Chernobyl disaster optimizer (CDO): a novel meta-heuristic method for global optimization

Article *in* Neural Computing and Applications · February 2023

DOI: 10.1007/s00521-023-08261-1

1 author:

Hisham Shehadeh
Amman Arab University

Neural Computing and Applications
https://doi.org/10.1007/s00521-023-08261-1

ORIGINAL ARTICLE

Chernobyl disaster optimizer (CDO): a novel meta-heuristic method for global optimization

Hisham A. Shehadeh[1]

Received: 20 April 2022 / Accepted: 6 January 2023

Abstract
This paper proposes a novel meta-heuristic optimization method, namely "Chernobyl Disaster Optimizer (CDO)". The underlying concepts and principles behind the proposed approach is inspired by the nuclear reactor core explosion of Chernobyl. In CDO, radioactivity happened because of nuclear instability, which different types of radiations are emitted from nuclei. The most common kinds of these radiations are called gamma, beta, and alpha particles. These particles fly away from the explosion point (high pressure point) to the low pressure point (the human standing point), which are harmful to the humans. The CDO mimics the process of nuclear radiation while attaching human after the nuclear explosion. The main steps of nuclear explosion and attaching human are implemented in which gamma, beta, and alpha particles are involved in this process. The CDO is evaluated with optimizing "Congress on Evolutionary Computation (CEC 2017)" test bed suites. In addition, it is compared against well-known optimization methods, such as "Sperm Swarm Optimization" and "Gravitational Search Algorithm". The experimental results prove its efficiency, which can be considered as viable alternative.

Keywords Physical-based algorithms · Swarm-based algorithms · Sperm swarm optimization (SSO) · Gravitational search algorithm (GSA) · (CEC) 2017 benchmarks functions

1 Introduction

In few years ago, meta-heuristic optimization methods have been emerged. Those methods can be divide into four parts include: "evolutionary-based methods, human-based methods, swarm-based methods, and physical-based methods" [1]. First, evolutionary-based methods are inspired by the evaluation theory of Darwin. An example of this method is "Genetics Algorithm (GA)" [2]. Second, human-based methods are inspired by the lifestyle, attitude or perception of human. Examples of these methods are ''Harmony Search Algorithm (HSA)'' and ''Fireworks Algorithm (FA)'' [3, 4]. Third, swarm-based methods are inspired by swarms of creatures, which simulates their ability to reproduction or survive. Example of this algorithm is "Sperm Swarm Optimization (SSO)" [5, 6]. Fourth, physical-based methods are inspired by the physical rules and theories of the universe. Example of these algorithms is "Gravitational Search Algorithm (GSA)" [7].

The basic goal of these approaches is to generate the best solution with the highest rate of convergence. To achieve this, the aforementioned methods should utilize exploration and exploitation principles in an attempt to identify the solution of global optimal. The idea of exploitation refers to a method's ability to converge on the best solution of any problem. On the other hand, the idea of exploration refers to a method's ability to discover every portion of domain of search space. Hence, the primary goal of all meta-heuristic approaches is to balance between the aforementioned principles.

According to the per above, various methods are found to get a solution of different problems in real-life paradigm. Various researches in the literature have been attained to solve various types of problems, which few of them are summarized as follows :

✉ Hisham A. Shehadeh
h.shehadeh@aau.edu.jo

1 Departments of Computer Information System and Computer Science, Faculty of Computer Science and Informatics, Amman Arab University, Amman, Jordan

Shehadeh et al. [3, 4] have proposed new meta-heuristic optimization algorithm, called "Sperm Swarm Optimization (SSO)". This algorithm is inspired by motility of sperm while fertilizing the egg. They have tested the algorithm on different uni-model and multi-model benchmark functions. The experimental results have proved its good efficiency and convergence speed. The limitation of this method is easily tripped into a local minimum. To overcome this problem, Shehadeh et al. [1] have hybridize "Sperm Swarm Optimization (SSO)" with ''Gravitational Search Algorithm (GSA)'' to increase the algorithm convergence and speed. By merging exploration technique of GSA and exploitation technique of SSO, this algorithm has been tested on ''congress on evolutionary computation (CEC)'' 2017 benchmark suite. The results have illustrated that the hybrid version outperformed the classical SSO and GSA.

