
*Research article***A Quantum-Inspired Sperm Motility Algorithm****Ibrahim M. Hezam^{1,*}, Osama Abdul-Raof², Abdelaziz Foul¹ and Faisal Aqlan³**

¹ Statistics and Operations Research Department, College of Sciences, King Saud University, Riyadh 11451, Saudi Arabia

² Operations Research and Decision Support Department, Faculty of Computers and Information, Menoufia University, Menoufia, Egypt

³ Industrial Engineering in the School of Engineering, The Behrend College, The Pennsylvania State University, Erie, PA, 16563, USA

* **Correspondence:** Email: ialmishnanah@ksu.edu.sa.

Abstract: Sperm Motility Algorithm (SMA), inspired by the human fertilization process, was proposed by Abdul-Raof and Hezam [1] to solve global optimization problems. Sperm flow obeys the Stokes equation or the Schrödinger equation as its derived equivalent. This paper combines a classical SMA with quantum computation features to propose two novel Quantum-Inspired Evolutionary Algorithms: The first is called the Quantum Sperm Motility Algorithm (QSMA), and the second is called the Improved Quantum Sperm Motility Algorithm (IQSMA). The IQSMA is based on the characteristics of QSMA and uses an interpolation operator to generate a new solution vector in the search space. The two proposed algorithms are global convergence guaranteed population-based optimization algorithms, which outperform the original SMA in terms of their search-ability and have fewer parameters to control. The two proposed algorithms are tested using thirty-three standard dissimilarities benchmark functions. Performance and optimization results of the QSMA and IQSMA are compared with corresponding results obtained using the original SMA and those obtained from three state-of-the-art metaheuristics algorithms. The algorithms were tested on a series of numerical optimization problems. The results indicate that the two proposed algorithms significantly outperform the other presented algorithms.

Keywords: quantum computation; sperm motility; interpolation; metaheuristic; optimization

Mathematics Subject Classification: 68T20, 90C26

1. Introduction

Quantum computation (QC) is a new research field that has attracted the attention of many researchers in the past decade. Quantum computation is based on several principles of quantum theory, such as the qubit, the superposition of quantum states, the collapsing into one state, and entanglement [2,3]. Several review studies reviewed quantum computation computing methods, such as [4] which reviewed quantum computing methods, and [5] reviewed the important quantum supremacy and theoretical challenges. The authors in [6] addressed the scalable model for a distributed quantum computation. At the same time, a programmable superconducting processor was used to Quantum supremacy in [7]. In [8], Quantum Computing was discussed in the Noisy Intermediate-Scale Quantum era. Aaronson, S. and Chen, L in [9] reviewed some theoretical foundations for quantum computers as evidence of quantum supremacy. The challenges of building quantum computers and their applications were discussed in [10]. The authors in [11] addressed the challenges of quantum information technology such as quantum interconnects, quantum internet, superconducting, etc. Also, in [12], they proposed a framework of quantum neural networks based on supervised learning over classical and quantum data. Lloyd, S in [13] modified the quantum approximate optimization algorithm using two alternating Hamiltonians.

In quantum computing, a quantum algorithm is an algorithm that runs on a realistic model of quantum computation, which can be classified into two categories [2,3,14]. The first category comprises the quantum algorithms that are implemented on quantum computers [15–17], such as Grover's quantum search [18], quantum annealing [19], and the adiabatic quantum algorithm [20]. The second category encompasses the Quantum-Inspired Evolutionary Algorithms based on principles of quantum computation, including standing waves, entanglement, and collapse algorithms. The second category is divided into two sub-categories. The first one includes quantum computing-inspired evolutionary algorithms that are based on several postulates of quantum computing. Q-bits are used instead of binary representations. Superposition, quantum gates, and quantum measurements are applied to solve varied optimization problems. Several good examples of Quantum-Inspired Algorithms (QIA) are the genetic algorithm [21], the cuckoo search and firefly algorithm [22], Quantum-inspired immune clone optimization [23], the artificial bee colony algorithm [24], the gravitational search algorithm [25], the binary firefly algorithm [26], the space search algorithm [27], and the harmony search algorithm [28]. The second one embraces evolutionary algorithms based on quantum concepts. The experience of every particle is used in quantum space. An individual particle of the quantum system moves in quantum multi-dimensional space based on the state of a particle. It is depicted by the wave function $\psi(x, t)$ instead or by the position and velocity. Particle swarm optimization [29] and the gravitational search algorithm [25] are good examples of this type. Several previous studies established the quantum delta potential well-based model for several algorithms [30–34]. Moreover, the authors in [35] proposed a quantum-behaved gravitational search algorithm, where each candidate mass moves in a Delta potential well in the feasible space with a weighted average center of all k bests. In contrast, the authors in [36] proposed a novel version of the harmony search algorithm based on quantum concepts to solve the multi-objective optimizations problems. A similar study in [29], added a parameter to enhance the balance between the diversification and intensification of the quantum-behaved particle swarm optimization algorithm. Likewise, [37] used two methods to enhance a quantum particle swarm optimization based on Lévy flight and straight flight. The fruit fly optimization algorithm was improved based on a quantum Delta potential well [38]. The same, in [39] used quantum behaved to enhance fruit fly optimization algorithm. The authors in [40] hybridized the quantum evolutionary algorithm with the colliding bodies, then, they used delta potential well and

Schrodinger's uncertainty principle to enhance the proposed hybrid algorithm. A truncated mean stabilization strategy is used to improve the diversification of quantum PSO in [41]. Three metaheuristics algorithms, particle swarm optimization, genetic algorithm, and firefly algorithm, were enhanced using the quantum concepts in [42]. In a similar study [43], beetle antennae search was improved based on quantum behaved.

On the other hand, several studies used quadratic interpolations to improve the convergences, such as [44] employed the quadratic interpolations to improve the controlled random search algorithm. Also, [45] compared some versions of the controlled random search algorithm. Also, they proposed a novel modification of the original algorithm to improve its efficiency based on the quadratic interpolations. While [46] improved the exploitation process of the whale optimization algorithm using quadratic interpolations. A similar study in [47] extended a self-organizing migrating algorithm using quadratic interpolations. The authors in [48] hybridized the colliding bodies optimization Morlet wavelet mutation and quadratic interpolation. Furthermore, the authors in [49] proposed a quadratic interpolation method to speed the convergence and improve the intensification performance of the whale optimization algorithm.

On the other hand, the Sperm Motility Algorithm SMA [1] was inspired by the human fertilization process. In the original SMA, Stokes equations were chosen as the best mathematical model for simulating sperm flow. A heuristic mechanism for sperm, guided by chemoattractant ovum secretion, was mathematically modeled so that the moving agent (sperm) approached the goal (ovum).

In order to further improve efficiency and global search capability and based on the fact that the Schrödinger equation can be derived from the Stokes equation, we can replace the Stokes equation with the time-independent Schrodinger equation. Then, the Delta-potential well function is used to solve the time-independent Schrodinger equation.

Hence, the delta potential well-based QSMA is then proposed. In this algorithm, the state of sperm is described by the wave function $\psi(x, t)$ in place of the Stokes mathematical model.

However, the QSMA algorithm still needs to enhance convergence acceleration and more accurate solutions while solving global optimization problems.

Therefore, the intensification ability and fast convergence of QSMA will be improved by adding a quadratic interpolation operator to the position update process.

Hence, we can summarize the contribution of this work through the following main points:

- We derived the equivalent of the Schrödinger equation of Stokes equation.
- We proposed a new algorithm, QSMA, which is based on a hybrid Sperm Motility Algorithm and a quantum delta potential well-based.
- We improved the QSMA by using a quadratic interpolation operator for more balanced between the exploitation and exploration process, which leads to more accuracy and quality of solutions.

This paper is organized as follows: In Section 2, an overview of quantum computing is introduced. In Section 3, the standard SMA is described. The proposed algorithms are introduced in Section 4. All benchmark examples and results are presented in Section 5. Finally, the paper is concluded in Section 6.

2. Overview of quantum computing

2.1. Q-bit definition

The quantum bit (Q-bit) is the smallest unit of information that can be stored in a two-state quantum computer [50]. Quantum computers work based on the manipulation of Q-bits and the

application of quantum logic gates. Three states of a Q-bit can be presented by “1”, or “0”, or any superposition of “1”, and “0”. It is denoted by $|\psi\rangle$ and defined as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad (1)$$

where α and β are complex numbers that refer to the probability amplitudes of the states “0” and “1”, respectively. The values $|\alpha|^2$ and $|\beta|^2$ are the probabilities that the Q-bit states that are “0” or “1,” respectively. The equation used for the normalization is:

$$|\alpha|^2 + |\beta|^2 = 1. \quad (2)$$

Therefore, a Q-bit is a pair of numbers (α, β) , denoted by

$$q = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}. \quad (3)$$

2.2. Quantum superposition

In a typical quantum system, n Q-bits are employed to represent an agent, as follows:

$$q = [q^1, q^2, \dots, q^n] = \begin{bmatrix} \alpha^1 & \alpha^2 & \dots & \alpha^n \\ \beta^1 & \beta^2 & \dots & \beta^n \end{bmatrix} \quad (4)$$

where $|\alpha^d|^2 + |\beta^d|^2 = 1$, for $d = 1, 2, \dots, n$. In a quantum system, $|\psi_n\rangle$, an arbitrary superposition of states up to 2^n different states exists and is given by:

$$|\psi_n\rangle = \sum_{i=1}^{2^n} C_i |S_i\rangle \quad (5)$$

where C_i is the probability amplitude of state i , and S_i is the binary string (x^1, x^2, \dots, x^n) , $x^d \in \{0, 1\}$, $d = 1, 2, \dots, n$. In contrast, only one state of these 2^n states can exist in a standard computer at any time. The probabilistic superposition of 0 and 1 are used to represent a Q-bit. Moreover, this representation can be extended to multi-Q-bits systems.

2.3. Quantum measurement

The probabilistic observation process for measuring the d^{th} Q-bit is presented in the following pseudo-code [50]. Here, a random number is generated by a uniformly distributed probability function in the range $[0, 1]$, as follows:

$$\begin{aligned} & \text{If } rand[0, 1] < (\alpha^d)^2, \\ & \quad \text{then } \alpha^d = 0, \\ & \quad \text{Else } \alpha^d = 1. \end{aligned}$$

The observation process is in QIAs.

3. Overview of Sperm Motility Algorithm

The Sperm Motility Algorithm (SMA) [1] is a metaheuristic algorithm that mimics the fertilization process in humans. The following rules are assumed in the search process:

- (1) All sperms are attracted to the ovum due to the secreted chemoattractant.
- (2) Attractiveness is proportional to chemoattractant concentration.
- (3) Type A sperm, the highest quality sperm, will be selected for the next generation.

(4) The ovum is penetrated by the best sperm. In the case of fraternal twins, the ovum is penetrated by more than one sperm. This case suits the multi-objective optimization.

(5) More than 250 million sperm move randomly with the velocity v_i at position x_i toward the ovum. The Stokes equation is used to describe the sperm motility [51, 52] as follows:

$$Re \left(\frac{\partial v}{\partial t} + v \cdot \nabla v \right) + \nabla p = \mu \nabla^2 v + f \quad (6)$$

$$\nabla \cdot v = 0, x \in \Omega$$

where p is the pressure (including the gravitational potential), μ is the kinematic viscosity, $f(x) = \delta(x - \xi)F$ is the force density, and v is the velocity vector field in the domain Ω .

For a micro-swimmer such as sperm, the Reynolds number (Re) is approximately 0.01 and can be neglected. Hence, the Stokes equation can be transformed to its linearized form (Navier-Stokes equations), as follows:

$$\nabla p = \mu \nabla^2 v + f \quad (7)$$

$$\nabla \cdot v = 0, x \in \Omega.$$

The velocity solution corresponding to this fundamental singularity is given as follows:

$$\begin{aligned} v_i(t) &= \left(\frac{1}{8\pi\mu} \right) \left(\frac{\delta_{ij}}{r} + \frac{r_i r_j}{r^3} \right) F_j \\ &= \left(\frac{1}{8\pi\mu} \right) S_{ij}(x, \xi) F_j; i, j = 1, 2, 3, \end{aligned} \quad (8)$$

where $S_{ij}(x, \xi) = \left(\frac{\delta_{ij}}{r} + \frac{r_i r_j}{r^3} \right)$ is known as the Stokeslet that is the solution of the Stokes flow equations, or the Oseen-Burgers tensor, δ_{ij} is the Dirac delta distribution centered at ζ , the flow is due to a force F_j concentrated at the point ζ , r is dimensionless radial length in two dimensions, and $r_i = x_i - \xi_i$, $r^2 = |x - \xi|^2 = r_1^2 + r_2^2 + r_3^2$.

The position is updated as follows:

$$x_{i+1}(t) = x_i(t) + \left(\frac{\delta t}{2} \right) (v_{i+1}(t) + v_i(t)) + \beta(x_i(t) - g^*). \quad (9)$$

The nonlinear spatial chemoattractant concentration gradient field is given by:

$$c_i(t) = c_0(t) + c_1(\|g^* - x_i(t)\|)^{-b} \quad (10)$$

where $c(t)$ is the concentration, $x(t)$ is the position, c_l is the proportion coefficient, b is the power of the significant term position, c_0 represents the initial concentration and g^* is the current best solution found among all solutions at the current generation/iteration.

The basic steps of the SMA can be summarized by the pseudo code shown below.

Algorithm 1: The original Sperm Motility Algorithm

Begin

Define objective function $f(x)$, $x = (x_1, x_2, \dots, x_d)^T$

initialize N sperm population size.

generate initial position x_0 and velocity v_0 and initial concentration c_0 of N sperm.

define all SMA parameters ($c_0, \beta, \mu \dots$ etc.).

while (stopping criterion not met).

for $i=1:N$ **do**

 calculate velocity v_i from data at $t = t_i$; Eq (8);

 update position x_i for sperm i from Eq (9);

 evaluate each sperm individual according to its position.

 if the new solution is better, update it in the population;

 calculate c_i from Eq (10).

if $c_i \leq c_{i-1}$ **then** neglect [Abandon a fraction (P_a) of worse sperm];

 Check constraints satisfactions.

end for

Sort the population/sperm from best to worst and find the current best.

end while

Post-processing the results and visualization.

