\lvert\sin(x_1)\ \cos(x_2)\exp\lvert 100 - [x_1{}^2 + x_2{}^2]^{0.5}/\pi\rvert\right\rvert + 1\right]^{0.1}$	Cross-in-Tray	2	[-10,10]	0
35	$f_{35}(x) = -\left(\frac{1}{\left(\left\lvert e^{\left\lvert 100 - \frac{\sqrt{x_1{}^2 + x_2{}^2}}{\pi}\right\rvert} \sin(x_1)\cos(x_2)\right\rvert + 1\right)^{0.1}}\right)$	ScCrossLegTable *	2	[-10,10]	0
36	$f_{36}(x) = x_1{}^2 + x_2{}^2 + 25\ (Sin^2(x_1) + Sin^2(x_2)$	Egg Crate	2	[-5,5]	0
37	$f_{37}(x) = -\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{6} a_{ij}\ (x_j - p_{ij})^2\right]$	Hartman	6	[0,1]	0
38	$f_{38}(x) = 0.26(x_1{}^2 + x_2{}^2) - 0.04x_1x_2$	Matyas	2	[-10,10]	0
39	$f_{39}(x) = 1 + Sin^2(x_1) + Sin^2(x_2) - 0.1e^{-x_1{}^2 - x_2{}^2}$	Periodic	2	[-10,10]	0
40	$f_{40}(x) = (333.75 - x_1{}^2) - x_2{}^6 + x_1{}^2(11x_1{}^2x_2{}^4 - 2) + 5.5x_2{}^8 + \frac{x_1}{2x_2}$	Rump	2	[-500,500]	0
41	$f_{41}(x) = x_1{}^2 - x_1x_2 + x_2{}^2$	Rotated Ellipse	2	[-500,500]	0
42	$f_{42}(x) = g(r).h(t)$, $where, g(r) = \left[\sin(r) - \frac{\sin(2r)}{2} + \frac{\sin(3r)}{3} + \frac{\sin(4r)}{4} + 4\right](\frac{r^2}{r+1})$ $, h(t) = 0.5\cos(2t - 0.5) + \cos(t) + 2\ , r = \sqrt{x_1{}^2 + x_2{}^2}\ , t = atan2(x_1, x_2)$	Sawtoothxy	2	[-20,20]	-1

43	$f_{43}(x) = 0.5 + \frac{\sin^2({x_1}^2 + {x_2}^2)^2 - 0.5}{1 + 0.001({x_1}^2 + {x_2}^2)^2}$	Scahffer1	2	[-100,100]	0
44	$f_{44}(x) = 0.5 + \frac{\sin^2\sqrt{{x_1}^2 + {x_2}^2} - 0.5}{\left[1 + 0.001({x_1}^2 + {x_2}^2)^2\right]}$	Schaffer6	2	[-100,100]	2
45	$f_{45}(x) = ({x_1}^2 - 4{x_2}^2)^2 + ({x_1}^2 - 2x_1 + 4x_2)^2$	Stenger	2	[-1,4]	2
46	$f_{46}(x) = 4 - 4{x_1}^3 + 4x_1 + {x_2}^2$	Trecanni	2	[-5,5]	0
47	$f_{47}(x) = {x_1}^2 - 100\cos(x_1)^2 - 100\cos\left(\frac{{x_1}^2}{30}\right) + {x_2}^2$	Venter	2	[-50,50]	0

5.2.2. Analysis of exploration and exploitation and measures of performance

In terms of trading off between exploration and exploitation to better search the space solution and avoiding any local optimum situations, our proposed algorithm is benefited from three parameters to reach the aforementioned points: $\boldsymbol{S_{i,j}}$, $\boldsymbol{A}$ and $\boldsymbol{F}$. $\boldsymbol{S_{i,j}}$ represents the solubility for each individual gas $\boldsymbol{i}$ at every group $\boldsymbol{j}$ which is rely on each iteration. $\boldsymbol{A}$ is a chaotic coefficient for updating and escaping local optimum situations. And finally, $\boldsymbol{F}$ represents the way which has a potential to create diversity by means of changing the way of seeking for some search agents.

In this research, for gaining the exploration and exploitation, a dimension-wise manifold evaluation technique represented by Hussain et al. [72] is used. Therefore, the average value together with high distance within dimensions indicates exploration and the opposite situation represents the exploitation. Therefore, for evaluating the performance of each algorithm in comparison with our proposed metaheuristics, the following measures are used:

Mean of fitness values (Mean):

$$Mean = \frac{1}{N_r}\sum_{i=1}^{N_r} F_i, \tag{20}$$

Standard deviation (*STD*):

$$STD = \sqrt{\frac{1}{N_r-1}\sum_{i=1}^{N_r}(F_i - mean)^2}. \tag{21}$$

5.3. Results and discussion

In this section, the comparison results between the proposed quantum HGSO algorithm and other methods such as HGSO, SSA, GOA, WOA, DA, SCA, MFO, ALO and GWO algorithms. The results are given in Table 3-6 and Figure 2.

From these results it can be noticed the superiority of our proposed algorithm in terms of Mean and STD metrics as in Tables 3-5. For example, the QHGSO obtains best values for F1-F11, F15-F17, F24, F26-F34, and F36-F46. In the case of assessing the performance of the proposed QHGSO to solve the unimodal functions, it can be observed that it has high ability to find solution than other algorithms. For the multi modal test functions, all algorithms obtained near optimal solutions, however, the proposed QHSGO still provides better performance than most of them. Finally, in the fixed dimension test functions, QHGSO reached optimal point in F27, F29, F31, F32, F34, F36, F38-F47 while others do not reach the optimal point in these test functions. It is worth mentioning that QHGSO obtained the best results in comparison with the traditional HGSO at F1 till F11 except F5 which had the equal results. In the following, in the fixed-dimension functions, QHGSO reached best results among other algorithms specially HGSO in 13 test functions and in the rest of them obtained equal except F35. Besides, in terms of investigating performance stability of algorithms, it can be seen the high stability of the proposed QHGSO which allocates the first rank in STD metric among other comparison algorithms.

Table 3 : Comparison results obtained for the unimodal benchmark functions in terms of average of fitness value.