On the other hand, Shehadeh et al. [8] have hybridized "Sperm Swarm Optimization (SSO)" with "Genetic Algorithm (GA)" to increase the algorithm convergence while solving a multi-model functions. These types of functions mainly have a set of local minims and single global minima, which is a challenging procedure to skip these multi-local minims to reach the global optima. This algorithm is compared against the classical SSO and GA. The results proved its efficiency and convergence speed.

In a different presentation, Shehadeh et al. [9] have hybridized "Sperm Swarm Optimization (SSO)" with "Grey Wolf Optimizer (GWO)" to increase the algorithm convergence. The hybrid algorithm is tested on different benchmark functions, which consist of uni-model, multi-model, and fix dimensions multi-model benchmark functions. These benchmark functions are taken from "Congress on Evolutionary Computation (CEC 2017)". The results proved that the proposed algorithm outperformed the standard ones in term of efficiency and convergence speed.

An optimal design of "Reinforced Concrete Cantilever Retaining Walls (RCCRW)" is proposed in Khajehzadeh et al. [10] by using an enhancement version of "Sperm Swarm Optimization (SSO)", namely "Adaptive Sperm Swarm Optimization Algorithm (ASSO)". The design of the aforementioned walls are taken in consideration with some aspects, such as economic design, seismic and static loading. The results showed that the ASSO outperformed other algorithms in the term of accuracy.

In different view, Sundararaju et al. [11] have proposed an enhancement version of standard SSO, namely "Chaotic Search-Based Hybrid Sperm Swarm Optimized-Gravitational Search Algorithm (CSSO-GSA)". They use this method in optimizing "Automatic Load Frequency Control (ALFC)" of "Hybrid Power System (HPS)". The objective functions of the HPS are related to different resources, such as Renewable Energy (RE), bio-fuel, and thermal, which are used to generate power to balance the system's demand. The results showed that the CSSO-GSA outperformed other algorithms in optimizing these models.

On the other hand, Khajehzadeh et al. [12] have proposed an enhancement version of standard SSO, namely "Chaotic Sperm Swarm Optimization (CSSO)". They used this method to analyse the stability of seismic slope of earth. The objective function of this problem can be considered as a complex problem, which is a multi-model objective function. The outcome of this study showed that the new method could generate better optimal solution than the other methods in the literature.

The "Potassium Chloride (KCl)" is a very important chemical, which plays a significant role in induce stress. Based on that Concepcion et al. [13], studied the effects of KCl dynamics on crops especially tomato, which is the most consumed vegetables in the world. Concepcion et al. [13] have used two algorithms along SSO, which are jellyfish, and moth-flame optimization algorithms to optimize fitness models of five phytomorphological phenotypes related to determine the most suitable KCl concentration for growth promotion. The results showed that SSO algorithm outperformed the other algorithms in optimizing the aforementioned models with an accurate consistency.

In different discussion, Cvetkovski et al. [14] have applied GSA in optimizing the efficiency of permanent magnet synchronous motor. Where the objective model of this problem is the investigated motor efficiency. The outcomes proved that GSA outperformed the other methods in optimizing the aforementioned models with an accurate consistency.

On the other hand, Li et al. [15] have proposed a hybrid algorithm, namely "Chaotic Search and Gravitational Search Algorithm (GSA)-SVM (CGSA-SVM)", which marge "Support Vector Machine (SVM)" with Chaotic Search of GSA to increase the classification accuracy. Fourteen UCI datasets are utilized to test the accuracy of the proposed algorithm. The results of this algorithm are compared against PSO-SVM, GA-SVM, and GSA-SVM. The outcomes of this study showed that the new method could classify the proposed datasets with better accuracy than the other methods in the literature.