End

4. The proposed Quantum-Inspired Sperm Motility Algorithm

4.1. Schrödinger wave representation of Navier-Stokes equation

Several studies have derived a Schrödinger equation equivalent to the Navier-Stokes equation [53–55]. The Schrödinger equation has been solved, while the exact solution of the Navier-Stokes equation remains an open problem in mathematical physics. Therefore, Schrödinger's equation is preferred to the Navier-Stokes equation. Moreover, the Schrödinger equation can describe the quantum state of the velocity field and defines the velocity potential.

Equation (6) can be rewritten as follows:

$$\rho \left(\frac{\partial v}{\partial t} + (v \cdot \nabla) v \right) + \nabla p = \xi \nabla \operatorname{div} v + \eta \Delta v. \quad (11)$$

If the viscosity coefficients ξ and η are linearly dependent of the density ρ , then,

$$\xi = \mu \rho \text{ and } \eta = \nu \rho,$$

where the kinematic viscosity coefficients of ξ and ν are constant. The solutions to Eq (11) can be found in the form, provided that the function χ satisfies the following equation:

$$\dot{\chi} + \frac{1}{2} (\nabla \chi)^2 + h(\rho) = (\mu + \nu) \Delta \chi \quad (12)$$

where $h(\rho)$ is the specific enthalpy. Based on [54], the Navier-Stokes equation (11) can be reduced to its equivalent Schrödinger equation. This could be written as follows:

$$j\hbar \frac{\partial \Psi}{\partial t} = \left\{ -\frac{\hbar^2}{2m} \Delta + V + \frac{\hbar}{2i} (\mu + \nu) \Delta \log \left(\frac{\Psi^*}{\Psi} \right) \right\} \Psi \quad (13)$$

where V is the potential energy, j is the imaginary unit, \hbar is Planck's constant, and Ψ is the wave function.

4.2. Quantum mechanics

In quantum mechanics, the changes in the quantum state through time can be described by the Schrödinger equation. In addition, the complex wave function $\Psi(X, t)$ can be used to depict the quantum state instead of using the position and velocity. The time-dependent Schrödinger equation (13) is the governing equation that can be expressed as:

$$j\hbar \frac{\partial}{\partial t} \Psi(X, t) = H(X)\Psi(X, t) \quad (14)$$

where $\Psi(X, t)$ is the wave function that has no physical meaning, but the square of its amplitude gives the probability of sperm position in anyone-dimension X at time t , and H is a time-independent Hamiltonian operator given as:

$$H(X) = -\frac{\hbar^2}{2m} \nabla^2 + V(X) \quad (15)$$

where ∇^2 is the Laplacian operator, m is the mass of the sperm, and $V(X)$ is the potential energy distribution. In the Schrödinger equation, the unknown is the wave function $\Psi(X, t)$. However, its amplitude squared, $|\Psi|^2$, is the probability of the agent's (the sperm in this case) motion. This relationship allows the probability of finding sperm in a particular region and computing the time t .

The collapsing concept is used to map the probability of finding a particular position in the quantum search space and its actual position in the solution space. Hence, the wave function $\Psi(X, t)$ in a 3-Dimensional space with coordinates x_1, x_2, x_3 can be written as:

$$|\Psi|^2 dx_1 dx_2 dx_3 = Q dx_1 dx_2 dx_3, \quad (16)$$

where $Q dx_1 dx_2 dx_3$ is the probability of measurement of the sperm's position at time t in the 3-Dimensional space and $|\Psi|^2$ is then a probability density function that satisfies the condition:

$$\int_{-\infty}^{+\infty} |\Psi|^2 dX = \int_{-\infty}^{+\infty} Q dX = 1. \quad (17)$$

The integration is taken over the range of the entire feasible space. The separation of variables method is used to solve (14). The wave function becomes a product of the spatial term $\psi(X)$ and the temporal term $f(t)$,

$$\Psi(X, t) = \psi(X)f(t). \quad (18)$$

The time-independent equation is derived from the time-dependent equation by substituting (18) in (14),

$$j\hbar \psi(X) \frac{d}{dt} f(t) = f(t) \left(-\frac{\hbar^2}{2m} \nabla^2 + V(X) \right) \psi(X). \quad (19)$$

This equation can be rewritten as:

$$\frac{j\hbar}{f(t)} \cdot \frac{d}{dt} f(t) = \frac{1}{\psi(X)} \cdot \left(-\frac{\hbar^2}{2m} \nabla^2 + V(X) \right) \psi(X). \quad (20)$$

The left side of Eq (20) is a function of t , and the right side is a function of X . Thus, the two sides must be equal to some constant, called it E , which represents the energy of the sperm. Therefore, two differential equations can be derived from (20):

$$\frac{1}{f(t)} \cdot \frac{d}{dt} f(t) = -\frac{jE}{\hbar} \quad (21)$$

and

$$\frac{1}{\psi(X)} \cdot \left(-\frac{\hbar^2}{2m} \nabla^2 + V(X) \right) \psi(X) = E \rightarrow H\psi(X) = E\psi(X). \quad (22)$$

(21) is easily solved to yield:

$$f(t) = e^{-\frac{jEt}{\hbar}}. \quad (23)$$

Hence:

$$\Psi(X, t) = \psi(X) e^{-\frac{jEt}{\hbar}}, \quad (24)$$

$|\Psi|^2$ refers to the probability of finding sperm at a particular position in the quantum search space and is given by:

$$|\Psi(X, t)|^2 = \psi(X) e^{-\frac{jEt}{\hbar}} \psi^*(X) e^{\frac{jEt}{\hbar}} = \psi(X) \psi^*(X) = |\psi(X)|^2. \quad (25)$$

The probability density function $|\Psi(X, t)|^2$ is a time-independent function. Therefore, solving the time-independent Schrödinger equation (22) gives the wave function $\psi(X)$.

4.3. The quantum model of SMA

Now, suppose that each sperm has a quantum behavior and a wave function that describe the sperm state. For the sake of simplicity, a sperm in a quantum system with a 1-dimensional search space will be considered. In this case, the position vector X is reduced to scalar x .

Let

$$y = x - g \quad (26)$$

where g is a position that has a high chemoattractant concentration. To increase the convergence of QSMA, y should approach zero. Therefore, an attractive potential field centered at zero is applied. According to the analysis of convergence behavior of sperms in SMA, several forms of potential attraction centered around g must exist. Therefore, the δ potential well can be set at g and the most simple and effective expression of its potential energy function is the delta potential well,

$$V(y) = -\gamma\delta(y), \quad (27)$$

where $\delta(y)$ is the Dirac delta function. This function is called a delta potential well if γ is negative and a delta potential barrier if γ is positive. Here, γ is a positive number related to the “depth” of the potential well. The depth at the origin is infinite and is zero elsewhere. The Schrödinger equation is:

$$E\psi(y) = \left(-\frac{\hbar^2}{2m} \nabla^2 - \gamma\delta(y) \right) \psi(y) \quad (28)$$

$$\frac{\hbar^2}{2m} \cdot \frac{d^2}{dy^2} \psi(y) + \gamma\delta(y)\psi(y) + E\psi(y) = 0. \quad (29)$$

When $y \neq 0$, (29) can be written as:

$$\frac{d^2}{dy^2} \psi(y) + \frac{2m}{\hbar^2} E\psi = 0. \quad (30)$$

The following boundary condition is used for more convergence of sperm:

$$|y| \rightarrow \infty \leftrightarrow \psi \rightarrow 0. \quad (31)$$

Then, the solution to (30) is:

$$\psi(y) = e^{-\frac{\sqrt{-2mE}}{\hbar}|y|} \text{ when } y \neq 0. \quad (32)$$

In addition, with the normalization condition being satisfied.

$$\int_{-\infty}^{+\infty} |\psi|^2 dy = 1, \quad (33)$$

the probability density function Q will be as follows:

$$Q(y) = |\psi(y)|^2 = e^{-\frac{2\sqrt{-2mE}}{\hbar}|y|}. \quad (34)$$

The position and the velocity cannot be determined simultaneously according to the Heisenberg principle of uncertainty [56]. Thus, in the QSMA and IQSMA, the velocity and position terms will not exist together in the updated equations, compared to those in the standard SMA.

4.4. Measurement of the position

The positions of sperm in the SMA are the candidate solutions for a given optimization problem. And the position of the sperm in the QSMA can be measured by the Stokes equation, while the sperm itself follows quantum rules. To connect these workspaces, the quantum state needs to “collapse” the wave function of a moving sperm into the measurement space. The Monte Carlo Method (MCM) is used to achieve localization.

The quantum boundary of the sperm is guaranteed by the delta potential well. The position of the sperm should be determined to evaluate the fitness value. Since the quantum state function can calculate the probability density function, the sperm appears at the position y . Hence, we must calculate the position of the sperm, which is termed collapsing the quantum state to the classical state.

This measurement process is simulated by MCM, which generates a random variable. Equation (34) can be simplified to:

$$Q(y) = |\psi(y)|^2 = e^{-\frac{2\sqrt{-2mE}}{\hbar}|y|} = rand \quad (35)$$

or

$$-\frac{2\sqrt{-2mE}}{\hbar}|y| = \ln(rand), \quad (36)$$

that is

$$|y| = \frac{\hbar}{2\sqrt{-2mE}} \ln\left(\frac{1}{rand}\right) \quad (37)$$

$$y = x - g = \pm \frac{\hbar}{2\sqrt{-2mE}} \ln\left(\frac{1}{rand}\right). \quad (38)$$

Thus, the precise position of a sperm is measured by:

$$x = g \pm \frac{\hbar}{2\sqrt{-2mE}} \ln\left(\frac{1}{rand}\right). \quad (39)$$

It is supposed that $\frac{\hbar}{2\sqrt{-2mE}} = c|g - x|$, where c is a constant and $|g - x|$ is used to scale the variation of the newly created sperm around one of the ovum. Therefore, for an n -dimensional search space system, this equation can be written as follows:

$$\begin{cases} x_i^d(t+1) = g_i^d + c|g_i^d - x_i^d(t)| \cdot \ln\left(\frac{1}{rand}\right) & \text{if } S \geq 0.5 \\ x_i^d(t+1) = g_i^d - c|g_i^d - x_i^d(t)| \cdot \ln\left(\frac{1}{rand}\right) & \text{otherwise} \end{cases} \quad (40)$$

where r and S are two random numbers in the interval $(0, 1)$, c is a parameter used to balance the exploration and exploitation, and g_i^d is the current best position with the highest chemoattractant concentration.

4.5. The proposed QSMA

In this proposed algorithm, the position of each sperm is updated using the finally deducted Eq (40) in the sperm movement solution search. At the end of each generation, the new sperm is accepted if it gives a better fitness than the other sperm. The process is repeated until the stopping criterion is satisfied.

The steps of the proposed quantum Delta-Potential-well-based SMA Algorithm are as follows:

Algorithm 2: Quantum Sperm Motility Algorithm

Begin

Define objective function $f(x)$, $x = (x_1, x_2, \dots, x_d)^T$

initialize N sperm population size

generate initial position x_0 and velocity v_0 and initial concentration c_0 of N sperm.

define all QSMA parameters ($c_0, \beta, \mu \dots$ etc.)

while (stopping criterion not met);

for $i=1: N$ **do**

 generate s randomly from $[0,1]$

if $s \geq 0.5$

 update position x_i for sperm i using Eq (40);

$$x_i^d(t+1) = g_i^d + c|g_i^d - x_i^d(t)| \cdot \ln\left(\frac{1}{rand}\right)$$

Else

 update position x_i for sperm i using Eq (40);

$$x_i^d(t+1) = g_i^d - c|g_i^d - x_i^d(t)| \cdot \ln\left(\frac{1}{rand}\right)$$

 evaluate each sperm individual according to its position.

if the new solution is better, update it in the population

 calculate c_i from Eq (14)

if $c_i \leq c_{i-1}$ then neglect [Abandon a fraction (P_a) of worse sperm]

 Check constraints satisfactions

end for

Sort the population/sperm from best to worst and find the current best.

end while

Post-processing the results and visualization.

End

4.6. Improved QSMA with interpolation operator

In this method, three points in the current population are selected. x_l is obtained as the best optimization point (the sperms have a minimum fitness function value), where x_2, x_3 are randomly chosen [57]. The objective functions values are assumed as follows [58,59]:

$$f_i = f(x_i); i = 1, 2, 3.$$

Trial point design variables p_j are defined as the minimum of the parabola, written as follows:

$$q_j = \frac{1}{2} \frac{(x_{2j}^2 - x_{3j}^2)f_1 + (x_{3j}^2 - x_{1j}^2)f_2 + (x_{2j}^2 - x_{1j}^2)f_3}{(x_{2j} - x_{3j})f_1 + (x_{3j} - x_{1j})f_2 + (x_{2j} - x_{1j})f_3}, \quad (41)$$

$j = 1, 2, \dots, N$. To avoid quadratic interpolations that become ill-conditioned or present trial point design variables as maxima of the parabola (see Figure 1), a mean objective function value f_g and a local variability measure around the best point, α , are used and are usually calculated as follows:

$$f_g = \frac{1}{2}(f_2 + f_3) \quad (42)$$

$$\alpha = \frac{f_g - f_l}{f_h - f_l} \quad (43)$$

where f_h and f_l are the highest and lowest function values, respectively, these equations are used when quadratic interpolation is well-conditioned, and the best point is not contained between any other two points.

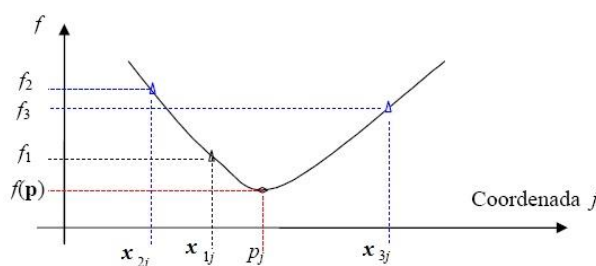


Figure 1. Graphical representation of quadratic interpolation.