F	Measure	QHGSO	HGSO	SSA	GOA	WOA	DA	SCA	MFO	ALO	GWO
F1	Mean	**0.00E+00**	8.05E-137	3.74E+03	5.14E+04	1.02E-48	1.53E+07	3.44E+06	3.25E+07	8.31E+05	1.45E-16
	STD	**0.00E+00**	4.41E-136	3.66E+03	1.11E+05	3.90E-48	1.97E+07	8.90E+06	5.64E+07	1.53E+06	2.82E-16
F2	Mean	**0.00E+00**	3.44E-75	1.22E-01	5.81E-01	1.93E-29	8.26E+00	2.74E+00	6.11E+00	2.08E+00	1.31E-11
	STD	**0.00E+00**	1.88E-74	8.14E-02	4.94E-01	9.42E-29	4.48E+00	3.09E+00	6.76E+00	1.47E+00	8.80E-12
F3	Mean	**0.00E+00**	9.57E-68	1.39E+01	7.87E+00	1.56E-07	2.40E+02	1.83E+02	1.31E+03	1.91E+01	1.30E-04
	STD	**0.00E+00**	5.24E-67	1.19E+01	6.17E+00	4.18E-07	1.92E+02	1.88E+02	1.42E+03	1.44E+01	9.80E-05
F4	Mean	**0.00E+00**	2.71E-60	3.15E+01	4.18E+01	1.86E-27	9.39E+02	5.97E+02	4.17E+03	7.87E+01	1.77E-07
	STD	**0.00E+00**	1.48E-59	2.53E+01	4.76E+01	7.47E-27	7.17E+02	6.36E+02	5.37E+03	8.01E+01	2.49E-07
F5	Mean	**0.00E+00**	0.00E+00	2.59E-262	5.17E-229	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.16E-243	0.00E+00
	STD	**0.00E+00**	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F6	Mean	**0.00E+00**	2.44E-61	7.76E-08	1.34E-06	4.42E-70	1.66E-16	2.03E-23	1.03E-43	2.56E-07	4.60E-84
	STD	**0.00E+00**	7.06E-61	8.60E-08	1.45E-06	1.93E-69	8.51E-16	4.71E-23	5.64E-43	2.66E-07	2.52E-83
F7	Mean	**0.00E+00**	3.02E-37	1.82E+01	9.72E+00	5.44E+01	3.63E+01	5.65E+01	6.90E+01	2.45E+01	3.54E-02
	STD	**0.00E+00**	1.31E-36	3.82E+00	2.70E+00	2.51E+01	9.91E+00	8.02E+00	9.14E+00	5.10E+00	2.86E-02
F8	Mean	**0.00E+00**	2.77E-38	5.67E+18	1.61E+31	1.11E-18	3.31E+02	6.12E+00	7.47E+02	9.47E+24	8.54E-05
	STD	**0.00E+00**	1.07E-37	3.05E+19	6.27E+31	3.52E-18	1.84E+02	6.20E+00	2.39E+02	5.18E+25	6.60E-05
F9	Mean	**0.00E+00**	0.00E+00	1.32E+01	1.32E-01	8.43E-61	4.05E+05	3.53E+07	4.66E+06	2.40E+03	1.01E-31
	STD	**0.00E+00**	0.00E+00	3.26E+01	6.02E-01	4.59E-60	8.74E+05	7.80E+07	1.11E+07	9.87E+03	3.53E-31
F10	Mean	**0.00E+00**	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	STD	**0.00E+00**	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F11	Mean	**0.00E+00**	1.10E-121	3.40E-15	6.32E-13	7.72E-142	2.23E-26	1.40E-44	1.95E-89	5.11E-14	9.10E-164
	STD	**0.00E+00**	6.00E-121	5.89E-15	8.37E-13	3.99E-141	1.18E-25	6.57E-44	8.51E-89	8.69E-14	0.00E+00

Table 4 : Statistical results obtained for the multimodal functions in terms of average of fitness value.

F	Measure	QHGSO	HGSO	SSA	GOA	WOA	DA	SCA	MFO	ALO	GWO
F12	Mean	8.70E+00	1.87E+01	3.92E-20	9.61E+00	7.28E+00	4.26E+00	1.46E+01	7.83E-03	0.00E+00	1.39E-87
	STD	6.12E+00	1.39E+01	3.36E-19	1.67E+01	1.88E+00	2.97E+00	7.07E+00	3.81E-03	0.00E+00	7.84E-89
F13	Mean	8.65E-01	1.26E+01	5.86E-30	1.04E+02	5.34E-02	1.09E+01	3.20E-02	7.07E-11	0.00E+00	2.00E-104
	STD	2.31E+00	1.03E+01	1.53E-27	9.77E+01	6.46E-02	2.95E+01	3.65E-03	9.32E-12	0.00E+00	5.14E-102
F14	Mean	7.17E+02	1.00E+03	1.47E-57	9.25E+02	2.80E-31	3.37E-42	6.22E+03	4.94E-95	-1.00E+00	-1.00E+00
	STD	1.90E+03	2.17E+03	1.45E-48	0.00E+00	1.38E-30	3.32E-36	4.27E+02	1.11E-98	-1.00E+00	-1.00E+00
F15	Mean	**-9.98E-01**	-3.74E-01	-1.00E+00	-8.30E-01	-9.69E-01	-9.48E-01	-8.34E-01	-1.00E+00	0.00E+00	0.00E+00
	STD	**-9.98E-01**	-2.87E-01	-1.00E+00	-9.37E-01	-9.74E-01	-9.71E-01	-7.83E-01	-1.00E+00	0.00E+00	0.00E+00
F16	Mean	1.48E+00	2.61E+00	0.00E+00	1.02E+02	5.25E+00	2.64E+01	1.01E+01	3.37E-02	2.00E+00	2.00E+00
	STD	1.49E+00	4.19E+00	0.00E+00	1.69E+01	6.87E+00	1.34E+01	7.23E+00	4.75E-09	2.00E+00	2.00E+00
F17	Mean	1.61E+01	6.31E+08	2.00E+00	2.00E+00	1.30E+11	2.00E+00	2.00E+00	2.30E+00	2.00E+00	2.00E+00
	STD	1.36E+01	5.61E+07	2.00E+00	2.00E+00	3.06E+11	2.00E+00	2.00E+00	1.07E+01	2.00E+00	2.00E+00
F18	Mean	4.35E+00	5.40E+08	2.00E+00	2.00E+00	1.28E+10	2.00E+00	2.00E+00	2.54E+00	0.00E+00	4.19E-05
	STD	1.78E+01	1.31E+10	2.00E+00	2.00E+00	5.22E+11	2.00E+00	2.00E+00	5.32E+00	0.00E+00	2.30E-09
F19	Mean	7.35E-19	3.05E-18	0.00E+00	4.71E-27	3.25E-08	0.00E+00	0.00E+00	2.76E-12	2.52E-05	5.01E-04
	STD	4.08E-17	3.64E-17	0.00E+00	7.39E-26	0.00E+00	0.00E+00	0.00E+00	1.02E-10	1.07E-04	5.47E-05
F20	Mean	**4.16E-01**	2.85E+00	1.11E-02	9.72E-01	5.73E-01	9.25E-01	1.28E+00	4.90E-03	0.00E+00	0.00E+00
	STD	**2.56E-01**	3.03E+00	3.34E-02	7.68E-01	3.29E-01	4.05E-01	6.57E-01	4.73E-03	0.00E+00	0.00E+00
F21	Mean	6.56E+01	1.77E+02	0.00E+00	1.92E+02	3.57E+01	1.52E+02	8.85E+01	1.35E+01	0.00E+00	8.12E-86
	STD	4.64E+01	2.06E+02	0.00E+00	2.36E+02	8.95E+01	1.86E+02	8.05E+01	1.05E+00	0.00E+00	3.15E-84
F22	Mean	2.69E+02	4.62E+02	0.00E+00	3.84E+02	0.00E+00	1.40E-03	0.00E+00	1.20E-11	0.00E+00	3.23E-63
	STD	1.95E+02	1.75E+02	0.00E+00	0.00E+00	0.00E+00	6.27E+02	0.00E+00	6.08E-10	0.00E+00	6.14E-65
F23	Mean	**3.27E-09**	9.85E-08	1.01E-76	7.92E-14	1.55E-26	5.85E-49	1.86E-08	1.77E-106	1.94E-08	1.94E-08
	STD	**1.03E-09**	1.16E-07	4.79E-78	0.00E+00	1.11E-23	6.18E-48	3.50E-08	7.68E-113	3.45E-32	0.00E+00
F24	Mean	1.94E-08	1.94E-08	1.94E-08	1.94E-08	1.94E-08	1.94E-08	1.94E-08	1.94E-08	0.00E+00	1.07E-39
	STD	1.94E-08	1.94E-08	1.94E-08	1.94E-08	1.94E-08	1.94E-08	1.94E-08	1.94E-08	0.00E+00	6.25E-51
F25	Mean	2.43E+02	4.51E+02	6.59E+02	1.81E+02	6.54E+01	4.00E+02	3.77E+02	3.33E-02	-2.00E+02	-2.00E+02
	STD	9.38E+01	1.46E+02	5.96E+02	6.64E+02	5.62E+01	4.90E+02	2.26E+02	1.99E-01	-2.00E+02	-2.00E+02
F26	Mean	**-2.00E+02**	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	1.00E+00	1.00E+00
	STD	**-2.00E+02**	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	1.00E+00	1.00E+00

Table 5 : Statistical results obtained for the fixed-dimension multimodal functions in terms of average of fitness value.