Later on, Kumar et al. [16] have proposed a new method that select a set of clusters along with relevant data features simultaneously and automatically. This method is created based on GSA algorithm, which can find the optimal number of clusters and their corresponding features through the run time. This approach is compared against recently developed well-known clustering techniques. The results prove the accuracy and effectiveness of the proposed method.

There are different proses of SSO and GSA, which can be stated in the following dots [1, 5–8]:

- GSA is very easy to implement and understand.

- The exploration ability in GSA is very good.
- GSA can be easily implemented to solve various types of real-life tasks, such as feature subset selection [17], optimal power flow problem [18], classifications problems [19], flow shop scheduling problem [20], timetable scheduling problems [21], and forecasting of water level [22].
- GSA can be utilized to overcome non-continuous, parametrical problems nonparametric, non-differential, continuous, even multidimensional.
- SSO has merit exploitation capability.
- The SSO capability of discovering global optimum, simplicity, and even speed of convergence is proved in different studies in the literature.
- The concept of SSO is inferred from the merit of intelligence, which can be used in various disciplines, such as engineering and scientific studies.
- In SSO, the overlapped measurements are not available.
- In SSO, the computation and theory are very tiny and simple, which can be utilized and understand easily.

On the other hand, there are various odds of SSO and GSA, which can be listed as in the following dots [6, 23].

- The rate of convergence in GSA is very slow.
- The search procedure in GSA is very.
- The ability of search is very slow in GSA, which affect its capability of exploitation.
- For tasks with a wide dimensional search space, SSO may face problem in escaping the local minima.

The aforementioned drawbacks of previous mentioned algorithms are opened the door for us to propose a new method of optimization. This algorithm called "Chernobyl Disaster Optimizer (CDO)", which is inspired by the disaster that was happened in Chernobyl due to the explosion of unclear rector. Generally, the unclear explosion triggers three types of radiation (particles), which are beta, gamma, and alpha. These particles transferred in the medium in different speeds. The proposed algorithm will be capable to discover and search any domain of search space with good efficiency and speed with escaping from local minima easily. For this purpose, in this paper, various test bed problems have been taken from "Congress on Evolutionary Computation (CEC 2017)", which are evaluated to compare the performance of proposed algorithm with both standard SSO and GSA.

The rest of the paper is structured as follows. Background on "Sperm Swarm Optimization (SSO)" and "Gravitational Search Algorithm (GSA)" are discussed in Sect. 2. The "Chernobyl Disaster and Chernobyl Disaster Optimizer (CDO)" is discussed in Sect. 3. Experimental and outcomes of the study are presented in Sect. 4. Discussion is shown in Sect. 5. We conclude the outcomes in Sect. 6.

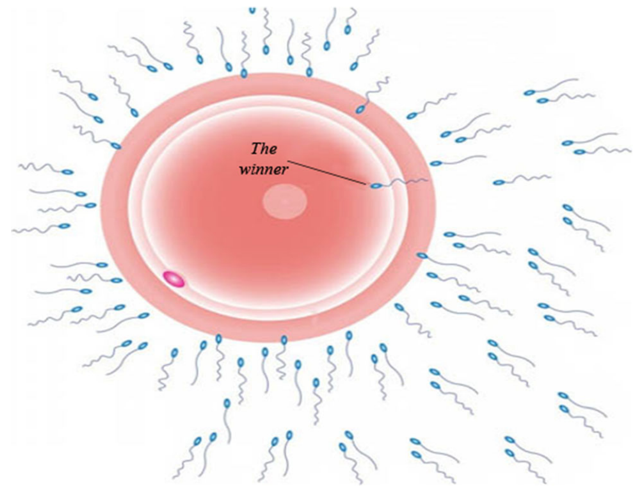


Fig. 1 Swarm of sperm and the winner (solution of global best) [5, 6]

2 "Background on standard sperm swarm optimization (SSO) and standard gravitational search algorithm (GSA)"

This section gives a brief elaboration of both standard GSA and SSO methods in which we present their structure, metaphor, and mathematical formulation.