In this case, a set of centroid design variables g_j are defined, as in Eq (44), and through these variables, the trial point design variables p_j , are defined by the variability-based reflection around the best point, as in (45) [57,60,61], as follows:

$$g_j = \frac{1}{2} \frac{(f_2 - f_1)x_{2j} + (f_3 - f_1)x_{3j}}{(f_2 - f_1) + (f_3 - f_1)} \quad (44)$$

$$p_j = (2 - \alpha)x_{1j} - (1 - \alpha)g_j. \quad (45)$$

4.7. The proposed IQSMA

Herein, a Controlled Random Search-Variability Based Reflections CRS-VBR Algorithm is used to generate a new sperm in the search space [57,60,61]. In the sperm movement solution search, the position of each sperm is updated using the Eq (40). After that, the interpolation operator is implemented by selecting another two sperm randomly and applying the Eqs (41–45). So, herein, the final position of each sperm is updated using the Eq (45). At the end of each generation, the new sperm is accepted if it is better than the other sperm. The process is repeated until a better solution is obtained

or for a certain number of generations. This method increases the diversity as promising new solution points are investigated because of the interpolation process. The new sperm positions are included in the domination comparison interpolation process. This methodology is very different from the usual method of retaining sperm with the best fitness function values for the recombination process. Moreover, for simplicity, we can evaluate the time complexity for both proposed algorithms based on the steps of algorithms 2 and 3. Both algorithms contain a number of primitive operations and two loops, so we can consider that the time complexity of both algorithms belongs to the quadratic class. Thus, the time complexity of both algorithms depends on the population's size and the number of iterations.

The computational steps of the improved algorithm are as follows:

Algorithm 3: Improved Quantum Sperm Motility Algorithm

Begin

Define objective function $f(x)$, $x = (x_1, x_2, \dots, x_d)^T$

initialize N sperm population size

generate initial position x_0 and velocity v_0 and initial concentration c_0 of N sperm.

define all QSMA parameters (c_0, β, μ ...etc.)

while (stopping criterion not met);

for $i=1: N$ **do**

 generate s randomly from $[0,1]$

if $s \geq 0.5$

 update position x_i for sperm i using Eq (40);

$$x_i^d(t+1) = g_i^d + c|g_i^d - x_i^d(t)| \cdot \ln\left(\frac{1}{rand}\right)$$

Else

 update position x_i for sperm i using Eq (40);

$$x_i^d(t+1) = g_i^d - c|g_i^d - x_i^d(t)| \cdot \ln\left(\frac{1}{rand}\right)$$

 Evaluate each sperm individual according to its position.

if the new solution is better, update it in the population

 select two sperm randomly

 apply the interpolation operator using Eqs (41–45)

 find a new sperm using Eq (45)

$$p_j = (2 - \alpha)x_{1j} - (1 - \alpha)g_j$$

if the new solution is better, update it in the population

 calculate c_i from Eq (14)

if $c_i \leq c_{i-1}$ then neglect [Abandon a fraction (P_a) of worse sperm]

 Check constraints satisfactions

end for

Sort the population/sperm from best to worst and find the current best.

end while

Post-processing the results and visualization.

End

5. Results and discussion

To investigate the performance of the proposed algorithms, we choose various functions that are widely used in the literature [1,62–65]. We classify the test benchmarks function into four groups where the first group represents unimodal benchmarks functions shown in Table 1. In contrast, the second group represents the multimodal benchmarks functions listed in Table 2. Also, Table 3 reports the fixed-dimension multimodal benchmarks functions, and Table 4 presents the CEC-C06 2019 test functions. Comparison between IQSMA and QSMA algorithms and SMA [1], EO [66], SCA [67], and SSA [68] algorithms were performed. The number of iterations and the population size of all algorithms were fixed at 500 and 30, respectively, to achieve fairness, and the algorithms were also implemented using MATLAB R2011 on a Core (TM) i3, 2.27 GHz processor. The simulation parameter settings that were the result of EO, SCA, SSA, and SMA algorithms are present in Table 5 as follows:

The unimodal benchmarks functions (F01–F07) are used to investigate the proposed algorithms' intensification, and the multimodal benchmarks functions (F08–F13) are employed to examine the proposed algorithms' diversification, while the fixed-dimension multimodal benchmarks functions (F14–F23) are used to test the ability of the proposed algorithms to scan the feasible region efficiently in low dimensions. Also, the CEC-C06 2019 test functions [65] are used to further evaluate the performance of the proposed algorithms. All present algorithms are compared using average, standard deviation, and elapsed time.

The numerical results presented in Table 6 show the objective function in terms of the average, standard deviation, and elapsed time. The best solution obtained for each test problem is shown in Tables 2–4; the numbers in bold indicate the best solution obtained according to mean fitness values and the convergence time. The results show that the performance of both proposed algorithms is superior in most benchmark functions, and they are competitors in some other benchmark functions compared to other present algorithms.

Table 1. The unimodal Benchmark functions.

ID	Formulation	Dimensions	Range	Global minimum
F01	$f(x) = \sum_{i=1}^n x_i^2$	30,100,500,1000	[-100,100]	0
F02	$f(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30,100,500,1000	[-10,10]	0
F03	$f(x) = \sum_{i=1}^d \left(\sum_{j=1}^i x_j \right)^2$	30,100,500,1000	[-100,100]	0
F04	$f(x) = \max_i \{ x_i , 1 \leq i \leq n\}$	30,100,500,1000	[-100,100]	0
F05	$f(x) = \sum_{i=1}^n [100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2]$	30,100,500,1000	[-30,30]	0
F06	$f(x) = \sum_{i=1}^n ([x_i + 0.5])^2$	30,100,500,1000	[-100,100]	0
F07	$f(x) = \sum_{i=1}^n ix_i^4 + \text{random } [0,1)$	30,100,500,1000	[-128,128]	0

Table 2. The multimodal Benchmark functions.

ID	Formulation	Dimensions	Range	Global minimum
F08	$f(x) = \sum_{i=1}^n (-x_i \sin(\sqrt{ x_i }))$	30,100,500,1000	[-500,500]	-418.9829 $\times n$
F09	$f(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30,100,500,1000	[-5.12,5.12]	0
F10	$f(x) = -20 \exp(-0.2 \sqrt{\frac{\sum_{i=1}^n x_i^2}{n}}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)) + 20 + e$	30,100,500,1000	[-32,32]	0
F11	$f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	30,100,500,1000	[-600,600]	0
F12	$f(x) = \frac{\pi}{n} \{10 \sin(\pi y_1)\} + \sum_{i=1}^n (y_i - 1)^2$ $+ \left[1 + 10 \sin^2(\pi y_{i+1}) + \sum_{i=1}^n u(x_i, 10, 100, 4) \right],$ <p>where</p> $y_i = 1 + \frac{x_i + 1}{4}, u(x_i, a, k, m)$ $= \begin{cases} K(x_i - a)^m & \text{if } x_i > a \\ 0 & \text{if } -a \leq x_i \leq a \\ K(-x_i - a)^m & \text{if } x_i < -a \end{cases}$	30,100,500,1000	[-50,50]	0
F13	$f(x) = 0.1 \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2$ $+ [1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2 (1 + \sin^2(3\pi x_n))$ $+ \sum_{i=1}^n u(x_i, 5, 100, 4)$	30,100,500,1000	[-50,50]	0

Table 3. The fixed-dimension multimodal benchmarks functions.

ID	Formulation	Dimensions	Range	Global minimum
F14	$f(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6} \right)^{-1}$	2	[-65,65]	1
F15	$f(x) = \sum_{i=1}^{11} \left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4} \right]^2$	4	[-5,5]	0.00030
F16	$f(x) = 4x_1^2 - 2.1x_1^4 + 0.33x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	[-5,5]	-1.0316
F17	$f(x) = \left(x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6 \right)^2 + 10 \left(1 - \frac{1}{8\pi} \right) \cos x_1$ $+ 10$	2	[-5,5]	0.398
F18	$f(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)]$ $\times [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	2	[-2,2]	3
F19	$f(x) = - \sum_{i=1}^4 c_i \exp \left(- \sum_{j=1}^3 a_{ij} (x_j - p_{ij})^2 \right)$	3	[-1,2]	-3.86
F20	$f(x) = - \sum_{i=1}^4 c_i \exp \left(- \sum_{j=1}^6 a_{ij} (x_j - p_{ij})^2 \right)$	6	[0,1]	-3.2
F21	$f(x) = - \sum_{i=1}^7 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,1]	-10.1532
F22	$f(x) = - \sum_{i=1}^7 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,1]	-10.4028
F23	$f(x) = - \sum_{i=1}^{10} [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,1]	-10.5363

Table 4. CEC-C06 2019 test functions [69].

ID	Formulation	Dimensions	Range	Global minimum
CEC01	Storn's Chebyshev Polynomial Fitting Problem	9	[-8192,8192]	1
CEC02	Inverse Hilbert Matrix Problem	16	[-16384,16384]	1
CEC03	Lennard-Jones Minimum Energy Cluster	18	[-4,4]	1
CEC04	Rastrigin's Function	10	[-100,100]	1
CEC05	Griewank's Function	10	[-100,100]	1
CEC06	Weierstrass Function	10	[-100,100]	1
CEC07	Modified Schwefel's Function	10	[-100,100]	1
CEC08	Expanded Schaffer's F6 Function	10	[-100,100]	1
CEC09	Happy Cat Function	10	[-100,100]	1
CEC10	Ackley's Function	10	[-100,100]	1

Table 5. Parameters of CS, FA, and PSO.

EO	Number of particles = 30
	Generation probability (GP) = 0.5
	$\alpha_1 = 2, \alpha_2 = 1$
SCA	Number of agents = 30
	Number of elites = 2
SSA	Leader position update probability = 0.5
SMA	$\mu = 10^{-3}$ Pa/s is the kinematic viscosity; F is the total force within the range (37-79 mW milliwatts); in Eq (2), set $c_0 = 10$ pM, b within the range (0.5-2).

Regarding the proposed algorithms' performance on the unimodal benchmark functions (F01–F07), as observed, IQAMA achieved the first rank in this category.

The superior performance of IQSMA is seen on average in functions of F01–F04, and F07. In F05, the superior performance was for the SCA algorithm, while the EO algorithm had the superior performance in F06. We can deduce the high exploitability of the proposed algorithms based on the results of the unimodal functions.

As for the performance of the proposed algorithms on the multimodal benchmark functions (F08–F013). As can be seen, IQSMA achieves the best performance in F08 F10. both IQSMA and EO reached the exact optimal solution in F09. Also, in F11, the three QSMA, SMA, and EO algorithms achieved the optimal solution. The EO algorithm obtained the best mean solution for the F12 function, while the best mean solution was obtained for the F13 function by SSA.

Regarding the fixed-dimension multimodal benchmarks functions (F14–F23), the best mean solution was reached by IQSMA, EO in F14. the performance of QSMA is the best in F15, F17. Four algorithms, IQSMA, QSMA, SMA, and SSA, got on the best mean solution in F16. Also, the IQSMA, QSMA, SMA, EO are equal in the performance in F18. While in F19, both proposed algorithms and the EO algorithm reached the best optimal solution. Similarly, in F21 and F22, the best solutions were obtained by both proposed algorithms and the SSA algorithm. IQSMA reached the best optimum in F20. In contrast, both proposed algorithms were equal in the best performance in the F23.

Moreover, we calculated the root of the sum of the squares of deviations from the optimal solutions mentioned in the Tables 2–4, we found that the rank of the IQSMA algorithm comes first, and then QSMA comes second, while the OE algorithm comes in the third rank, and the SMA occupies the fourth rank. SSA and SCA algorithms occupied the last places, respectively. When summing the computational times, we found that the IQSMA algorithm has the minimum computational time, then the SCA algorithm, after that, the SSA algorithm, the fourth was SMA, and the fifth was QSMA, while the EO algorithm is the last in terms of convergence speed. The convergence speed of the IQAMA algorithm is due to the importance of the interpolation operator.