F	Measure	QHGSO	HGSO	SSA	GOA	WOA	DA	SCA	MFO	ALO	GWO
F27	Mean	**-2.00E+02**	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02	-2.00E+02
	STD	**0.00E+00**	0.00E+00	1.60E-06	1.78E-05	3.08E-14	2.81E-05	2.33E-13	0.00E+00	3.65E-06	0.00E+00
F28	Mean	**1.00E+00**	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00
	STD	**0.00E+00**	0.00E+00	9.88E-05	1.57E-03	0.00E+00	5.07E-04	0.00E+00	0.00E+00	4.80E-05	0.00E+00
F29	Mean	**0.00E+00**	0.00E+00	1.55E-10	1.38E-02	0.00E+00	6.14E-04	0.00E+00	0.00E+00	3.45E-10	0.00E+00
	STD	**0.00E+00**	0.00E+00	1.35E-10	7.54E-02	0.00E+00	3.36E-03	0.00E+00	0.00E+00	3.21E-10	0.00E+00
F30	Mean	**1.80E-01**	1.80E-01	1.80E-01	1.80E-01	1.80E-01	1.80E-01	1.80E-01	1.80E-01	1.80E-01	1.80E-01
	STD	**0.00E+00**	0.00E+00	6.42E-11	7.54E-09	0.00E+00	1.02E-04	0.00E+00	0.00E+00	1.61E-10	0.00E+00
F31	Mean	**0.00E+00**	0.00E+00	4.46E-11	2.53E-08	5.16E-05	5.05E-05	0.00E+00	3.56E-08	3.48E-10	0.00E+00
	STD	**0.00E+00**	0.00E+00	7.50E-11	2.97E-08	1.16E-04	2.74E-04	0.00E+00	1.89E-07	6.80E-10	0.00E+00
F32	Mean	**0.00E+00**	4.88E-106	2.22E-14	5.07E-12	2.99E-02	8.52E-12	1.54E-29	3.57E-39	9.95E-03	7.63E-74
	STD	**0.00E+00**	1.86E-105	2.87E-14	3.05E-12	9.11E-02	4.67E-11	7.73E-29	1.94E-38	5.45E-02	4.12E-73
F33	Mean	-4.26E+01	-4.29E+01	-4.27E+01	-4.26E+01	-4.26E+01	-4.26E+01	-4.28E+01	-4.29E+01	-4.26E+01	-4.27E+01
	STD	1.54E-01	1.09E-01	2.19E-01	1.92E-01	2.02E-01	1.69E-01	1.73E-01	3.61E-14	2.14E-01	2.07E-01
F34	Mean	**-2.06E+00**	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00
	STD	**1.17E-07**	1.00E-04	5.49E-15	1.00E-12	1.78E-06	2.48E-12	5.89E-05	9.03E-16	3.24E-14	8.52E-08
F35	Mean	6.55E+04	-1.00E+00	-1.00E+00	-1.00E+00	-1.00E+00	5.68E+04	-1.00E+00	-1.00E+00	-1.00E+00	-1.00E+00
	STD	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	2.27E+04	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F36	Mean	**0.00E+00**	3.46E-107	4.95E-13	6.33E+00	5.10E-46	4.87E-13	1.94E-30	9.53E-42	1.50E-12	5.24E-85
	STD	**0.00E+00**	8.94E-107	5.75E-13	5.19E+00	2.66E-45	2.38E-12	6.01E-30	2.69E-41	1.16E-12	2.35E-84
F37	Mean	-2.33E+00	-2.87E+00	-3.23E+00	-3.25E+00	-3.16E+00	-3.25E+00	-2.79E+00	-3.23E+00	-3.27E+00	-3.24E+00
	STD	3.79E-01	2.02E-01	8.31E-02	6.23E-02	2.39E-01	8.59E-02	4.57E-01	6.08E-02	7.07E-02	1.14E-01
F38	Mean	**0.00E+00**	6.45E-96	4.85E-15	2.96E-12	4.35E-76	1.99E-09	6.06E-22	3.09E-13	1.46E-14	9.38E-42
	STD	**0.00E+00**	2.74E-95	5.20E-15	5.96E-12	2.36E-75	1.09E-08	2.34E-21	9.87E-13	1.81E-14	4.08E-41
F39	Mean	**9.00E-01**	9.00E-01	9.33E-01	9.97E-01	9.27E-01	9.33E-01	9.10E-01	9.60E-01	9.43E-01	9.37E-01
	STD	**4.52E-16**	4.52E-16	4.79E-02	1.83E-02	4.50E-02	4.79E-02	3.08E-02	4.98E-02	5.04E-02	4.92E-02
F40	Mean	**0.00E+00**	2.48E-05	1.99E-03	1.19E+03	7.52E+02	5.75E-09	2.95E-04	4.46E-17	6.22E+00	2.85E-05
	STD	**0.00E+00**	1.36E-04	7.37E-03	3.79E+03	4.12E+03	1.50E-08	7.02E-04	1.66E-16	2.77E+01	5.25E-05
F41	Mean	**0.00E+00**	4.32E-98	1.09E-10	2.81E-08	7.00E-61	3.23E-13	5.50E-26	1.59E-35	3.54E-10	1.28E-62
	STD	**0.00E+00**	2.37E-97	2.11E-10	2.77E-08	3.83E-60	1.77E-12	2.12E-25	8.21E-35	3.23E-10	7.01E-62
F42	Mean	**0.00E+00**	2.71E-103	2.05E-12	2.85E-10	6.73E-34	5.72E-08	4.73E-28	1.90E-40	8.02E-12	8.35E-73
	STD	**0.00E+00**	1.48E-102	2.09E-12	2.34E-10	2.79E-33	3.13E-07	1.33E-27	4.14E-40	9.41E-12	4.58E-72
F43	Mean	**0.00E+00**	0.00E+00	5.96E-15	1.09E-12	8.32E-05	6.00E-13	0.00E+00	0.00E+00	4.03E-14	0.00E+00

	STD	**0.00E+00**	0.00E+00	6.31E-15	7.88E-13	4.56E-04	1.96E-12	0.00E+00	0.00E+00	4.20E-14	0.00E+00
F44	**Mean**	**0.00E+00**	0.00E+00	7.77E-03	7.45E-03	9.88E-03	7.13E-03	3.45E-03	8.42E-03	7.13E-03	7.61E-03
	STD	**0.00E+00**	0.00E+00	3.95E-03	4.18E-03	1.02E-02	4.37E-03	4.56E-03	3.36E-03	4.37E-03	3.93E-03
F45	**Mean**	**0.00E+00**	9.63E-100	7.25E-14	2.01E-11	4.76E-08	1.32E-14	9.75E-05	3.28E-14	1.42E-13	1.31E-06
	STD	**0.00E+00**	3.66E-99	1.04E-13	2.17E-11	1.65E-07	7.22E-14	3.48E-04	1.80E-13	1.29E-13	2.77E-06
F46	**Mean**	**0.00E+00**	3.26E-102	3.42E-14	7.02E-12	1.20E-05	2.39E-15	1.08E-12	-1.54E-15	9.14E-14	7.08E-06
	STD	**0.00E+00**	1.78E-101	3.14E-14	5.67E-12	5.29E-05	1.93E-14	5.93E-12	1.79E-15	1.06E-13	1.22E-05
F47	**Mean**	-4.00E+02	-4.00E+02	-3.99E+02	-4.00E+02	-4.00E+02	-3.99E+02	-4.00E+02	-4.00E+02	-3.98E+02	-4.00E+02
	STD	0.00E+00	0.00E+00	3.76E+00	2.34E-08	0.00E+00	4.52E+00	0.00E+00	0.00E+00	5.62E+00	0.00E+00

To recapitulate, as can be seen in Figure 2, the convergence curve for QHGSO algorithm in test function 46 has a very fast slope and it converges faster than the other algorithms. Therefore, due to the aforementioned facts, it can be inferred that our proposed algorithm has the superiority on other well-known algorithms, including the original HGSO algorithm.