2.1 Standard "sperm swarm optimization (SSO)"

This algorithm is created by Shehadeh et al. in 2018, which mimics the swarm of sperm in their swimming through the female reproduction system through the procedure of fertilization [1, 5]. In this algorithm, every individual in the population has three velocities, including, initial velocity of sperm, personal velocity of sperm, and global velocity of sperm. The global velocity is marked by the closest one to the egg (Ovum). As we can notice in Fig. 1, the swarm moves forward from zone of Cervix to the zone of fallopian tube where the Ovum is located. In the other meaning, the swarm swims from low to high temperature zone [6–8, 24].

$$V_i(t)=D\cdot \overbrace{Log_{10}(pH\_Rand_1)\cdot V_i}^{\text{Sperm initial velocity}}+\overbrace{Log_{10}(pH\_Rand_2)\cdot Log_{10}(Temp\_Rand_1)\cdot(x_{sbest_i}-x_i(t))}^{\text{Personal best solution}}+\underbrace{Log_{10}(pH\_Rand_3)\cdot Log_{10}(Temp\_Rand_2)\cdot(x_{sgbest}-x_i(t))}_{\text{Global best solution}} \tag{1}$$

In SSO, every sperm consider the current location and velocity as well as, the distances to x_{sbest} and x_{sgbest}. Every individual update the prior location in the memory just in case of if the current one is better than the old one. In SSO, the mathematical modelling can be formulated as in the following equation:

where v_i—is the velocity of individual i at iteration t; D—is a random factor between 0 and 1, which is a velocity damping factor; pH_Rand$_1$, pH_Rand$_2$, and pH_Rand$_3$—these factors take a value between 7 and 14, which consider as pH values of visited zone; Temp_Rand$_1$, Temp_Rand$_2$—these factors take a value between 35.1 and 38.5, which consider as temperature values of visited zone; x_i—current location of individual i at iteration t; x_{sbest}—is location of personal best of individual i at iteration t; x_{sgbest}—is location of global best of the whole swarm.

Equation (2) is used to calculate the current best solution.

$$x_i(t) = x_i(t) + v_i(t) \tag{2}$$

The SSO procedure can be summarized as in algorithm 1 [5, 6]. In SSO, the velocity and position of swarm are updated in each iteration of algorithm, which requires O ($N * m$) time, where N indicates the size of population and m indicates the discrete variable dimension.

Algorithm 1 *"Sperm Swarm Optimization (SSO)"*

Begin

Step 1: ***for every individual in the swarm, initialize the position.***

Step 2: ***for*** *i=1: size of individuals* ***do***

Step 3: *for the swarm, estimate the fitness***.**

if *obtained fitness > x_{sbest}* ***then***

Assign the x_{sbest}

end if

end for

Step 4: *assign the x_{sgbest} that is marked by the winner.*

Step 5: ***for*** *i=1: size of individuals* ***do***

Using Eq. (1), Update velocity,

*Using Eq. (2)***,** *Update position,*

end for

Step 6: ***while*** *the final criterion is not obtained jump to step 2***.**

End.

2.2 "Standard gravitational search algorithm (GSA)"

Rashedi et al. proposed this method as a physical-based method, which is inspired by the Newton's rule and theory [1]. The state of Newton's theory is "every object in the universe attracts every other object with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them" [1]. GSA method simulates this theory, which assumes a set of individuals (agents) in search space domain. Each individual in that domain generates a gravity force to attract other individual, which the heavier individual has a greatest force of attraction. The masse of heavier weight as a higher ability to be near to the optimal solution.

In GSA, the procedure begins the iterations by placing the agents randomly in the search space domain. Depends on that the forces of gravity among potential solutions (agents) i and j can be calculated as follow [1]:

$$F_{ij}^{d}(t) = G(t)\frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \varepsilon}\left(x_j^d(t) - x_i^d(t)\right), \tag{3}$$

where M_{aj}—is active mass of gravity of potential solution j; M_{pi}—is the passive mass of gravity of potential solution i; $G(t)$—is the constant of gravitational at time; R_{ij}—is the Euclidian distance among two potential solutions j and i. ε—is a constant factor.