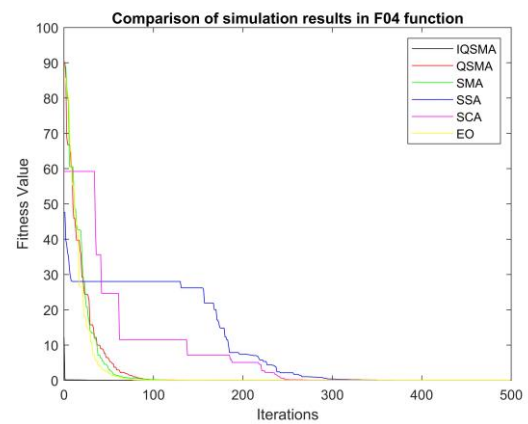
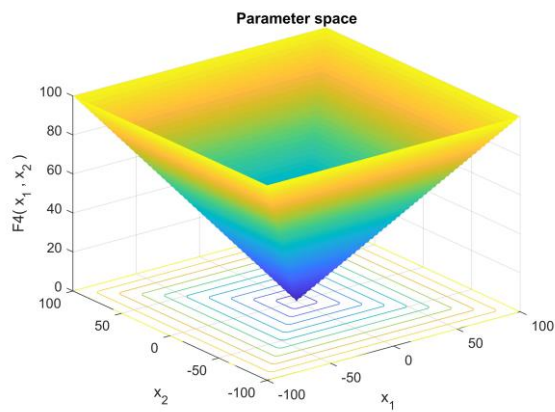
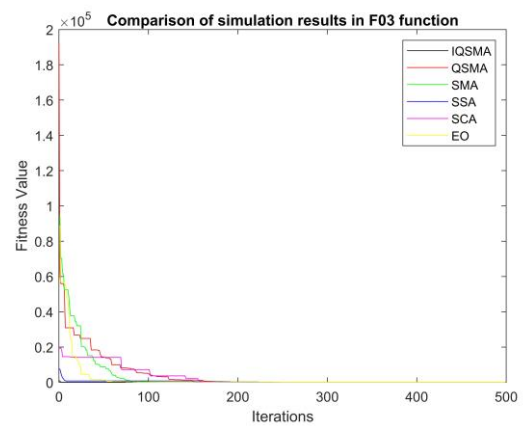
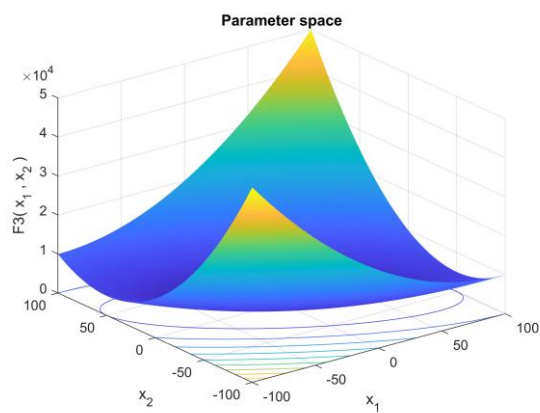
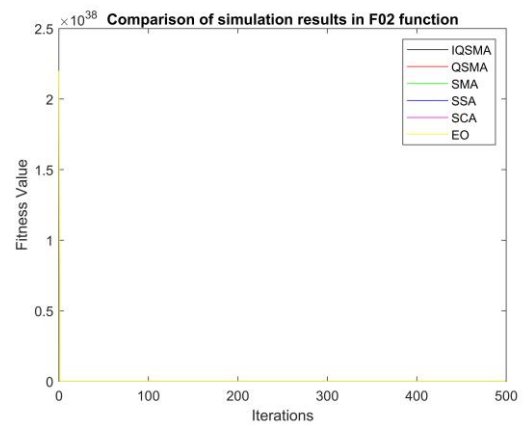
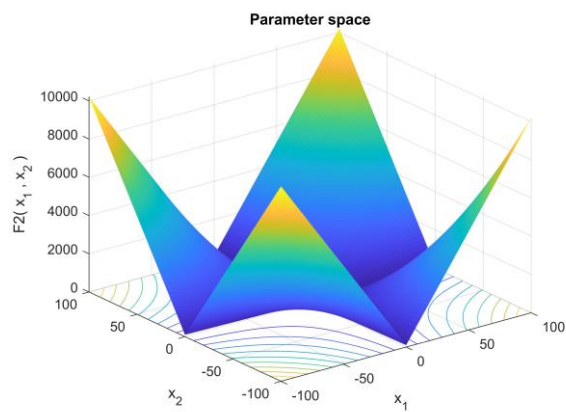
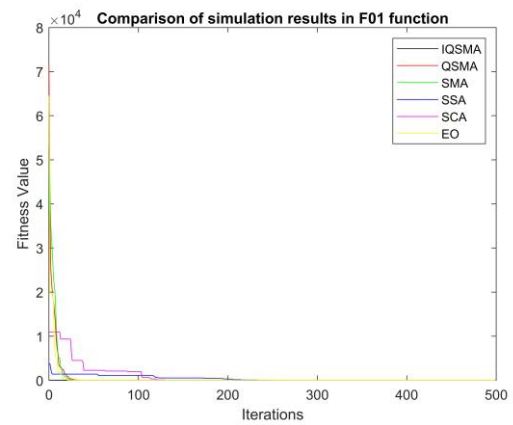
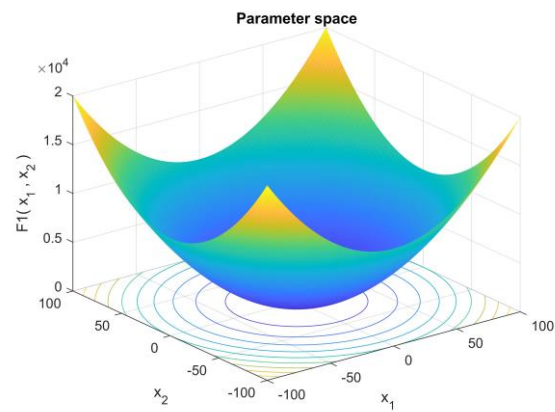
Figure 2 presents two panels for all functions (F01–F23), where the left panel displays the function in 2-D versions while the right panel represents a comparison of convergence curves for all present optimization algorithms. We can see from Figure 2 that both proposed algorithms, QSMA, IQSMA have a steady convergence and a slow convergence acceleration on these benchmark functions compared with (SMA, EO, SCA, and SSA). Furthermore, the IQSMA got better solutions and faster convergence compared to other algorithms.

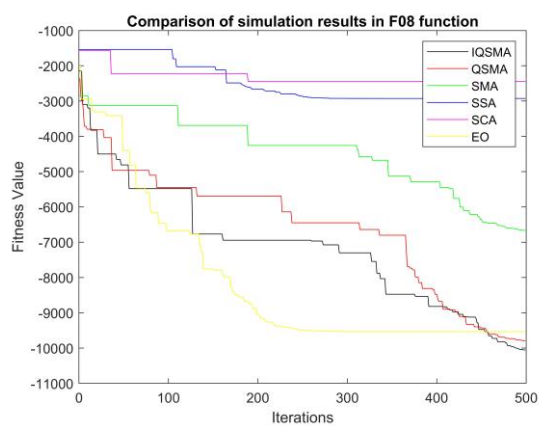
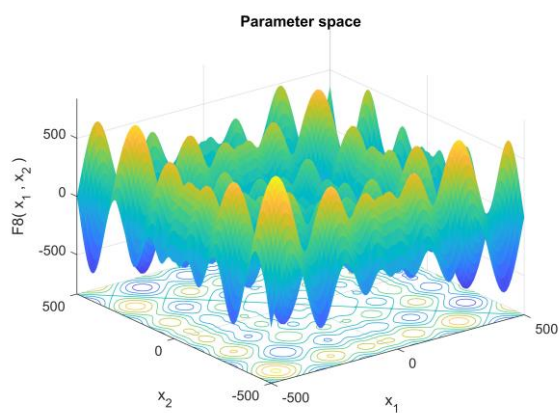
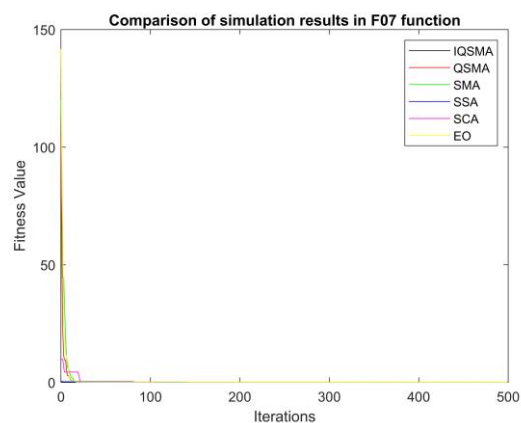
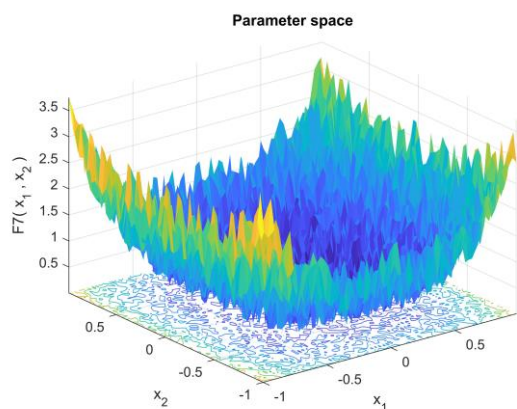
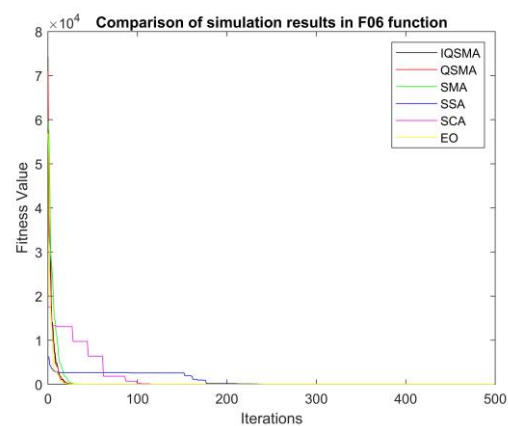
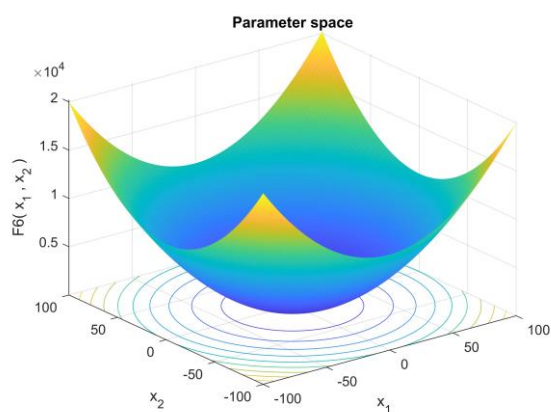
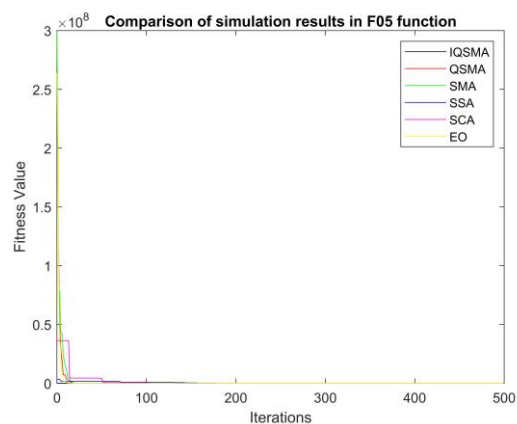
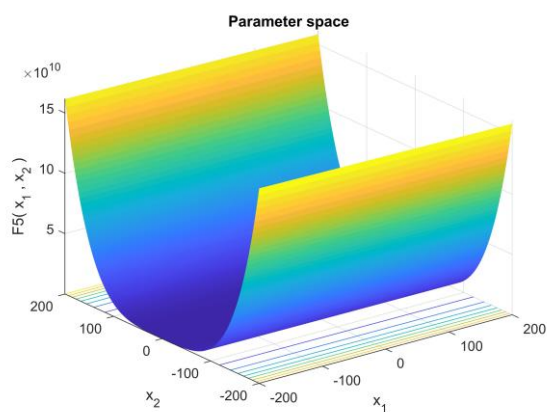
Table 6. Comparison with EO, SCA, SSA, SMA, QSMA, and IQSMA algorithms on test functions. The best results obtained for a function are bold.