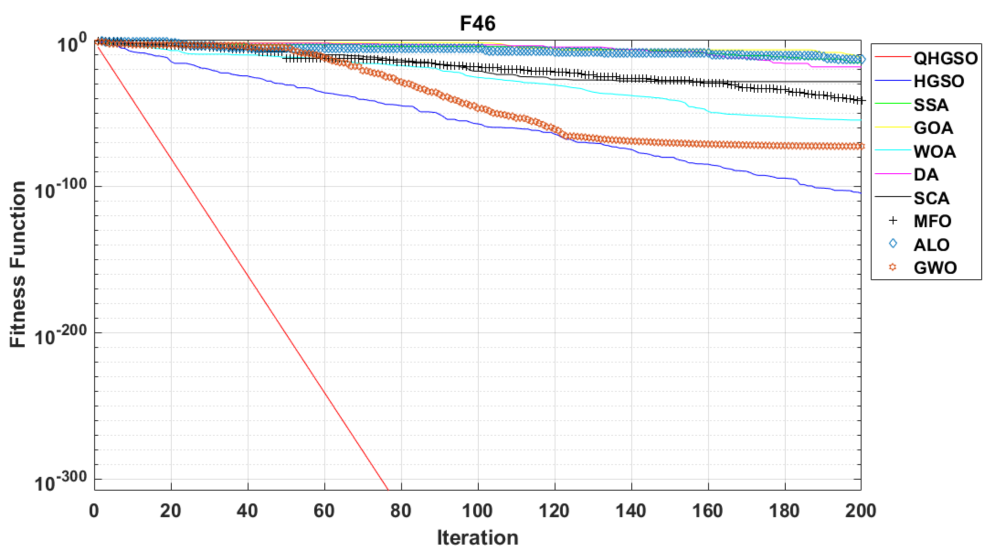

Figure 2 : Convergence curves of competitive algorithms

5.4. Non-parametric test analysis

Overall, it is evident that evaluating the performance of our proposed algorithm just based on mean value and STD may not be reliable since the uncertainty of 30 runs is an inevitable process. Due to this fact, we use Wilcoxon sum rank test [78] with p-value at 5% as a non-parametric test for determining the significance difference between our proposed algorithm comparing to the other nine famous aforementioned metaheuristics in Table 6.

This method illustrates the supremacy of our proposed algorithm due to the p-values, that are lower than 0.05. However, there is no significant difference between the proposed method other methods at some functions such as F10 and F12. Also, no significant difference with HGSO, WOA, DA, SCA, MFO, and GWO at F5. At F27-29 there is no significant difference between the proposed QHGSO and WOA, SCA, MFO, and GWO.

Table 6 : The results of the Wilcoxon's rank sum test for comparison between QHGSO and other algorithms.

		HGSO	SSA	GOA	WOA	DA	SCA	MFO	ALO	GWO
F1	P-val	1.21E-12	1.2E-12	1.2E-12	1.2E-12	1.2E-12	1.2E-12	1.2E-12	1.2E-12	1.2E-12
	H	1	1	1	1	1	1	1	1	1
F2	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F3	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F4	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F5	P-val	NAN	5.85E-09	1.21E-12	NAN	NAN	NAN	NAN	1.21E-12	NAN
	H	0	1	1	0	0	0	0	1	0
F6	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	0.160802	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	0	1	1	1	1
F7	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F8	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F9	P-val	0.333711	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	0	1	1	1	1	1	1	1	1
F10	P-val	NAN	NAN	NAN	NAN	NAN	NAN	NAN	NAN	NAN
	H	0	0	0	0	0	0	0	0	0
F11	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	0.081523	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	0	1	1	1	1
F12	P-val	NAN	NAN	NAN	NAN	NAN	NAN	NAN	NAN	NAN
	H	0	0	0	0	0	0	0	0	0
F13	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F14	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F15	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	3.45E-07	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F16	P-val	NAN	3.49E-12	1.21E-12	NAN	1.21E-12	1.21E-12	1.2E-12	1.21E-12	NAN
	H	0	1	1	0	1	1	1	1	0
F17	P-val	NAN	1.21E-12	1.21E-12	0.041911	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	0	1	1	1	1	1	1	1	1

F18	P-val	NAN	1.21E-12	1.21E-12	NAN	NAN	1.21E-12	0.041926	NAN	1.21E-12
	H	0	1	1	0	0	1	1	0	1
F19	P-val	NAN	1.21E-12	1.21E-12	NAN	NAN	1.21E-12	0.081523	NAN	1.21E-12
	H	0	1	1	0	0	1	0	0	1
F20	P-val	1.21E-12	1.21E-12	1.21E-12	NAN	1.94E-09	1.27E-05	0.000313		1.21E-12
	H	1	1	1	0	1	1	1	0	1
F21	P-val	6.12E-10	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
	H	1	1	1	1	1	1	1	1	1
F22	P-val	NAN	1.21E-12	1.21E-12	0.010994	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	0	1	1	1	1	1	1	1	1
F23	P-val	1.21E-12	1.21E-12	1.21E-12	NAN	1.27E-05	NAN	1.21E-12	NAN	1.21E-12
	H	1	1	1	0	1	0	1	0	1
F24	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	0.011035	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F25	P-val	0.068253	0.000661	0.000661	0.000661	0.000661	0.000661	0.000661	0.000661	0.000661
	H	0	1	1	1	1	1	1	1	1
F26	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F27	P-val	NAN	1.2E-12	1.21E-12	NAN	0.041926	NAN	NAN	1.21E-12	NAN
	H	0	1	1	0	1	0	0	1	0
F28	P-val	NAN	1.21E-12	1.21E-12	NAN	5.85E-09	NAN	NAN	1.21E-12	NAN
	H	0	1	1	0	1	0	0	1	0
F29	P-val	NAN	1.21E-12	1.21E-12	NAN	0.011035	NAN	NAN	1.21E-12	NAN
	H	0	1	1	0	1	0	0	1	0
F30	P-val	NAN	NAN	1.19E-12	NAN	0.160802	NAN	NAN	0.160742	NAN
	H	0	0	1	0	0	0	0	0	0
F31	P-val	NAN	1.21E-12	1.21E-12	1.21E-12	0.005584		0.000662	1.21E-12	
	H	0	1	1	1	1	0	1	1	0
F32	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	5.85E-09	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F33	P-val	6.01E-08	1.06E-05	0.000224	6.09E-05	0.000569	7.2E-05	1.21E-12	3.35E-05	3.57E-06
	H	1	1	1	1	1	1	1	1	1
F34	P-val	2.78E-11	5.26E-11	5.26E-11	0.279929	5.26E-11	7.54E-11	5.26E-11	5.26E-11	1.13E-05
	H	1	1	1	0	1	1	1	1	1
F35	P-val	1.69E-14	1.69E-14	1.69E-14	1.69E-14	0.041774	1.69E-14	1.69E-14	1.69E-14	1.69E-14
	H	1	1	1	1	1	1	1	1	1
F36	P-val	1.21E-12	1.21E-12	5.36E-13	1.21E-12	6.25E-10	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F37	P-val	1.16E-07	2.52E-11	2.63E-11	5.07E-10	3.02E-11	6.74E-06	1.65E-11	1.44E-11	3.69E-11
	H	1	1	1	1	1	1	1	1	1
F38	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	6.25E-10	1.21E-12	1.21E-12	1.21E-12	1.21E-12