The following formula is utilized to calculate $G(t)$ [1]:

$$G(t) = G_0 \times \exp(-a \times \text{iter}/\text{max iter}), \tag{4}$$

where G_0—is the potential value that is created initially; iter—is the iteration of current; a—coefficient of descending; maxiter—is value of final iteration.

In a dimension d of the search space of the problem, the total force that effect on potential solution i can be calculated as follows [1]:

$$F_i^d(t) = \sum_{j=1, j\neq i}^{N} \text{rand}_j F_{ij}^d(t), \tag{5}$$

where rand$_j$ is a random factor between (0, 1).

Depends on the motion low, "the acceleration of potential solution is proportional to the force and the inverse of its mass", hence the acceleration of all of potential solutions should be computed as in Eq. 6 [1]:

$$ac_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)'}, \tag{6}$$

where t—is a required time; M_{ii}—is the inertia mass of potential solution i.

The velocity along with position of potential solutions can be calculated as follows:

$$\mathrm{vel}_i^d(t+1) = \mathrm{rand}_i \times \mathrm{vel}_i^d(t) + ac_i^d(t), \quad (7)$$

$$x_i^d(t+1) = x_i^d(t) + \mathrm{vel}_i^d(t+1), \quad (8)$$

where rand_i is a value between (0, 1), which is taken randomly.

The full procedure of "Gravitational Search Algorithm (GSA)" can be stated as in algorithm 2 [1]. The computational complexity of GSA is $O(N^2)$.

Algorithm 2 *"Gravitational Search Algorithm (GSA)"*

Begin

Step 1: ***Initialize the*** potential solutions ***Xi(i = 1, 2, . . ., N)*****.**

Step 2: **while** (*end criterion is not reached*) **do**

Step 3: *Compute the fitness of potential solutions***.**

Compute Mj, Mi, worst, best,.

Update G(t).

Using Eq. (3), compute forces F_{ij}

Using Eq. (6), update acceleration

Using Eq. (7), update velocities

Using Eq. (8), update positions

Step 4: *retrieve the best potential solution(search agent)***.**

End.

3 Chernobyl disaster

Chernobyl disaster was a nuclear accident that happened in 1986. It is deemed the worst nuclear disaster in history both in casualties and cost [25]. This accident happened when the rector number four exploded due to temperature and pressure rising inside reactor core. These rectors were used as "Nuclear Power Plant (NPP)", which was supplying the electricity to the cities near to Chernobyl. This accident can be depicted in Fig. 2.

In this figure, we can notice that the unclear exploration can generate three various types of unclear radiations. These radiations can be divided into three particles, such as follows [19–29]:

- Alpha particle is created from two protons and two neutrons. In addition, alpha particle does not have electrons. Generally, this type of radiation is quite large in size, which does not penetrate far into materials. Alpha can be absorbed easily by a single human skin and sheet of paper. Moreover, alpha particle has a positive charge and large in its size, which are strongly ionizing and harmful to humans.
- Beta particle is created by unstable nucleus, which has a high energy. In the nucleus, beta particle is formed when a neutron splits from a proton and an electron, which the electron is released at a very high speed and the proton stays in the nucleus. These types of particles has a very small mass with negative charge of -1, which can be absorbed easily by a sheet of aluminium. Beta particles moderately ionizing and harmful to humans.
- Gamma rays can be considered as a waves of electromagnetic radiation with a short wavelength and a high frequency. Gamma rays are mainly emitted alongside alpha and beta particles. These types of waives have no mass. In the other meaning, it cannot collide with other atoms. Gamma rays can be absorbed easily by a thick lead. These types of particles are weakly ionizing and very harmful to humans. The aforementioned kinds of radiations after unclear explosion can be depicted in Fig. 3 [30]. The speeds of the aforementioned radiations are listed Table 1.