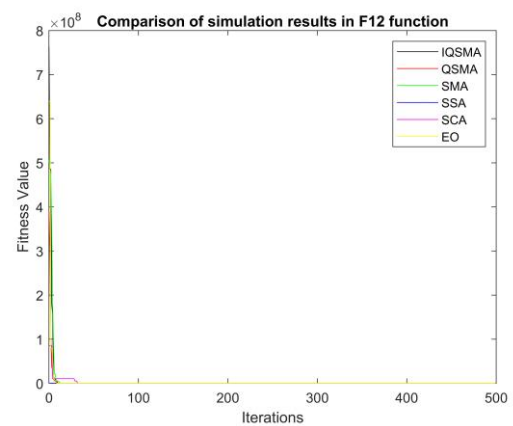
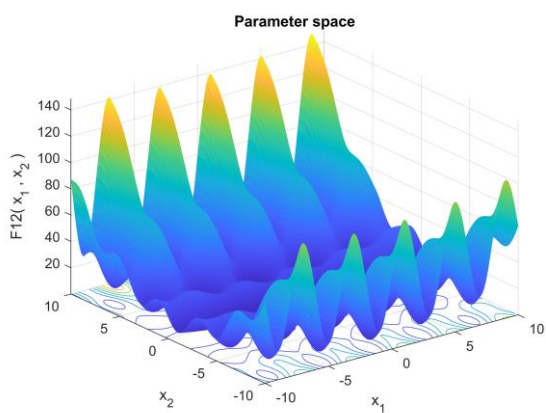
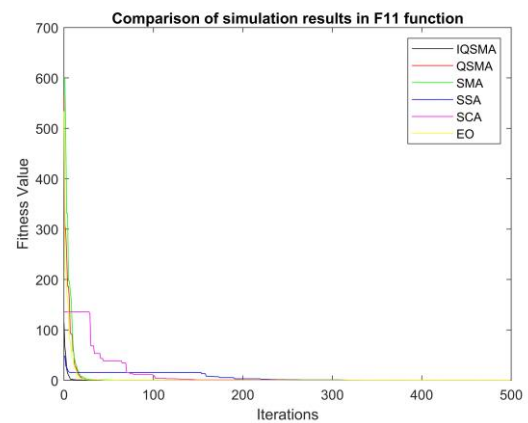
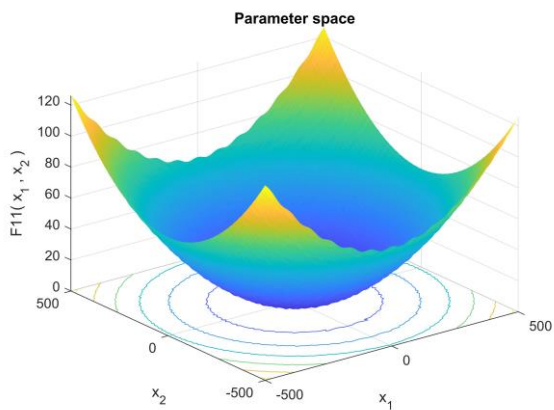
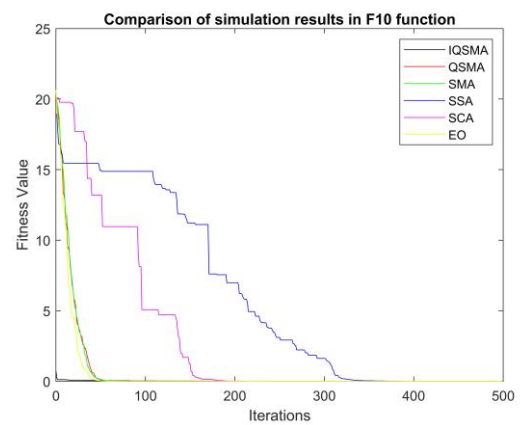
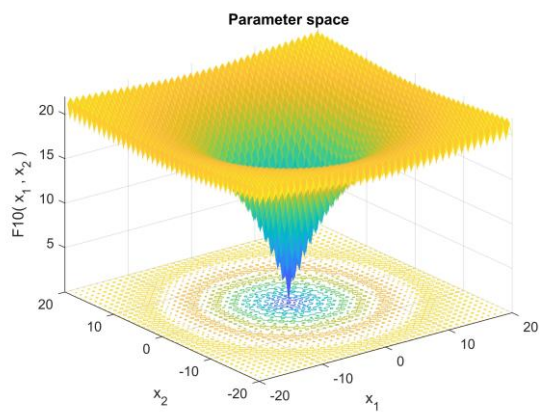
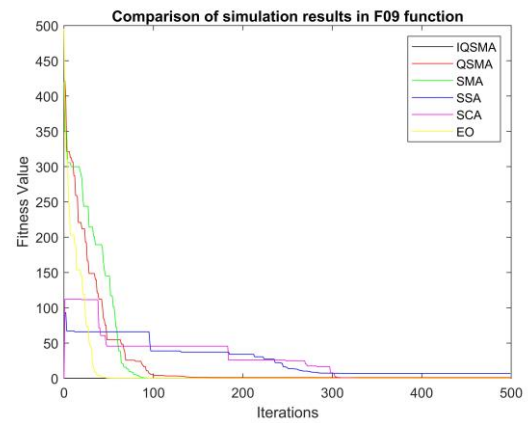
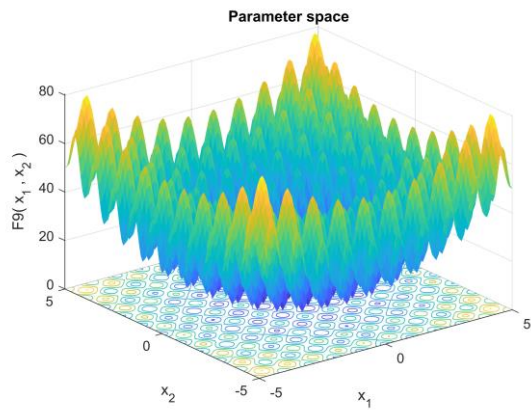
	Function	Stats	EO	SCA	SSA	SMA	QSMA	IQSMA
The unimodal Benchmark functions	F01	Ave.	5.270044e-41	3.815644e-12	1.15892e-09	1.96083e-58	3.44175e-29	0
		Std.	1.1478E-29	3.19286E-08	1.754458502	5.95362E-23	1.27644E-22	4.59816E-12
		E. Time	1.763346	0.099508	0.119495	0.138723	0.432752	0.086588
	F02	Ave.	7.433996e-23	9.295555e-11	6.70439e-06	2.0412e-16	3.97089e-18	0
		Std.	6.7792E-16	1.23476E-08	0.2443115	1.69035E-13	6.73771E-15	1.44342E-08
		E. Time	1.873320	0.105210	0.113219	0.138737	0.448891	0.086043
	F03	Ave.	9.55667e-10	1.959382e-06	0.000127312	3.41714e-06	0.000711066	0
		Std.	9.33802E-07	0.003877813	9.678659161	0.003450459	1.47699419	7.78582E-07
		E. Time	4.732245	0.135475	0.147158	0.230611	0.656757	0.110012
	F04	Ave.	3.955665e-10	0.01256998	5.35723e-05	1.15027e-06	7.5942e-06	0
		Std.	4.90605E-09	0.038970216	0.502296605	9.21444E-06	5.46082E-05	4.78796E-06
		E. Time	1.797801	0.104312	0.112364	0.135770	0.435996	0.084655
	F05	Ave.	25.33655	7.13525	9.44728	28.7173	24.146	7.23231
		Std.	0.153621184	0.305271292	26.43741398	0.002445074	0.517780781	0.004513947
		E. Time	2.156487	0.118066	0.126814	0.149490	0.468276	0.094461
	F06	Ave.	8.503934e-06	0.5243447	6.27924e-10	1.24478	7.56279e-05	8.542847e-06
		Std.	3.90715E-05	0.009659895	2.118305208	0.104756839	0.011343472	4.47065E-05
		E. Time	1.808397	0.102875	0.112237	0.135519	0.437551	0.085192
	F07	Ave.	0.001447345	0.003814452	0.0111709	0.00112319	0.00363481	2.31077e-05
		Std.	0.000325365	1.04083E-17	3.50012E-05	0.000511906	0.000179631	0.000236256
		E. Time	3.447838	0.130378	0.141389	0.187837	0.555963	0.114156
The multimodal Benchmark functions	F08	Ave.	-9192.053	-2443.102	-2926.33	-6665.2	-9798.05	-10058.7
		Std.	9.056721377	1.81899E-12	9.093386519	851.6044849	1322.043877	1010.058747
		E. Time	2.220483	0.125450	0.123094	0.152042	0.478125	0.095939
	F09	Ave.	0	0.004608816	6.96471	1.13687e-13	0.999771	0
		Std.	0	8.164094498	1.593208694	2.61272E-14	0.008047992	0
		E. Time	1.926799	0.108756	0.116575	0.139566	0.140789	0.086963
	F10	Ave.	8.940996e-15	1.402814e-07	1.17605e-05	1.03917e-13	5.77316e-14	8.88178e-16
		Std.	1.64558E-15	2.12116E-05	0.940165488	1.05747E-12	5.30501E-13	3.26728E-11
		E. Time	1.994378	0.111272	0.117104	0.143151	0.448554	0.087436
	F11	Ave.	0	1.084875e-07	0.0885782	0	0	5.38844e-10
		Std.	0	0.015329891	0.329520851	0	0	0.071495982
		E. Time	2.281016	0.120962	0.134175	0.151796	0.482352	0.098219
	F12	Ave.	5.214672e-07	0.1416303	0.31286	0.0248439	1.4269e-05	5.601069e-07
		Std.	4.15054E-06	0.024669585	0.225503305	0.008776058	0.001123422	2.75205E-06
		E. Time	6.035551	0.187610	0.197894	0.287159	0.773408	0.269709

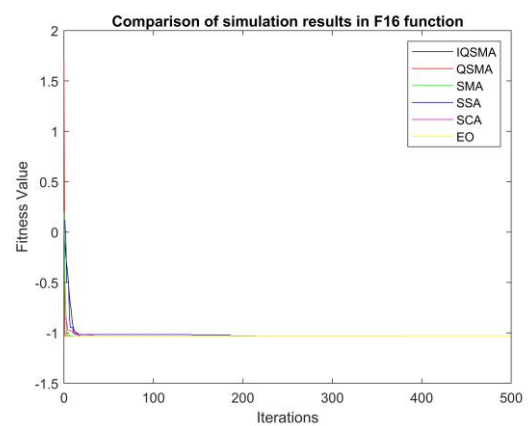
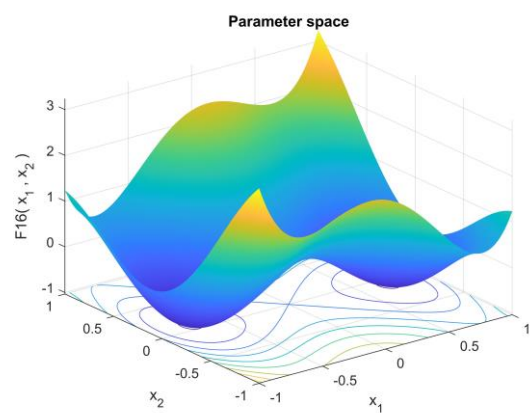
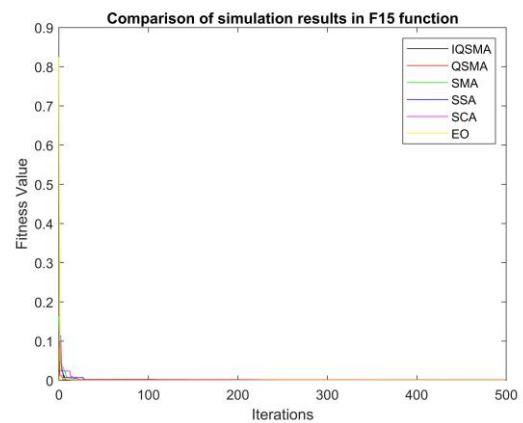
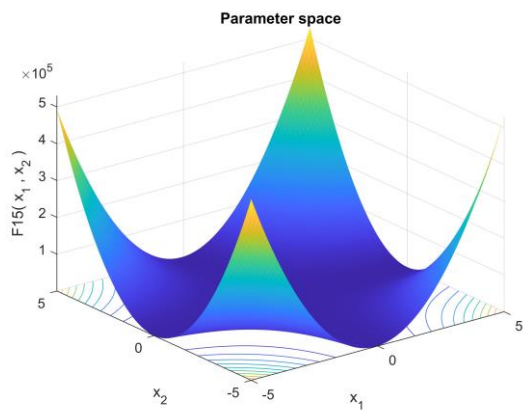
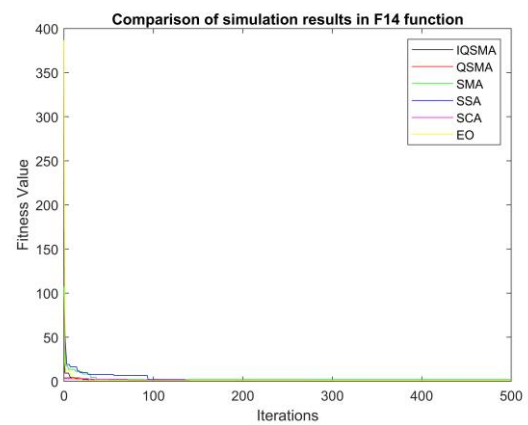
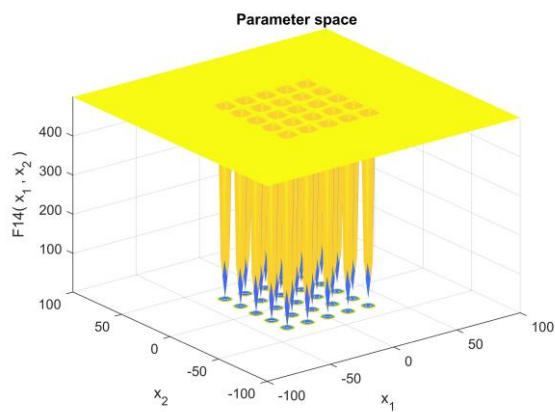
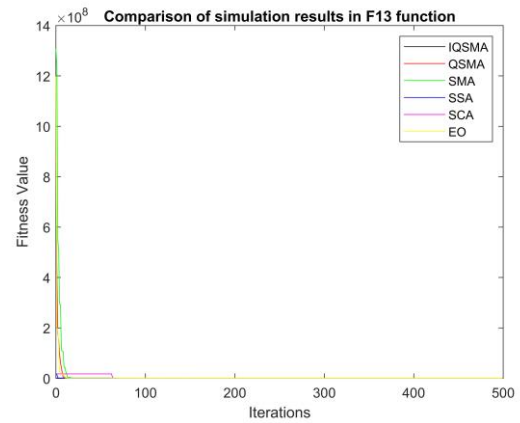
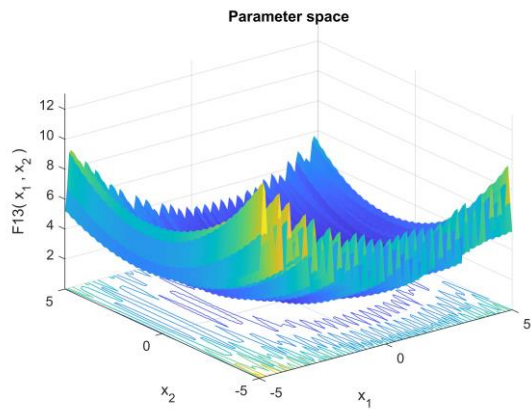
Continued on next page