	H	1	1	1	1	1	1	1	1	1
F39	P-val	NAN	0.00063	2.57E-13	0.002787	0.000618	0.081523	5.19E-07	5.98E-05	0.000313
	H	0	1	1	1	1	0	1	1	1
F40	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	0.005582	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F41	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.66E-11	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F42	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	3.45E-07	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F43	P-val	NAN	1.2E-12	1.21E-12	0.160802	0.021577	NAN	NAN	1.21E-12	NAN
	H	0	1	1	0	1	0	0	1	0
F44	P-val	NAN	1.6E-13	2.07E-13	2.75E-09	1.77E-09	2.21E-06	1.97E-11	2.62E-13	1.67E-09
	H	0	1	1	1	1	1	1	1	1
F45	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	0.011035	1.21E-12	4.57E-12	1.21E-12	1.21E-12
	H	1	1	1	1	1	1	1	1	1
F46	P-val	1.21E-12	1.21E-12	1.21E-12	1.21E-12	0.143116	1.21E-12	0.344232	1.21E-12	1.21E-12
	H	1	1	1	1	0	1	0	1	1
F47	P-val	NAN	0.160742	0.002698	NAN	0.041865	NAN	NAN	0.010953	NAN
	H	0	0	1	0	1	0	0	1	0

5.5 Influence of changing the parameters

In this section, we study the influence of changing the value of the parameters on the performance of the proposed algorithm. This achieved through considering five Scenarios according to the parameters L1, L2, L3, C1, and C2 as given in Table 7. The comparison results are given in Table 8 and it can be noticed the scenarios 2, 3, and 5 provide better results than other scenarios. In addition, by using the Friedman test it can be noticed that mean rank of scenarios 1, 2, 3, 4, and 5 is 2.5, 2.87, 2.6, 4, and 3 with p-value 0.1842. This indicates there is no significant difference.

Table 7: The five Scenarios to test the influence of parameters.

Parameter	Original Value	Scenarios				
		1	2	3	4	5
L1	5E−02	5E−04	5E−02	5E−02	5E−02	5E−02
L2	100	100	120	100	100	100
L3	1E−02	1E−02	1E−02	1E−07	1E−02	1E−02
$C1$	0.1	0.1	0.1	0.1	0.2	0.1
$C2$	0.2	0.2	0.2	0.2	0.2	0.3

Table 8: Results of variant set of values for five parameters.

Scenarios	Functions								
	1	3	8	15	18	21	26	30	42

1	0	6.19E-197	1.27E-97	4.89E-194	2	5.22E-05	1.28E-195	0.18	3.65E-212
2	0	2.12E-196	7.39E-98	3.13E-195	2	8.31E-05	9.14E-196	0.18	7.21E-205
3	0	2.13E-197	6.63E-98	2.73E-192	2	6.21E-05	1.72E-195	0.18	6.58E-217
4	0	2.12E-196	1.29E-97	8.89E-193	2	0.000276	2.99E-195	0.18	5.04E-206
5	0	2.12E-196	6.78E-98	7.35E-193	2	0.000787	1.35E-195	0.18	4.38E-217

From the previous results it can be observed the high performance of the proposed QHGSO to find the optimal solution for the most tested functions. Moreover, in terms of diversity, the quantum behavior provides HGSO with a suitable tool to explore the areas of search space with high probability to having optimal point. It is worth noting that, this mechanism empowers the HGSO algorithm to reach better solutions in less time (iterations) among other algorithms. On the other hand, in terms of algorithm limitations, we should point to the variety of parameters to be tuned and its some random factors which effects on exploration performance of the proposed algorithm; although we enhance the exploration of original HGSO using quantum behavior in the updating position phase, but our proposed algorithm can be hybridized with other operators to boost its performance.

6. Applications of QHGSO in Classical Engineering Problems

We now investigate a typical three engineering design problems to show the ability of proposed approach in constrained optimization problems.

6.1. Welded Beam Design

The **Error! Reference source not found.** shows the schematic of this Welded Beam Design problem. The objective of this problem is to minimize the overall fabrication cost under various constrains. According to this figure, four different variables, including the width, length welded area, the depth, and the thickness are considered.

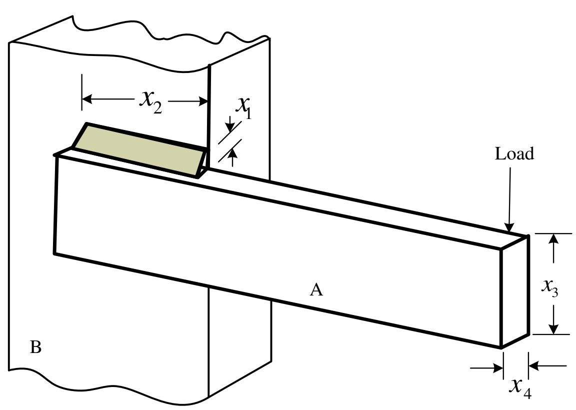

Figure 2: Welded beam design problem

The corresponding mathematical model of this problem is as follows:

$$Minimize\ f(\vec{x}) = 1.10471x_2x_1^2 + 0.04811x_3x_4(14.0 + x_2) \quad (1)$$

$$g_1(\vec{x}) = \tau(\vec{x}) - \tau_{max} \leq 0 \quad (2)$$

$$g_2(\vec{x}) = \sigma(\vec{x}) - \sigma_{max} \leq 0 \quad (3)$$

$$g_3(\vec{x}) = \delta(\vec{x}) - \delta_{max} \leq 0 \quad (4)$$
$$g_4(\vec{x}) = x_1 - x_4 \leq 0 \quad (5)$$
$$g_5(\vec{x}) = P - P_c(\vec{x}) \leq 0 \quad (6)$$
$$g_6(\vec{x}) = 0.125 - x_1 \leq 0 \quad (7)$$
$$g_7(\vec{x}) = 1.10471x_1^2 + 0.04811x_3x_4(14.0 + x_2) - 5.0 \leq 0 \quad (8)$$
$$0.10 \leq x_1 \leq 2.00, \quad (9)$$
$$0.10 \leq x_2 \leq 10.00, \quad (10)$$
$$0.10 \leq x_3 \leq 10.00, \quad (11)$$
$$0.10 \leq x_4 \leq 2.00, \quad (12)$$

where

$$\tau(\vec{x}) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2}$$

$$\tau' = \frac{P}{\sqrt{2}x_1x_2}, \tau'' = \frac{MR}{J}, M = P(L + \frac{x_2}{2})$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}$$

$$J = 2\left\{\sqrt{2}x_1x_2\left[\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\}$$

$$\sigma(\vec{x}) = \frac{6PL}{x_4x_3^2}, \delta(\vec{x}) = \frac{6PL^3}{Ex_4x_3^2}$$

$$P_c(\vec{x}) = \frac{4.013E\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)$$

$$P = 6000lb, L14\ in, \delta_{max} = 0.25\ in.,$$
$$E = 30 \times 10^6\ psi, G = 30 \times 10^6\ psi$$
$$\tau_{max} = 13600\ psi, \sigma_{max} = 30000\ psi$$

The objective function of this mathematical model is donated in Eq (1), the constraints are presented in in Eqs (2) - (8), and finally, variables are shown Eq. (9) - (12). There have been many studies that attempt to solve this problem. We here mention HS [79] , improved HS [80] Deb [81], CSS [82]MCSS [83], and ACO [84], for the comparison. Table 9 shows all results related to implementing proposed algorithm and other algorithms.

Table 9: Comparison of proposed methods for the welded beam design problem.

Algorithm	Optimum variables				Optimum cost
	x_1	x_2	x_3	x_4	
QHGSO	0.2152	6.8989	8.8150	0.2160	2.2864
HS	0.2442	6.2231	8.2915	0.2443	2.3807
Improved HS	0.4575	4.7313	5.0853	0.6600	4.1185
GA (Deb)	0.2489	6.1730	8.1789	0.2533	2.4331
CSS	0.2792	5.6256	7.7512	0.2796	2.5307
MCSS	0.2434	6.2552	8.2915	0.2444	2.3841
ACO	0.2444	6.2189	8. 2915	0.2444	2.3815

With respect to obtained results illustrated in Table 9, QHGSO can obtain the optimal result in comparison with others.