As aforementioned before, these particles are harmful to humans, which can make the following disease:

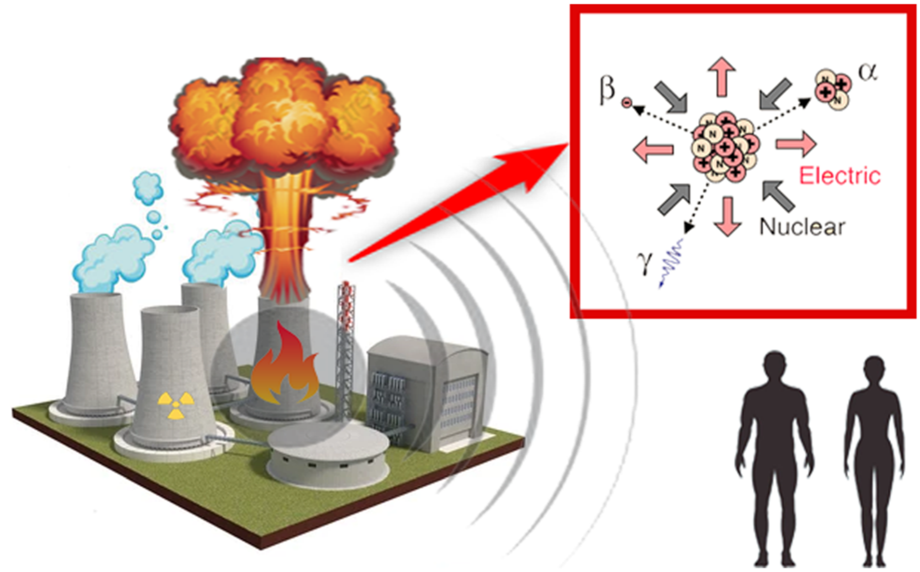
- "Acute Radiation Syndrome (ARS)" includes, fatigue, fever, loss of appetite, nausea, diarrhoea, vomiting and possibly even coma and seizures;
- Damage Thyroid;
- Blood cancer;
- Bones cancer;
- Burn skin;
- Genetic defect;
- and even death.

We can simulate this disaster as an algorithm. This algorithm can be discussed in the following subsection:

3.1 Chernobyl disaster optimizer (CDO)

As mentioned previously, there are three particles attack human after the rector number four exploded, which the explosion area can be depicted in Fig. 4. After explosion, three types of radiations are omitted from the nuclei, which are alpha, beta and gamma. These particles will travel far away from the core of rector (high pressure area) until reach the human areas (low pressure area), which the disaster will happened. We suppose that the victims (humans) are walking and these particles are attacking them at the