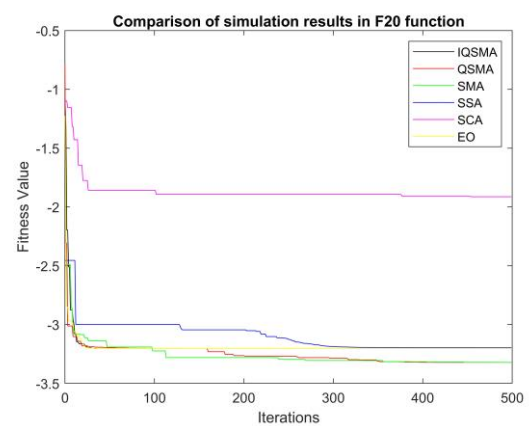
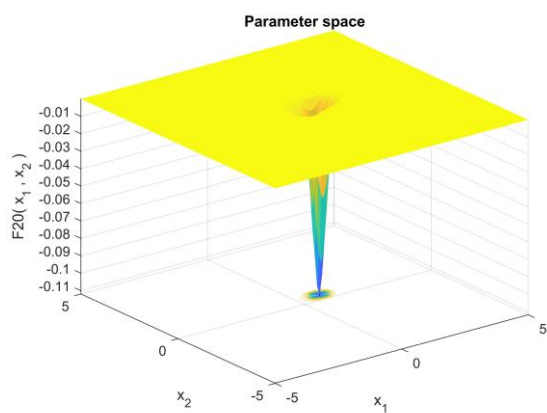
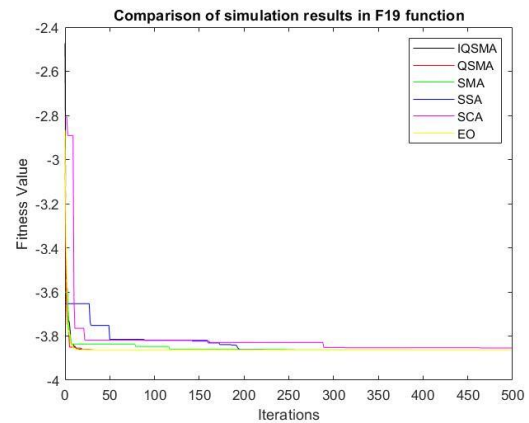
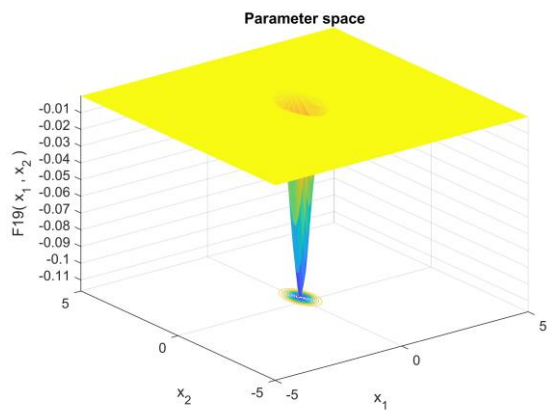
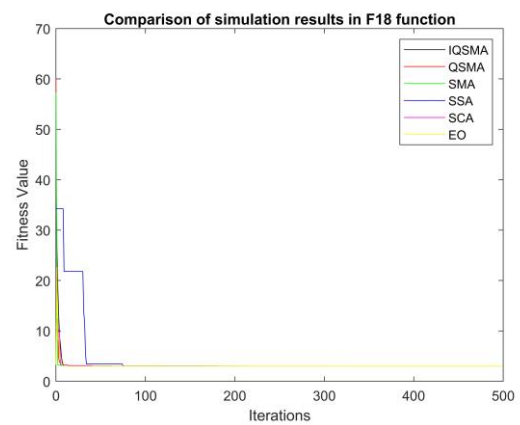
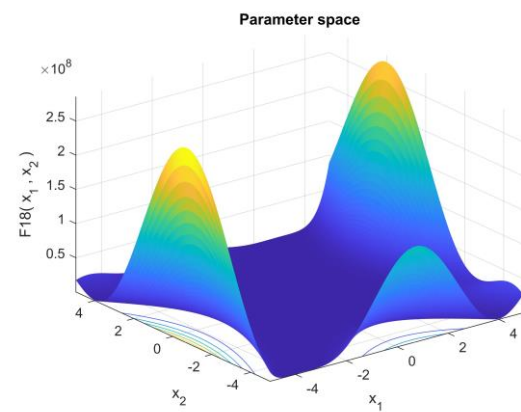
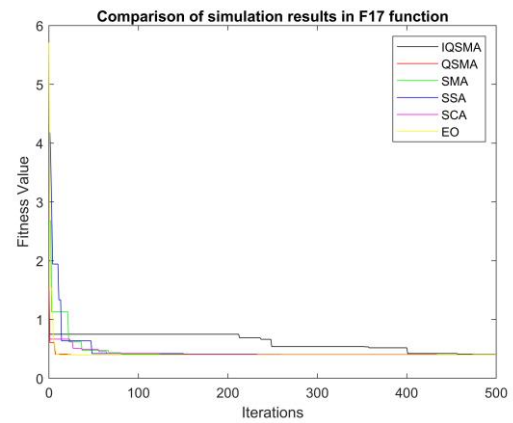
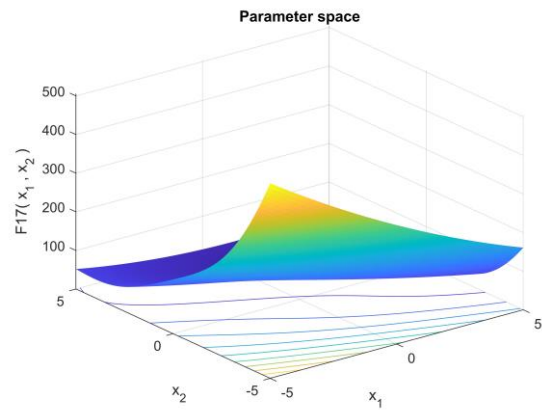
	Function	Stats	EO	SCA	SSA	SMA	Q SMA	IQ SMA
The multimodal Benchmark functions	F13	Ave.	0.03403333	0.3299261	2.90825e-10	0.784618	0.000110568	0.883174
		Std.	7.41539E-05	0.033597886	0.131970448	0.073548622	0.035259838	8.28361E-05
		E. Time	6.072919	0.188790	0.198885	0.286783	0.763508	0.140492
The fixed-dimension multimodal benchmarks functions	F14	Ave.	0.9980038	0.9985377	0.998004	1.99203	0.998004	0.9980038
		Std.	2.10942E-15	0.015017865	2.35563E-07	3.27014E-07	2.30205E-15	2.10942E-15
		E. Time	9.734882	0.374562	0.403516	0.362329	0.972085	0.279195
	F15	Ave.	0.003780983	0.001323074	0.00130454	0.000556103	0.000307492	0.000404767
		Std.	5.3977E-06	7.10932E-05	5.58188E-05	6.21979E-05	3.32098E-07	3.33868E-06
		E. Time	1.503389	0.101859	0.111344	0.089382	0.358223	0.082405
	F16	Ave.	-1.031628	-1.031576	-1.03163	-1.03163	-1.03163	-1.03163
		Std.	1.77636E-15	0.000107559	2.12702E-05	3.3154E-06	1.77636E-15	2.39832E-06
		E. Time	1.465349	0.098138	0.109979	0.084653	0.338040	0.084003
	F17	Ave.	0.3978874	0.3988672	0.3985107	0.397897	0.397887	0.405033
		Std.	1.22125E-15	0.002579281	0.001485582	0.000139089	2.1642E-12	0.058220749
		E. Time	1.320661	0.091624	0.090364	0.079659	0.349219	0.079330
	F18	Ave.	3	3.000004	3.0002	3	3	3
		Std.	3.9968E-15	0.002357281	0.000233576	3.5675E-07	6.66104E-15	2.49168E-06
		E. Time	1.251411	0.090643	0.103164	0.078686	0.326298	0.077892
	F19	Ave.	-3.862519	-3.854372	-3.86278	-3.86272	-3.86278	-3.86278
		Std.	1.46549E-14	0.008607892	7.58897E-05	0.000377588	1.72207E-14	1.86064E-14
		E. Time	1.612180	0.103914	0.114928	0.101968	0.368539	0.085507
	F20	Ave.	-3.270284	-1.913355	-3.19744	-3.32197	-3.32199	-3.248076
		Std.	1.55431E-14	0.009277187	0.015275908	0.007899945	0.017418897	1.55431E-14
		E. Time	1.656609	0.109952	0.118752	0.098775	0.371993	0.089935
	F21	Ave.	-8.963648	-0.4972941	-10.1532	-10.1507	-10.1532	-10.1532
		Std.	3.01981E-14	6.15291E-05	0.077741677	0.581263953	0.394616755	0.168155262
		E. Time	1.839444	0.113855	0.125382	0.100598	0.393808	0.095923
	F22	Ave.	-9.394223	-4.453645	-10.4029	-10.3994	-10.4029	-10.4029
		Std.	2.708409347	0.274960156	0.099958884	1.004334384	7.99579E-06	0.098183121
		E. Time	1.983121	0.126290	0.131289	0.106787	0.399857	0.097894
F23	Ave.	-9.47119	-0.9457124	-2.42173	-10.5353	-10.5364	-10.5364	
	Std.	3.19744E-14	0.000534606	0.001764989	0.956613229	1.56582E-05	1.16999E-05	
	E. Time	2.435197	0.131307	0.140816	0.116054	0.415984	0.106947	
		Rank	3	6	5	4	2	1
		Fast	6	2	3	4	5	1











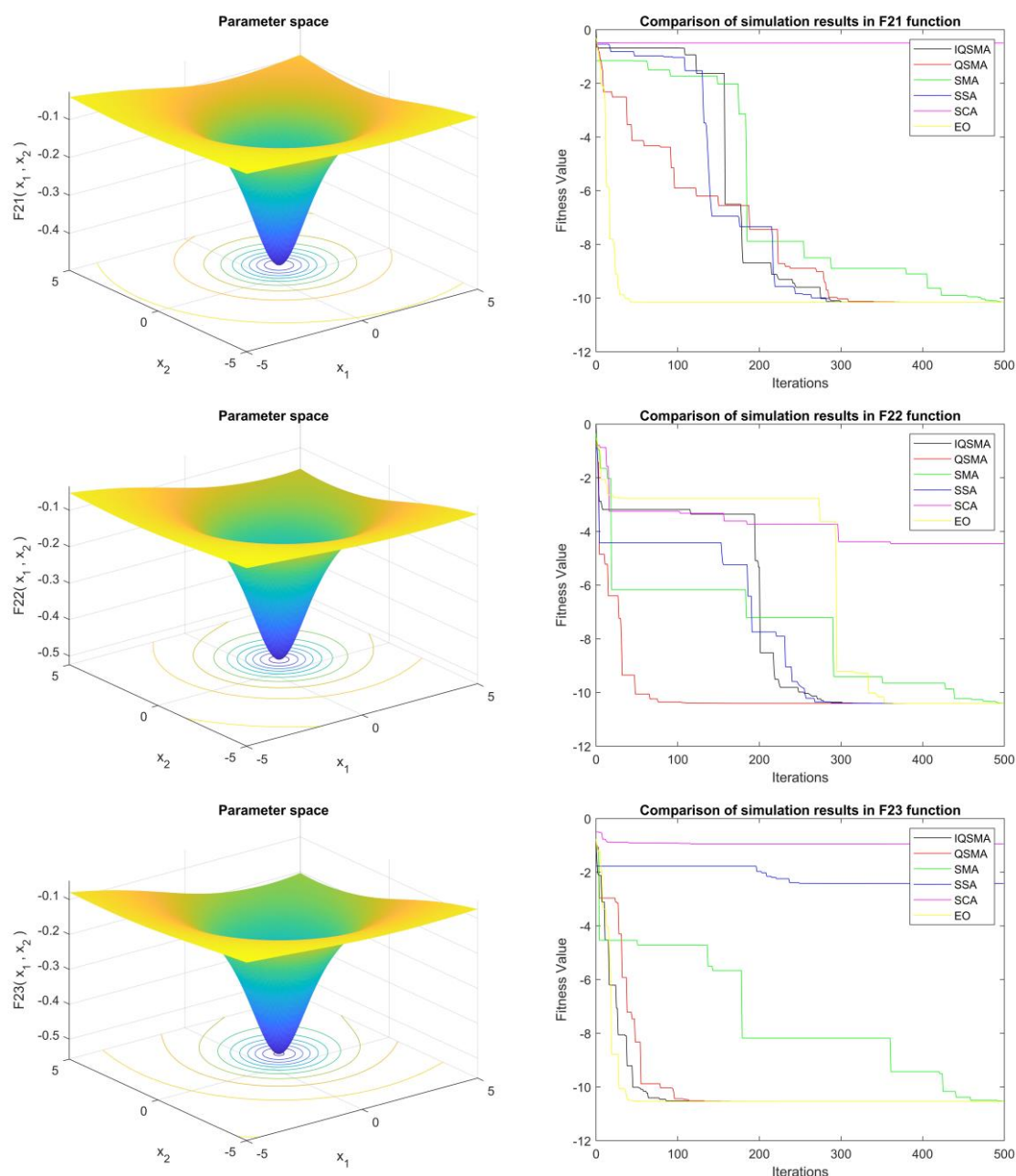


Figure 2. *Left panels:* 2-D perspective view of (F01-F23) functions, respectively. *Right panels:* comparison of convergence curves of EO, SCA, SSA, SMA, QSMA, and IQSMA algorithms for (F01-F23) functions, respectively.

Moreover, Table 7 presents a comparison of the numerical results of the CEC-2019 benchmarks functions mentioned in Table 5. Again, we find that the proposed algorithms gave the best solutions compared to the other algorithms presented in this work. It is worth noting here that the EO algorithm, although it takes a long time, performed better than the rest of the SCA, SSA, and SMA algorithms, as it was competitive with the QSMA and IQSMA algorithms. Also, the IQSMA algorithm got on the minimum root of the sum of deviations from the optimal solution, followed by the QSMA algorithm,

then the SMA algorithm, and then the EO algorithm, while it comes in the last ranks with the SCA and SSA algorithms respectively.