6.2. Tension/compression Spring Design

As shown in Figure 3, the second constrained engineering problem, named Tension/compression spring design, has three variables with the objective function of minimizing deflection. We have also different kinds of constraints, including surge frequency, and shear stress.

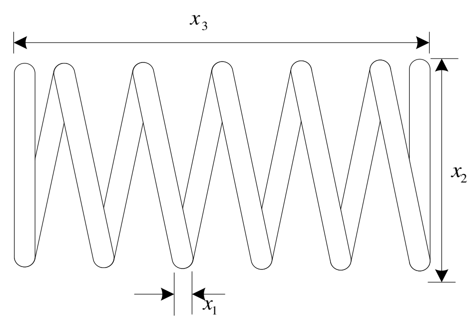

Figure 3: Tension/compression spring design

The mathematical model of this problem can be defined as follows:

$$Minimize\ f(\vec{x}) = (x_3 + 2)x_2x_1^2 \quad (13)$$

$$g_1(\vec{x}) = 1 - \frac{x_2^3x_3}{71785x_1^4} \leq 0 \quad (14)$$

$$g_2(\vec{x}) = \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} - 1 \leq 0 \quad (15)$$

$$g_3(\vec{x}) = 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0 \quad (16)$$

$$g_4(\vec{x}) = \frac{x_1 + x_2}{1.5} - 1 \leq 0 \quad (17)$$

$$0.05 \leq x_1 \leq 2.00, \quad (18)$$

$$0.25 \leq x_2 \leq 1.30, \quad (19)$$

$$2.00 \leq x_3 \leq 15.00, \quad (20)$$

We use a set of various methods, which are selected from [73], Coello [85], Kaveh and Talathari [84], He and Wang [86], Mahdavi [80] Kaveh [87] to show validity of performance proposed approach. The comparison results of implementing different approaches are shown in Table 10. The results show that VPL has been able to find very good results compared to the others.

Table 10: Comparison of proposed methods for Tension/compression spring design

Algorithm	Optimum variables			Optimum cost
	x_1	x_2	x_3	
QHGSO	0.05019	0.3316	12.8342	0.01239
GA (Coello)	0.05140	0.3516	11.6322	0.01270
ES	0.05190	0.3639	10.8905	0.01268
ACO	0.05180	0.3615	11.000	0.01264
DE	0.05160	0.3547	11.4100	0.01267
WOA	0.05120	0.3452	12.00400	0.01261
IHS	0.05110	0.3498	12.0764	0.01267

6.3. Pressure Vessel Design

The last constrained problem that we investigate in this study is Pressure Vessel Design. The objective of this problem is to minimize the total cost, including welding, forming and materials. Figure 4 shows the general schematic of this problem that include three different variables.

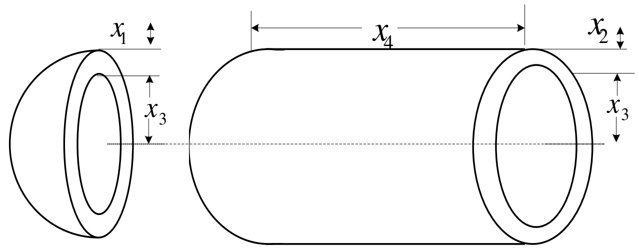

Figure 4: Pressure vessel design and its features

A typical form of pressure vessel design optimization problem can be expressed as follows:

$$Minimize\ f(\vec{x}) = 0.62224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_4x_1^2 + +19.84x_3x_1^2 \tag{21}$$
$$g_1(\vec{x}) = -x_1 + 0.0193x_3 \leq 0 \tag{22}$$
$$g_2(\vec{x}) = -x_3 + 0.00954x_3 \leq 0 \tag{23}$$
$$g_3(\vec{x}) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0 \tag{24}$$
$$g_4(\vec{x}) = x_4 - 240 \leq 0 \tag{25}$$
$$0 \leq x_1 \leq 99, \tag{26}$$
$$0 \leq x_2 \leq 99, \tag{27}$$
$$10 \leq x_3 \leq 200, \tag{28}$$
$$10 \leq x_4 \leq 200, \tag{29}$$

We implement various kinds of method, which inspired from the literature, to show the validation of proposed algorithm. These approaches can be mentioned as, [88], different genetic algorithm approaches, [85] ,[89], CPSO [86], and DE [90]. Table 11 illustrates the best obtained solutions from all proposed approaches. As clearly seen that proposed approach can obtain the best results in this problem, accordingly.

Table 11: Comparison of proposed approaches for Pressure vessel design problem.

Algorithm	Optimum variables				Optimum cost
	x_1	x_2	x_3	x_1	
QHGSO	0.8152	0.4265	42.09125	176.7423	6044.95
WOA	0.8125	0.4375	42.09829	176.6389	6059.74
CPSO	0.8125	0.4375	42.0912	176.7465	6061.07
GA(Coello)	0.8125	0.4375	40.3239	200.0000	6288.74
GA (Coello and Montes)	0.9375	0.4375	42.0973	176.6540	6059.94
DE	0.8125	0.4375	42.0984	176.6376	6059.73

7. Conclusions

In this study, a modified version of the Henry Gas Solubility Optimization is introduced. The HGSO is a recently developed metaheuristic algorithm which emulates the Henry's law. The proposed algorithm depends on applying the quantum theory for improving the behavior of exploring search space and the convergence of the solutions toward the global solution. In terms of performance measuring the obtained results proved that the QHGSO has high quality performance on test functions. The results are compared with a set of well-known MH algorithms including HGSO, SSA, GOA, WOA, DA, SCA, MFO, ALO and GWO. We have also investigated the performance of proposed algorithm in three constrained engineering problems. According to the experimental results, it can be concluded that the proposed QHGSO outperforms other MH methods in terms of average of fitness and the convergence curve.

Further research directions can be suggested in two approaches, namely methodological and application based as follows. In terms of the first approach, the QHGSO algorithm can be extended to the binary version using Q-bits. Furthermore, the multi-objective version of QHGSO is worth researching. Moreover, this algorithm can be hybridized with other algorithms to take their advantages to reach better performance in terms of exploration and exploitation. For the latter group, the following works can be implemented. The QHGSO can be adapted for solving the combinatorial optimization problems, including Vehicle Routing Problem, Scheduling Problem, Timetabling Problem, and Location Problem. It is worth noting that, the QHGSO algorithm also has the ability to use in other fields such as Image Segmentation, Feature Selection, and Engineering Design problems.

Acknowledgements

We thank the Editor and anonymous reviewers for their constructive comments and suggestions, which helped us to improve the manuscript.

References

[1] C. Lee, Y.-F. Wang, and T. Yang, "Global Optimization for Mapping Parallel Image Processing Tasks on Distributed Memory Machines," *Journal of Parallel and Distributed Computing,* vol. 45, no. 1, pp. 29-45, 1997/08/25/ 1997.

[2] A. Candelieri and F. Archetti, "Global optimization in machine learning: the design of a predictive analytics application," *Soft Computing,* vol. 23, no. 9, pp. 2969-2977, 2019/05/01 2019.

[3] N. N. Srinidhi, S. M. Dilip Kumar, and K. R. Venugopal, "Network optimizations in the Internet of Things: A review," *Engineering Science and Technology, an International Journal,* vol. 22, no. 1, pp. 1-21, 2019/02/01/ 2019.

[4] M. Boix, L. Montastruc, C. Azzaro-Pantel, and S. Domenech, "Optimization methods applied to the design of eco-industrial parks: a literature review," *Journal of Cleaner Production,* vol. 87, pp. 303-317, 2015/01/15/ 2015.

[5] A. Juels and M. Wattenberg, "Stochastic hillclimbing as a baseline method for evaluating genetic algorithms," in *Advances in Neural Information Processing Systems*, 1996, pp. 430-436.