Fig. 2 Chernobyl accident



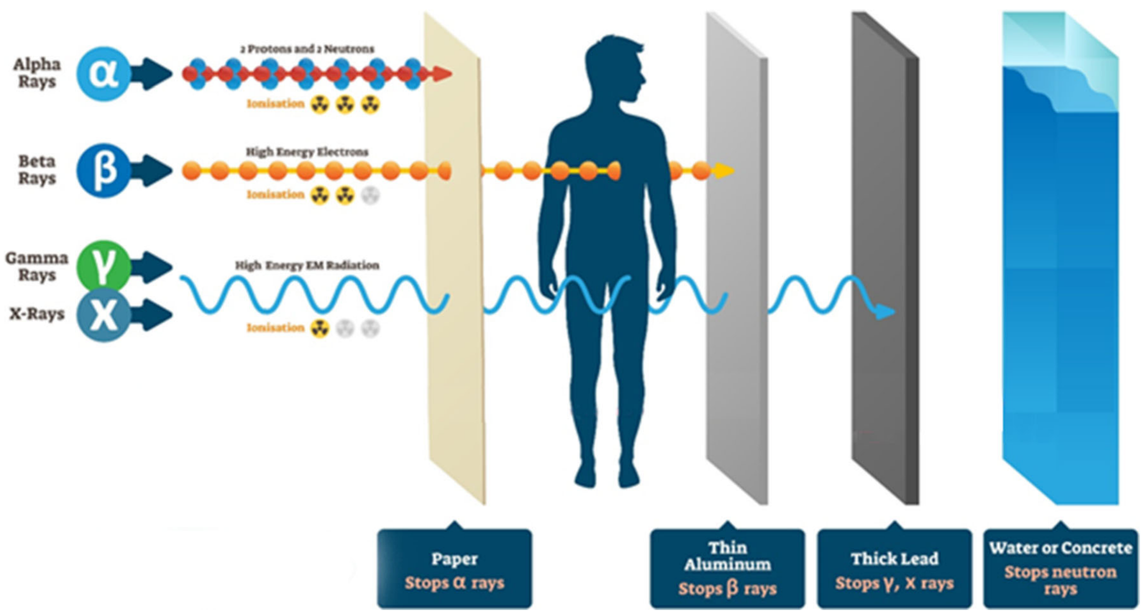


Fig. 3 Kinds of radiations after unclear explosion [30]

Table 1 The speeds of radiations [31, 32]

Radiation type	Speed of particle ($S_{\text{type of particle}}$)
Alpha (α)	16,000 km per second
Beta (β)	270,000 km per second
Gamma (γ)	300,000 km per second

same time. The positions of particles and human while the process of attack can be depicted in Fig. 5.

The walking speed of adults outdoor can be increased, which can be calculated in range of (0–3 miles per hour) [33]. Based on that we can simulate this speed to be linearly decreased from 3 to 0 as in the following equation.

$$\mathrm{WS}_h = 3 - 1 * ((3)/\text{Maximum\_Iteration}) \tag{9}$$

4 Gamma particle

We can calculate the gradient descent factor of gamma particle (v_γ) while attacking human as in the following equation.

$$v_\gamma = \left(X_\gamma(t) - \rho_\gamma.\Delta_\gamma\right) \tag{10}$$

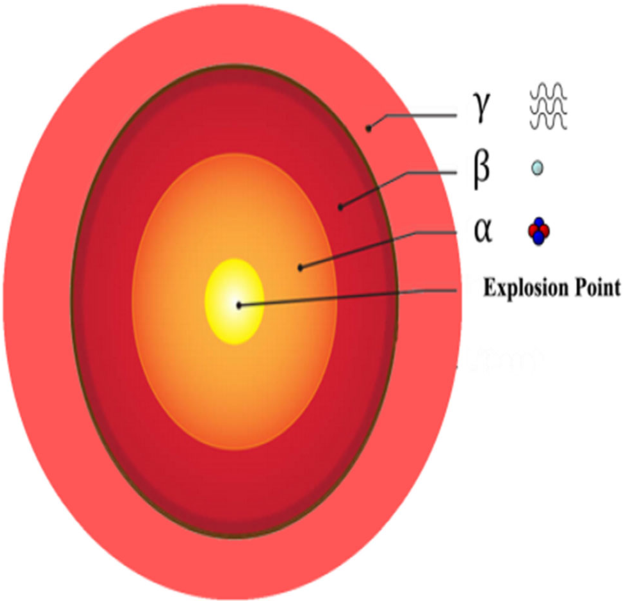


Fig. 4 Explosion point and the omitted radiations

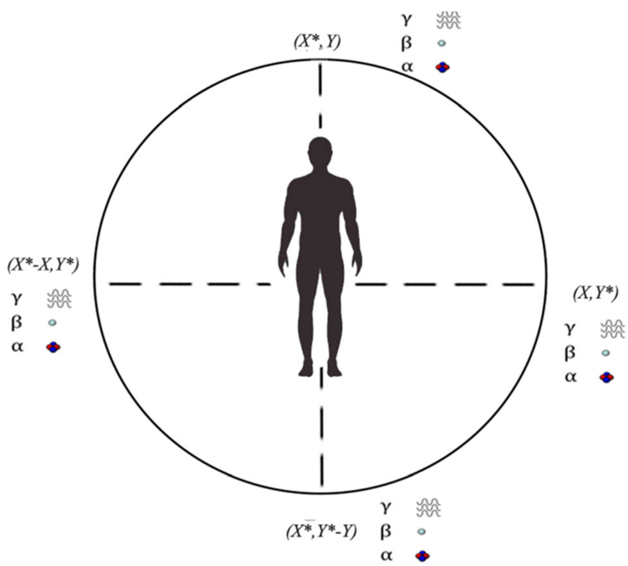


Fig. 5 The positions of particles and human while the process of attack

where $X_\gamma(t)$ is the current position of gamma particle; ρ_γ is the propagation of (γ) particle; Δ_γ is the difference between human position and position of gamma particles.