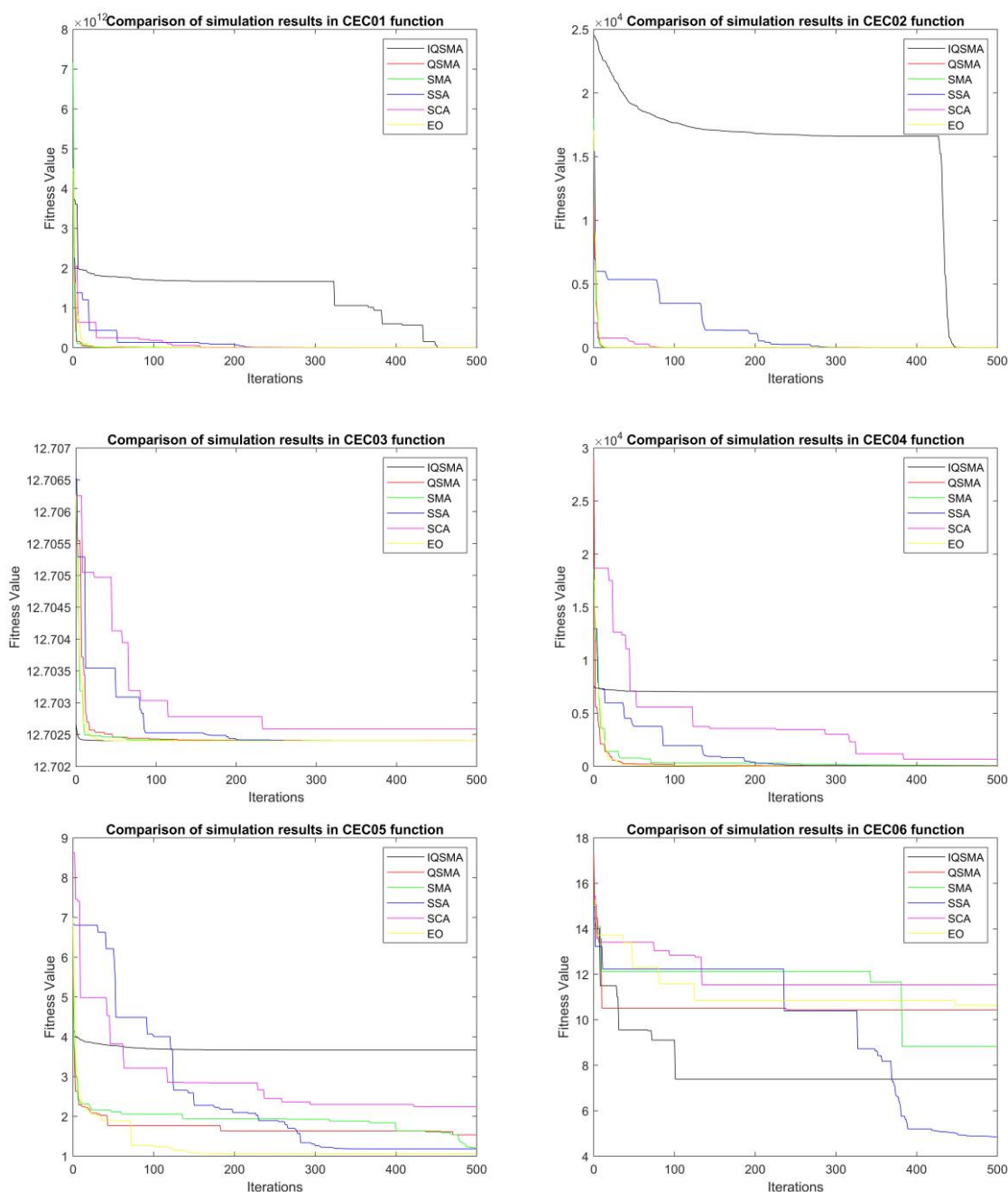
Table 7. Comparison with EO, SCA, SSA, SMA, QSMA, and IQSMA algorithms on CEC-2019 functions. The best results obtained for a function are bold.

Test Problem	Algorithm	EO	SCA	SSA	SMA	QSMA	IQSMA
CEC01	Ave.	3.145296e+07	2.812519e+09	3.87698e+09	8.81203e+06	3.33746e+06	1.20546e+06
	Std.	2.57E+11	2.46E+11	2.83E+11	3.63E+11	2.21E+11	6.72E+11
	E. Time	102.080347	3.031524	3.016315	2.990073	6.479033	1.993069
CEC02	Ave.	17.34286	17.70701	17.4466	17.3436	17.3429	18.9017
	Std.	9.94E+02	2.63E+02	2.12E+03	9.99E+02	6.95E+02	5.97E+03
	E. Time	2.011346	0.117808	0.126572	0.113322	0.400026	0.094950
CEC03	Ave.	12.7024	12.70259	12.7024	12.7024	12.7024	12.7024
	Std.	2.96E-04	5.68E-01	5.68E-01	2.97E-04	4.03E-04	1.47E-05
	E. Time	3.029242	0.148513	0.154776	0.145043	0.453654	0.115704
CEC04	Ave.	15.06255	681.1523	67.6567	71.7866	47.3253	7028
	Std.	1.52E+03	4.08E+03	2.10E+03	1.58E+03	1.60E+03	6.63E+01
	E. Time	2.093936	0.117039	0.127967	0.106275	0.400848	0.095648
CEC05	Ave.	1.041017	2.248647	1.18451	1.16497	1.53659	3.66541
	Std.	5.06E-01	9.64E-01	1.74E+00	3.36E-01	2.70E-01	1.03E-01
	E. Time	2.281686	0.118503	0.126857	0.108568	0.394070	0.094813
CEC06	Ave.	10.07397	11.53995	4.84922	8.81743	10.4301	7.3883
	Std.	9.13E-01	9.44E-01	2.98E+00	1.43E+00	5.75E-01	1.31E+00
	E. Time	34.076261	1.088700	1.073496	1.044657	2.321725	0.735507
CEC07	Ave.	237.8726	857.6024	345.415	844.943	601.238	443.888
	Std.	2.72E+02	2.20E+02	4.20E+02	2.34E+02	2.52E+02	6.72E+01
	E. Time	2.315113	0.117647	0.128836	0.109728	0.396605	0.101581
CEC08	Ave.	3.633052	5.765244	4.50117	4.74241	2.41176	5.75221
	Std.	9.77E-01	6.69E-01	8.76E-01	5.94E-01	1.13E+00	3.63E-01
	E. Time	2.162633	0.116816	0.128245	0.109465	0.392360	0.105646
CEC09	Ave.	2.392111	116.8006	2.76518	2.83258	2.42662	920.457
	Std.	4.57E+02	1.25E+03	1.07E+03	4.32E+02	1.79E+02	2.23E+02
	E. Time	1.930431	0.110602	0.121756	0.101801	0.381826	0.094407
CEC010	Ave.	19.12468	20.57925	20	20.4404	20.4411	20.156
	Std.	1.03E-01	9.28E-01	9.40E-01	1.06E-01	1.41E-01	1.19E-01
	E. Time	2.335176	0.116361	0.128556	0.110719	0.401695	0.100073
Rank		4	5	6	3	2	1
Fast		6	3	4	2	5	1

Regarding the amount of computational time, the IQSMA algorithm had the lowest value in this respect, followed by the SMA algorithm, then the SCA and SSA algorithms, then the QSMA, whereas EO had the highest value. This shows faster convergence and higher efficiency of the proposed

algorithms in solving optimization problems. Figure 3 illustrates the convergence curves of all algorithms presented in this paper.

We can conclude from the above comparisons and the behavior of convergence curves that the proposed algorithms are a competitor candidate for the well-known metaheuristics algorithms in solving mathematical optimization problems. They have proven their ability to reach the promised optimum solution at a high convergence speed in various unimodal, multimodal with high or fixed dimensional and composition optimization problems.



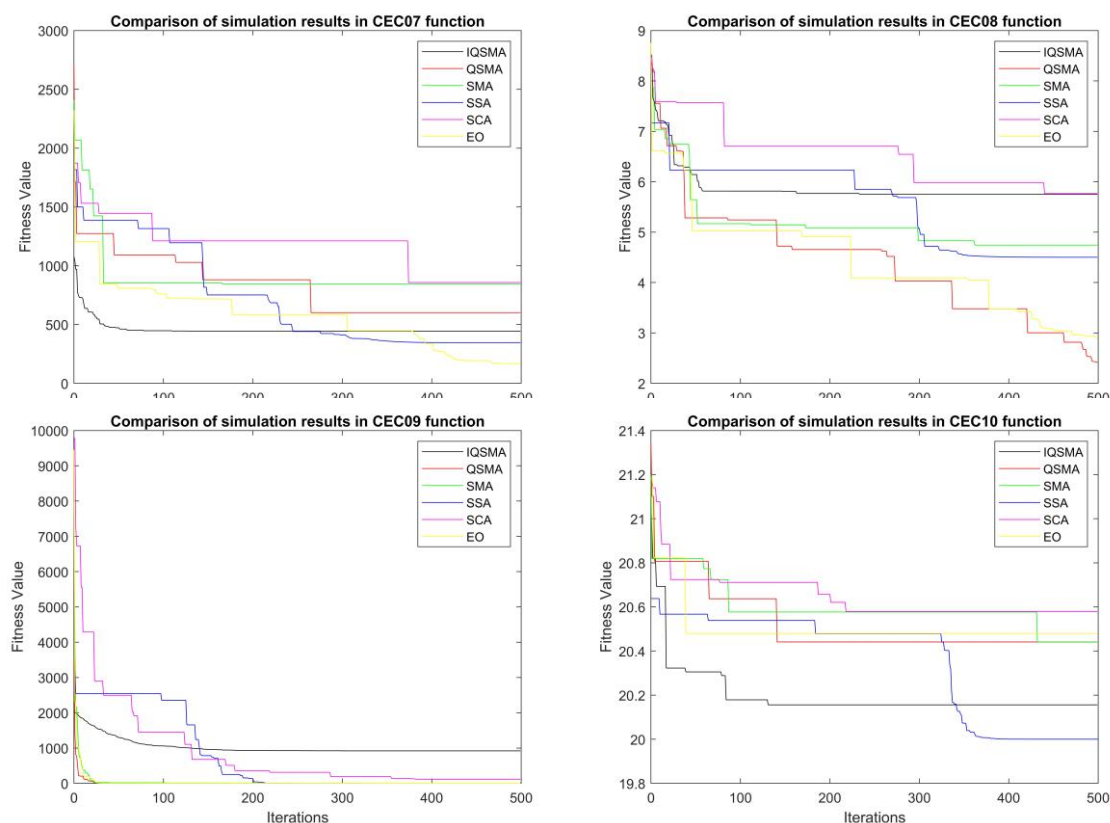


Figure 3. Comparison of convergence curves of EO, SCA, SSA, SMA, QSMA, and IQSMA algorithms for (CEC01-CEC10) respectively.

6. Conclusions

In this paper, we have formulated a quantum variant of the SMA algorithm named QSMA, and We proposed an improved version of the QSMA, namely the IQSMA algorithm based on an interpolation operator. Thirty-three various test benchmark functions were used to validate the performance of both proposed algorithms so that the convergence behavior and the ability to reach the optimal solution were analyzed. The results indicated the superiority of both the proposed algorithms and their competitors to the well-known metaheuristics algorithms; also, the results showed the superiority of the IQSMA, considering the convergence time due to the effectiveness of the interpolation operator. Future work could involve improving these algorithms to handle multi-objective optimization problems. Moreover, the performance of these algorithms could also be tested on combinatorial optimization problems and real-life optimization applications.

Acknowledgments

This work was supported by the Research Supporting Project Number (RSP-2021/389), King Saud University, Riyadh, Saudi Arabia.

Conflict of interest

We have no conflicts with competing interests to disclose.

References

1. O. A. Raouf, I. M. Hezam, Sperm Motility Algorithm: A novel metaheuristic approach for global optimisation, *Int. J. Oper. Res.*, **28** (2017), 143. <https://doi.org/10.1504/IJOR.2017.081473>
2. A. K. Mandal, R. Sen, S. Goswami, A. Chakrabarti, B. Chakraborty, A new approach for feature subset selection using quantum inspired owl search algorithm, In: *2020 10th International Conference on Information Science and Technology (ICIST)*, 2020, 266–273. <https://doi.org/10.1109/ICIST49303.2020.9202140>
3. M. Mirhosseini, M. Fazlali, H. T. Malazi, S. K. Izadi, H. Nezamabadi-pour, Parallel Quadri-valent Quantum-Inspired Gravitational Search Algorithm on a heterogeneous platform for wireless sensor networks, *Comput. Electr. Eng.*, **92** (2021), 107085. <https://doi.org/10.1016/j.compeleceng.2021.107085>
4. L. Gyongyosi, S. Imre, A survey on quantum computing technology, *Comput. Sci. Rev.*, **31** (2019), 51–71. <https://doi.org/10.1016/j.cosrev.2018.11.002>
5. A. W. Harrow, A. Montanaro, Quantum computational supremacy, *Nature*, **549** (2017), 203–209. <https://doi.org/10.1038/nature23458>
6. L. Gyongyosi, S. Imre, Scalable distributed gate-model quantum computers, *Sci. Rep.*, **11** (2021), 5172. <https://doi.org/10.1038/s41598-020-76728-5>
7. F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, et al., Quantum supremacy using a programmable superconducting processor, *Nature*, **574** (2019), 505–510. <https://doi.org/10.1038/s41586-019-1666-5>
8. J. Preskill, Quantum computing in the NISQ era and beyond, *Quantum*, **2** (2018), 79. <https://doi.org/10.22331/q-2018-08-06-79>
9. S. Aaronson, L. Chen, Complexity-theoretic foundations of quantum supremacy experiments, In: *32nd Computational Complexity Conference (CCC 2017)*, 2017. <https://doi.org/10.4230/LIPIcs.CCC.2017.22>
10. Y. Alexeev, D. Bacon, K. R. Brown, R. Calderbank, L. D. Carr, F. T. Chong, et al., Quantum computer systems for scientific discovery, *PRX Quantum*, **2** (2021), 017001. <https://doi.org/10.1103/PRXQuantum.2.017001>
11. D. Awschalom, K. K. Berggren, H. Bernien, S. Bhave, L. D. Carr, P. Davids, et al., Development of quantum interconnects (QuICs) for next-generation information technologies, *PRX Quantum*, **2** (2021), 017002. <https://doi.org/10.1103/PRXQuantum.2.017002>
12. E. Farhi, H. Neven, Classification with quantum neural networks on near termprocessors, *arXiv*, 2018. Available from: <http://arxiv.org/abs/1802.06002>.
13. S. Lloyd, Quantum approximate optimization is computationally universal, 2018. Available from: <http://arxiv.org/abs/1812.11075>.
14. A. Manju, M. J. Nigam, Applications of quantum inspired computational intelligence: A survey, *Artif. Intell. Rev.*, **42** (2014), 79–156. <https://doi.org/10.1007/s10462-012-9330-6>
15. G. Zhang, Quantum-Inspired Evolutionary Algorithms: A survey and empirical study, *J. Heuristics*, **17** (2011), 303–351. <https://doi.org/10.1007/s10732-010-9136-0>
16. D. Johannsen, P. P. Kurur, J. Lengler, Evolutionary algorithms for quantum computers, **68** (2014), 152–189. <https://doi.org/10.1007/s00453-013-9784-1>
17. E. R. Johnston, *Programming quantum computers: Essential algorithms and code samples*, O'Reilly Media, 2019.