[6] T. A. Feo and M. G. Resende, "Greedy randomized adaptive search procedures," *Journal of global optimization,* vol. 6, no. 2, pp. 109-133, 1995.

[7] C. Voudouris and E. P. Tsang, "Guided local search," in *Handbook of metaheuristics*: Springer, 2003, pp. 185-218.
[8] N. Baba, T. Shoman, and Y. Sawaragi, "A modified convergence theorem for a random optimization method," *Information Sciences,* vol. 13, no. 2, pp. 159-166, 1977.
[9] E. K. Burke, G. Kendall, and E. Soubeiga, "A tabu-search hyperheuristic for timetabling and rostering," *Journal of heuristics,* vol. 9, no. 6, pp. 451-470, 2003.
[10] H. Lourenзo, "A beginner's introduction to iterated local search," 2001.
[11] N. Mladenović and P. Hansen, "Variable neighborhood search," *Computers & operations research,* vol. 24, no. 11, pp. 1097-1100, 1997.
[12] D. E. Goldberg and J. H. Holland, "Genetic algorithms and machine learning," 1988.
[13] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies–A comprehensive introduction," *Natural computing,* vol. 1, no. 1, pp. 3-52, 2002.
[14] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Transactions on Evolutionary computation,* vol. 3, no. 2, pp. 82-102, 1999.
[15] L. Cui, G. Li, Q. Lin, J. Chen, and N. Lu, "Adaptive differential evolution algorithm with novel mutation strategies in multiple sub-populations," *Computers & Operations Research,* vol. 67, pp. 155-173, 2016.
[16] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization,* vol. 11, no. 4, pp. 341-359, 1997.
[17] C. Ferreira, "Gene expression programming: a new adaptive algorithm for solving problems," *arXiv preprint cs/0102027,* 2001.
[18] Y.-J. Zheng, "Water wave optimization: a new nature-inspired metaheuristic," *Computers & Operations Research,* vol. 55, pp. 1-11, 2015.
[19] B. Javidy, A. Hatamlou, and S. Mirjalili, "Ions motion algorithm for solving optimization problems," *Applied Soft Computing,* vol. 32, pp. 72-79, 2015.
[20] A. Sadollah, A. Bahreininejad, H. Eskandar, and M. Hamdi, "Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems," *Applied Soft Computing,* vol. 13, no. 5, pp. 2592-2612, 2013.
[21] S. D. Muller, J. Marchetto, S. Airaghi, and P. Kournoutsakos, "Optimization based on bacterial chemotaxis," *IEEE transactions on Evolutionary Computation,* vol. 6, no. 1, pp. 16-29, 2002.
[22] R. V. Rao, V. J. Savsani, and D. Vakharia, "Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems," *Computer-Aided Design,* vol. 43, no. 3, pp. 303-315, 2011.
[23] R. Eberhart and J. Kennedy, "Particle swarm optimization," in *Proceedings of the IEEE international conference on neural networks*, 1995, vol. 4, pp. 1942-1948: Citeseer.
[24] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software,* vol. 69, pp. 46-61, 3// 2014.
[25] R. Moghdani and K. Salimifard, "Volleyball Premier League Algorithm," *Applied Soft Computing,* vol. 64, pp. 161-185, 2018/03/01/ 2018.
[26] A. Kaveh and N. Farhoudi, "A new optimization method: Dolphin echolocation," *Advances in Engineering Software,* vol. 59, pp. 53-70, 2013.
[27] E. Duman, M. Uysal, and A. F. Alkaya, "Migrating Birds Optimization: A new metaheuristic approach and its performance on quadratic assignment problem," *Information Sciences,* vol. 217, pp. 65-77, 2012.

[28] G.-G. Wang, S. Deb, and L. d. S. Coelho, "Elephant herding optimization," in *2015 3rd International Symposium on Computational and Business Intelligence (ISCBI)*, 2015, pp. 1-5: IEEE.
[29] M. Dorigo and G. Di Caro, "Ant colony optimization: a new meta-heuristic," in *Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406)*, 1999, vol. 2, pp. 1470-1477: IEEE.
[30] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," *Engineering with computers,* vol. 29, no. 1, pp. 17-35, 2013.
[31] D. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi, "The bees algorithm," *Technical Note, Manufacturing Engineering Centre, Cardiff University, UK,* 2005.
[32] J. J. Q. Yu and V. O. K. Li, "A social spider algorithm for global optimization," *Applied Soft Computing,* vol. 30, pp. 614-627, 2015/05/01/ 2015.
[33] M. Eusuff, K. Lansey, and F. Pasha, "Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization," *Engineering optimization,* vol. 38, no. 2, pp. 129-154, 2006.
[34] S.-C. Chu, P.-W. Tsai, and J.-S. Pan, "Cat swarm optimization," in *Pacific Rim international conference on artificial intelligence*, 2006, pp. 854-858: Springer.
[35] X.-S. Yang, "Firefly algorithms for multimodal optimization," in *International symposium on stochastic algorithms*, 2009, pp. 169-178: Springer.
[36] D. Karaboga and C. Ozturk, "A novel clustering approach: Artificial Bee Colony (ABC) algorithm," *Applied Soft Computing,* vol. 11, no. 1, pp. 652-657, 2011/01/01/ 2011.
[37] R. P. Feynman, "Quantum mechanical computers," *Foundations of Physics,* vol. 16, no. 6, pp. 507-531, 1986/06/01 1986.
[38] A. Narayanan and M. Moore, "Quantum-inspired genetic algorithms," in *Proceedings of IEEE international conference on evolutionary computation*, 1996, pp. 61-66: IEEE.
[39] K. H. Han, "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization," *IEEE Trans. Evolutionary Computation,* vol. 6, no. 6, pp. 580-593, 2002 2002.
[40] F. A. Hashim, E. H. Houssein, M. S. Mabrouk, W. Al-Atabany, and S. Mirjalili, "Henry gas solubility optimization: A novel physics-based algorithm," *Future Generation Computer Systems,* vol. 101, pp. 646-667, 2019/12/01/ 2019.
[41] F. A. Hashim, E. H. Houssein, K. Hussain, M. S. Mabrouk, and W. Al-Atabany, "A modified Henry gas solubility optimization for solving motif discovery problem," *Neural Computing and Applications,* pp. 1-13, 2019.
[42] A. Biswas, K. Mishra, S. Tiwari, and A. Misra, "Physics-inspired optimization algorithms: a survey," *Journal of Optimization,* vol. 2013, 2013.
[43] J. Cao and H. Gao, "A quantum-inspired bacterial swarming optimization algorithm for discrete optimization problems," in *International Conference in Swarm Intelligence*, 2012, pp. 29-36: Springer.
[44] L. Jiao, Y. Li, M. Gong, and X. Zhang, "Quantum-inspired immune clonal algorithm for global optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics),* vol. 38, no. 5, pp. 1234-1253, 2008.
[45] P. Li and S. Li, "Quantum ant colony algorithm for continuous space optimization," *Control Theory and Applications,* vol. 25, no. 2, pp. 237-241, 2008.