18. D. Goswami, Quantum distributed computing applied to Grover's search algorithm, In: *Computing with new resources*, Lecture Notes in Computer Science, Springer, 2014. https://doi.org/10.1007/978-3-319-13350-8_14
19. A. B. Finnila, M. A. Gomez, C. Sebenik, C. Stenson, J. D. Doll, Quantum annealing: A new method for minimizing multidimensional functions, *Chem. Phys. Lett.*, **219** (1994), 343–348. [https://doi.org/10.1016/0009-2614\(94\)00117-0](https://doi.org/10.1016/0009-2614(94)00117-0)
20. M. Steffen, W. van Dam, T. Hogg, G. Breyta, I. Chuang, Experimental implementation of an adiabatic quantum optimization algorithm, *Phys. Rev. Lett.*, **90** (2003), 067903. <https://doi.org/10.1103/PhysRevLett.90.067903>
21. A. Narayanan, M. Moore, Quantum-Inspired Genetic Algorithms, In: *Proceedings of the IEEE Conference on Evolutionary Computation*, 1996, 61–66. <https://doi.org/10.1109/icec.1996.542334>
22. H. Kundra, W. Khan, M. Malik, K. P. Rane, R. Neware, V. Jain, Quantum-Inspired Firefly Algorithm integrated with cuckoo search for optimal path planning, *Int. J. Mod. Phys. C*, **33** (2021), 2250018. <https://doi.org/10.1142/S0129183122500188>
23. N. R. Eluri, G. R. Kancharla, S. Dara, V. Dondeti, Cancer data classification by quantum-inspired immune clone optimization-based optimal feature selection using gene expression data: Deep learning approach, *Date Technol. Appl.*, 2021. <https://doi.org/10.1108/DTA-05-2020-0109>
24. B. Arun, Quality materialised view selection using quantum inspired artificial bee colony optimisation, *Int. J. Intell. Inf. Database Syst.*, **13** (2020), 33–60. <https://doi.org/10.1504/IJIDS.2020.108215>
25. Z. Gao, Y. Zhang, S. Zhou, W. Lyu, An enhanced Quantum-Inspired Gravitational Search Algorithm for color prediction based on the absorption spectrum, *Text. Res. J.*, **91** (2021), 1211–1226. <https://doi.org/10.1177/0040517520977007>
26. Y. Meraihi, D. Acheli, A. R. Cherif, M. Mahseur, A Quantum-Inspired Binary Firefly Algorithm for QoS multicast routing, *Int. J. Metaheuristics*, **6** (2017), 309–333. <https://doi.org/10.1504/IJMHEUR.2017.086980>
27. T. C. Lu, J. C. Juang, Quantum-Inspired Space Search Algorithm (QSSA) for global numerical optimization, *Appl. Math. Comput.*, **218** (2011), 2516–2532. <https://doi.org/10.1016/j.amc.2011.07.067>
28. A. Layeb, A hybrid quantum inspired harmony search algorithm for 0–1 optimization problems, *J. Comput. Appl. Math.*, **253** (2013), 14–25. <https://doi.org/10.1016/j.cam.2013.04.004>
29. R. K. Agrawal, B. Kaur, P. Agarwal, Quantum Inspired Particle Swarm Optimization with guided exploration for function optimization, *Appl. Soft Comput.*, **102** (2021), 107122. <https://doi.org/10.1016/j.asoc.2021.107122>
30. A. S. Thakur, T. Biswas, P. Kuila, Binary quantum-inspired gravitational search algorithm-based multi-criteria scheduling for multi-processor computing systems, *J. Supercomput.*, **77** (2021), 796–817. <https://doi.org/10.1007/s11227-020-03292-0>
31. K. Mishra, R. Pradhan, S. K. Majhi, Quantum-Inspired Binary Chaotic Salp Swarm Algorithm (QBCSSA)-based dynamic task scheduling for multiprocessor cloud computing systems, *J. Supercomput.*, **77** (2021), 10377–10423. <https://doi.org/10.1007/s11227-021-03695-7>
32. R. Pradhan, M. R. Khan, P. K. Sethy, S. K. Majhi, QALO-MOR: Improved antlion optimizer based on quantum information theory for model order reduction, *J. Intell. Fuzzy Syst.*, **41** (2021), 5747–5757. <https://doi.org/10.3233/JIFS-189894>

33. S. A. Mohsin, A. Younes, S. M. Darwish, Dynamic cost ant colony algorithm to optimize query for distributed database based on quantum-inspired approach, *Symmetry*, **13** (2021), 1–20. <https://doi.org/10.3390/sym13010070>
34. V. P. Soloviev, C. Bielza, P. Larranaga, Quantum-Inspired Estimation of Distribution Algorithm to solve the travelling salesman problem, In: *2021 IEEE Congress on Evolutionary Computation (CEC)*, 2021, 416–425. <https://doi.org/10.1109/CEC45853.2021.9504821>
35. M. Soleimanpour-Moghadam, H. Nezamabadi-Pour, An improved quantum behaved gravitational search algorithm, In: *ICEE 2012-20th Iranian Conference on Electrical Engineering*, (2012), 711–715. <https://doi.org/10.1109/IranianCEE.2012.6292446>
36. A. S. Hesar, S. R. Kamel, M. Houshmand, A quantum multi-objective optimization algorithm based on harmony search method, *Soft. Comput.*, **25** (2021), 9427–9439. <https://doi.org/10.1007/s00500-021-05799-x>
37. X. Liu, G. G. Wang, L. Wang, LSFQPSO: Quantum particle swarm optimization with optimal guided Lévy flight and straight flight for solving optimization problems, *Eng. Comput.*, 2021. <https://doi.org/10.1007/s00366-021-01497-2>
38. X. Zhang, S. Xia, X. Li, Quantum behavior-based enhanced fruit fly optimization algorithm with application to UAV path planning, *Int. J. Comput. Intell. Syst.*, **13** (2020), 1315. <https://doi.org/10.2991/ijcis.d.200825.001>
39. X. Zhang, S. Xia, Quantum behaved fruit fly optimization algorithm for continuous function optimization problems, In: *Advances in swarm intelligence*, Lecture Notes in Computer Science, Springer, 2019, 331–340. https://doi.org/10.1007/978-3-030-26369-0_31
40. A. Kaveh, M. Kamalinejad, H. Arzani, Quantum evolutionary algorithm hybridized with Enhanced colliding bodies for optimization, *Structures*, **28** (2020), 1479–1501. <https://doi.org/10.1016/j.istruc.2020.09.079>
41. N. R. Zhou, S. H. Xia, Y. Ma, Y. Zhang, Quantum particle swarm optimization algorithm with the truncated mean stabilization strategy, *Quantum Inf. Process.*, **21** (2022), 42. <https://doi.org/10.1007/s11128-021-03380-x>
42. M. S. Alvarez-Alvarado, F. E. Alban-Chacón, E. A. Lamilla-Rubio, C. D. Rodríguez-Gallegos, W. Velásquez, Three novel quantum-inspired swarm optimization algorithms using different bounded potential fields, *Sci. Rep.*, **11** (2021), 11655. <https://doi.org/10.1038/s41598-021-90847-7>
43. A. T. Khan, X. Cao, S. Li, B. Hu, V. N. Katsikis, Quantum beetle antennae search: a novel technique for the constrained portfolio optimization problem, *Sci. China Inf. Sci.*, **64** (2021), 152204. <https://doi.org/10.1007/s11432-020-2894-9>
44. S. Palosaari, S. Parviainen, J. Hironen, J. Reunanen, P. Neittaanmaki, A random search algorithm for constrained global optimization, *Acta Polytech. Scand.-Chem. Technol.*, **172** (1986), 2–45.
45. N. Manzanares-Filho, R. B. F. Albuquerque, B. S. Sousa, L. G. C. Santos, A comparative study of controlled random search algorithms with application to inverse aerofoil design, *Eng. Optim.*, **50** (2018), 996–1015. <https://doi.org/10.1080/0305215X.2017.1359584>
46. Y. Sun, T. Yang, Z. Liu, A whale optimization algorithm based on quadratic interpolation for high-dimensional global optimization problems, *Appl. Soft Comput.*, **85** (2019), 105744. <https://doi.org/10.1016/j.asoc.2019.105744>
47. D. Singh, S. Agrawal, Self organizing migrating algorithm with quadratic interpolation for solving large scale global optimization problems, *Appl. Soft Comput.*, **38** (2016), 1040–1048. <https://doi.org/10.1016/j.asoc.2015.09.033>

48. A. Kaveh, M. I. Ghazaan, F. Saadatmand, Colliding bodies optimization with Morlet wavelet mutation and quadratic interpolation for global optimization problems, *Eng. Optim.*, 2021. <https://doi.org/10.1007/s00366-020-01236-z>
49. Y. Sun, Y. Chen, Multi-population improved whale optimization algorithm for high dimensional optimization, *Appl. Soft Comput.*, **112** (2021), 107854. <https://doi.org/10.1016/j.asoc.2021.107854>
50. H. Nezamabadi-pour, A Quantum-Inspired Gravitational Search Algorithm for binary encoded optimization problems, *Eng. Appl. Artif. Intell.*, **40** (2015), 62–75. <https://doi.org/10.1016/j.engappai.2015.01.002>
51. D. J. Smith, E. A. Gaffney, J. R. Blake, J. C. Kirkman-Brown, Human sperm accumulation near surfaces: A simulation study, *J. Fluid Mech.*, **621** (2009), 289–320. <https://doi.org/10.1017/S0022112008004953>
52. D. J. Smith, A boundary element regularized Stokeslet method applied to cilia-and flagella-driven flow, *Proc. R. Soc. A, Math. Phys. Eng. Sci.*, **465** (2009), 3605–3626. <https://doi.org/10.1098/rspa.2009.0295>
53. V. Christianto, F. Smarandache, An exact mapping from Navier-Stokes equation to Schrödinger equation via Riccati equation, *Prog. Phys.*, **1** (2007), 38–39.
54. K. Dietrich, D. Vautherin, Sur l'équivalence entre des types particuliers des équations de Navier-Stokes et de Schrödinger non linéaire, *J. Phys.*, **46** (1985), 313–316. <https://doi.org/10.1051/jphys:01985004603031300>
55. V. V. Kulish, J. L. Lage, Exact solutions to the Navier-Stokes equation for an incompressible flow from the interpretation of the Schroedinger wave function, *arXiv*, 2013. Available from: <https://arxiv.org/abs/1301.3586>.
56. T. Schürmann, I. Hoffmann, A closer look at the uncertainty relation of position and momentum, *Found. Phys.*, **39** (2009), 958–963. <https://doi.org/10.1007/s10701-009-9310-0>
57. N. Manzanares-Filho, C. A. A. Moino, A. B. Jorge, An Improved Controlled Random Search Algorithm for inverse airfoil cascade design, In: *Proceedings of 6th World Congresses of Structural and Multidisciplinary Optimization*, 2005.
58. M. Pant, R. Thangaraj, A. Abraham, A new quantum behaved particle swarm optimization, In: *GECCO'08: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, 2008, 87–94. <https://doi.org/10.1145/1389095.1389108>
59. A. Manju, M. J. Nigam, An improved quantum inspired firefly algorithm with interpolation operator, In: *Proceedings of the Third International Conference on Soft Computing for Problem Solving*, Advances in Intelligent Systems and Computing, Springer, 2014. https://doi.org/10.1007/978-81-322-1771-8_7
60. N. Manzanares-Filho, R. B. F. Albuquerque, Accelerating controlled random search algorithms using a distribution strategy, *EngOpt 2008-Int. Conf. Eng. Optim.*, 2008.
61. B. S. De Sousa, N. Manzanares-Filho, A. B. Jorge, Multiobjective laminar-flow airfoil shape optimization using a controlled random search algorithm, *EngOpt 2008-Int. Conf. Eng. Optim.*, 2008.
62. A. H. Gandomi, A. H. Alavi, Krill herd: A new bio-inspired optimization algorithm, *Commun. Nonlinear Sci. Numer. Simul.*, **17** (2012), 4831–4845. <https://doi.org/10.1016/j.cnsns.2012.05.010>
63. I. M. Hezam, O. A. Raouf, M. M. Hadhoud, A new compound swarm intelligence algorithms for solving global optimization problems, *Int. J. Comput. Technol.*, **10** (2013), 2010–2020. <https://doi.org/10.24297/ijct.v10i9.1389>

64. M. Jamil, X. S. Yang, A literature survey of benchmark functions for global optimisation problems, *Int. J. Math. Model. Numer. Optim.*, **4** (2013), 150–194. <https://doi.org/10.1504/IJMMNO.2013.055204>
65. K. V. Price, N. H. Awad, M. Z. Ali, P. N. Suganthan, *Problem definitions and the 100-digit challenge: Problem definitions and evaluation criteria for the 100-digit challenge special session and competition on single objective numerical optimization*, Technical Report, Singapore: Nanyang Technological University, 2018.
66. A. Faramarzi, M. Heidarinejad, B. Stephens, S. Mirjalili, Equilibrium optimizer: A novel optimization algorithm, *Knowl.-Based Syst.*, **191** (2020), 105190. <https://doi.org/10.1016/j.knosys.2019.105190>
67. S. Mirjalili, SCA: A Sine Cosine Algorithm for solving optimization problems, *Knowl.-Based Syst.*, **96** (2016), 120–133. <https://doi.org/10.1016/j.knosys.2015.12.022>
68. S. Mirjalilia, A. H. Gandomibf, S. Z. Mirjalilic, S. Saremi, H. Farisd, S. M. Mirjalilie, Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems, *Adv. Eng. Softw.*, **114** (2017), 163–191. <https://doi.org/10.1016/j.advengsoft.2017.07.002>



AIMS Press

© 2022 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)