[46] J. Sun, W. Xu, and B. Feng, "A global search strategy of quantum-behaved particle swarm optimization," in *IEEE Conference on Cybernetics and Intelligent Systems, 2004.*, 2004, vol. 1, pp. 111-116: IEEE.
[47] R. Zhang and H. Gao, "Improved quantum evolutionary algorithm for combinatorial optimization problem," in *2007 International Conference on Machine Learning and Cybernetics*, 2007, vol. 6, pp. 3501-3505: IEEE.
[48] Y. Wang *et al.*, "A novel quantum swarm evolutionary algorithm and its applications," *Neurocomputing,* vol. 70, no. 4-6, pp. 633-640, 2007.
[49] Y.-k. Zhang, J.-c. Liu, Y.-a. Cui, X.-h. Hei, and M.-h. Zhang, "An improved quantum genetic algorithm for test suite reduction," in *2011 IEEE International Conference on Computer Science and Automation Engineering*, 2011, vol. 2, pp. 149-153: IEEE.
[50] M. D. Platel, S. Schliebs, and N. Kasabov, "A versatile quantum-inspired evolutionary algorithm," in *2007 IEEE Congress on Evolutionary Computation*, 2007, pp. 423-430: IEEE.
[51] A. Kaveh and S. Talatahari, "A novel heuristic optimization method: charged system search," *Acta Mechanica,* vol. 213, no. 3, pp. 267-289, 2010/09/01 2010.
[52] Ş. İ. Birbil and S.-C. Fang, "An electromagnetism-like mechanism for global optimization," *Journal of global optimization,* vol. 25, no. 3, pp. 263-282, 2003.
[53] O. K. Erol and I. Eksin, "A new optimization method: Big Bang–Big Crunch," *Advances in Engineering Software,* vol. 37, no. 2, pp. 106-111, 2006/02/01/ 2006.
[54] H. Shah-Hosseini, "Principal components analysis by the galaxy-based search algorithm: a novel metaheuristic for continuous optimisation," *International Journal of Computational Science and Engineering,* vol. 6, no. 1-2, pp. 132-140, 2011.
[55] J. J. Flores, R. López, and J. Barrera, "Gravitational interactions optimization," in *International Conference on Learning and Intelligent Optimization*, 2011, pp. 226-237: Springer.
[56] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: A Gravitational Search Algorithm," *Information Sciences,* vol. 179, no. 13, pp. 2232-2248, 2009/06/13/ 2009.
[57] L. Xie, Y. Tan, J. Zeng, and Z. Cui, "Artificial physics optimisation: a brief survey," *International Journal of Bio-Inspired Computation,* vol. 2, no. 5, pp. 291-302, 2010.
[58] R. A. Formato, "Central force optimization," *Prog Electromagn Res,* vol. 77, pp. 425-491, 2007.
[59] A. Hatamlou, "Black hole: A new heuristic optimization approach for data clustering," *Information Sciences,* vol. 222, pp. 175-184, 2013/02/10/ 2013.
[60] A. Kaveh and M. Khayatazad, "A new meta-heuristic method: Ray Optimization," *Computers & Structures,* vol. 112-113, pp. 283-294, 2012/12/01/ 2012.
[61] H. Du, X. Wu, and J. Zhuang, "Small-World Optimization Algorithm for Function Optimization," ed.
[62] B. Alatas, "ACROA: Artificial Chemical Reaction Optimization Algorithm for global optimization," *Expert Systems with Applications,* vol. 38, no. 10, pp. 13170-13180, 2011/09/15/ 2011.
[63] A. Kaveh, M. Kamalinejad, and H. Arzani, "Quantum evolutionary algorithm hybridized with Enhanced colliding bodies for optimization," in *Structures*, 2020, vol. 28, pp. 1479-1501: Elsevier.

[64] A. Kaveh, H. Akbari, and S. M. Hosseini, "Plasma generation optimization: a new physically-based metaheuristic algorithm for solving constrained optimization problems," *Engineering Computations,* 2020.
[65] P. J. M. van Laarhoven and E. H. L. Aarts, "Simulated annealing," in *Simulated Annealing: Theory and Applications*, P. J. M. van Laarhoven and E. H. L. Aarts, Eds. Dordrecht: Springer Netherlands, 1987, pp. 7-15.
[66] M. Abdechiri, M. R. Meybodi, and H. Bahrami, "Gases Brownian Motion Optimization: an Algorithm for Optimization (GBMO)," *Applied Soft Computing,* vol. 13, no. 5, pp. 2932-2946, 2013/05/01/ 2013.
[67] T. L. Brown, *Chemistry: the central science*. Pearson Education, 2009.
[68] M. Mastrolilli and L. M. Gambardella, "Effective neighbourhood functions for the flexible job shop problem," *Journal of scheduling,* vol. 3, no. 1, pp. 3-20, 2000.
[69] L. dos Santos Coelho, "A quantum particle swarm optimizer with chaotic mutation operator," *Chaos, Solitons & Fractals,* vol. 37, no. 5, pp. 1409-1418, 2008.
[70] M. R. Singh and S. S. Mahapatra, "A quantum behaved particle swarm optimization for flexible job shop scheduling," *Computers & Industrial Engineering,* vol. 93, pp. 36-44, 2016.
[71] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," *Advances in Engineering Software,* vol. 114, pp. 163-191, 2017/12/01/ 2017.
[72] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper Optimisation Algorithm: Theory and application," *Advances in Engineering Software,* vol. 105, pp. 30-47, 2017/03/01/ 2017.
[73] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software,* vol. 95, pp. 51-67, 2016/05/01/ 2016.
[74] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-Verse Optimizer: a nature-inspired algorithm for global optimization," *Neural Computing and Applications,* vol. 27, no. 2, pp. 495-513, 2016/02/01 2016.
[75] S. Mirjalili, "SCA: A Sine Cosine Algorithm for solving optimization problems," *Knowledge-Based Systems,* vol. 96, pp. 120-133, 2016/03/15/ 2016.
[76] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowledge-Based Systems,* vol. 89, pp. 228-249, 2015/11/01/ 2015.
[77] S. Mirjalili, "The Ant Lion Optimizer," *Advances in Engineering Software,* vol. 83, pp. 80-98, 2015/05/01/ 2015.
[78] R. Woolson, "Wilcoxon signed - rank test," *Wiley encyclopedia of clinical trials,* pp. 1-3, 2007.
[79] K. S. Lee and Z. W. Geem, "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice," *Computer methods in applied mechanics and engineering,* vol. 194, no. 36, pp. 3902-3933, 2005.
[80] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems," *Applied mathematics and computation,* vol. 188, no. 2, pp. 1567-1579, 2007.
[81] K. Deb, "Optimal design of a welded beam via genetic algorithms," *AIAA journal,* vol. 29, no. 11, pp. 2013-2015, 1991.
[82] A. Kaveh and S. Talatahari, "Optimal design of skeletal structures via the charged system search algorithm," *Structural and Multidisciplinary Optimization,* vol. 41, no. 6, pp. 893-911, 2010.

[83] A. Kaveh, M. Motie Share, and M. Moslehi, "A new meta-heuristic algorithm for optimization: magnetic charged system search," *Acta Mech,* vol. 224, no. 1, pp. 85-107, 2013.

[84] A. Kaveh and S. Talatahari, "An improved ant colony optimization for constrained engineering design problems," *Engineering Computations,* vol. 27, no. 1, pp. 155-182, 2010.

[85] C. A. C. Coello, "Use of a self-adaptive penalty approach for engineering optimization problems," *Computers in Industry,* vol. 41, no. 2, pp. 113-127, 2000.

[86] Q. He and L. Wang, "An effective co-evolutionary particle swarm optimization for constrained engineering design problems," *Engineering Applications of Artificial Intelligence,* vol. 20, no. 1, pp. 89-99, 2007.

[87] A. Kaveh and M. Khayatazad, "A new meta-heuristic method: ray optimization," *Computers & Structures,* vol. 112, pp. 283-294, 2012.

[88] B. Kannan and S. N. Kramer, "An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design," *Journal of mechanical design,* vol. 116, no. 2, pp. 405-411, 1994.

[89] C. A. C. Coello and E. M. Montes, "Constraint-handling in genetic algorithms through the use of dominance-based tournament selection," *Advanced Engineering Informatics,* vol. 16, no. 3, pp. 193-203, 2002.

[90] L. Li, Z. Huang, F. Liu, and Q. Wu, "A heuristic particle swarm optimizer for optimization of pin connected structures," *Computers & Structures,* vol. 85, no. 7, pp. 340-349, 2007.