By Eq. (11), the ρ_γ can be calculated.

$$\rho_\gamma = \frac{x_h}{S_\gamma} - (WS_h \cdot \text{rand}()) \tag{11}$$

where x_h is the area of human walking, which is the area of circle and can be calculated as in Eq. (12), S_γ is the speed of (γ) particle, which can take a value randomly from 1 to 300,000 km per second as presented in Table 1. To normalize this value, we take the logarithm of it as presented in Eq. (13).

$$x_h = r^2 \cdot \pi \tag{12}$$

where r is a value between (0, 1), which is taken randomly.

$$S_\gamma = \log(\text{rand}(1 : 300{,}000)) \tag{13}$$

The Δ_γ is the difference between (γ) particle position and total position can be calculated in the following equation.

$$\Delta_\gamma = \left|A_\gamma . X_\gamma(t) - X_T(t)\right| \tag{14}$$

where A_γ is the area of propagation of beta particle, which is the area of circle and can be calculated as in the following equation: x_T is the averages of total positions, which can be calculated in Eq. (28).

$$A_\gamma = r^2 \cdot \pi \tag{15}$$

where r is a value between (0, 1), which is taken randomly.

5 Beta particle

We can calculate the gradient descent factor of beta particle (v_β) while attacking human as in the following formula.

$$v_\beta = 0.5 \cdot \left(X_\beta(t) - \rho_\beta . \Delta_\beta\right) \tag{16}$$

where $X_\beta(t)$ is the current location of particle of beta; ρ_β is the propagation of (β) particle; Δ_β is the difference between human position and position of beta particles.

By Eq. (17), the ρ_β can be calculated.

$$\rho_\beta = \frac{x_h}{0.5 \cdot S_\beta} - (\text{WS}_h \cdot \text{rand}()) \tag{17}$$

where x_h is the area of human walking, which is the area of circle and can be calculated as in Eq. (18). S_α is the speed of (α) particle, which can take a value randomly from 1 to 270,000 km per second as presented in Table 1. To normalize this value, we take the logarithm of it as presented in Eq. (19).

$$x_h = r^2 \cdot \pi \tag{18}$$

where r is a random number between 0 and 1.

$$S_\beta = \log(\text{rand}(1 : 270{,}000)) \tag{19}$$

The Δ_α is the difference between (α) particle position and total position can be calculated in the following equation.

$$\Delta_\beta = \left|A_\beta . X_\beta(t) - X_T(t)\right| \tag{20}$$

where x_T is the averages of total positions, which can be calculated in Eq. (28). A_β is the area of propagation of beta particle, which is the area of circle and can be calculated as in the following equation:

$$A_\beta = r^2 \cdot \pi \tag{21}$$

where r is a random number between 0 and 1.

6 Alpha particle

We can calculate the gradient descent factor of alpha particle (v_a) while attacking human as in the following equation.

$$v_a = 0.25 \cdot (X_a(t) - \rho_a . \Delta_\alpha) \tag{22}$$

where $X_\alpha(t)$ is the current position of alpha particle; ρ_a is the propagation of (α) particle; Δ is the difference between human position and position of alpha particles. By Eq. (23), the ρ_a can be calculated.

$$\rho_a = \frac{x_h}{0.25 \cdot S_a} - (\mathrm{WS}_h \cdot \mathrm{rand}()) \tag{23}$$

where x_h is the area of human walking, which is the area of circle and can be calculated as in Eq. (24). S_α is the speed of (α) particle, which can take a value randomly from 1 to 16,000 km per second as presented in Table 1. To normalize this value, we take the logarithm of it as presented in Eq. (25):

$$x_h = r^2 \cdot \pi \tag{24}$$

where r is a value between (0, 1), which is taken randomly.

$$S_a = \log(\mathrm{rand}(1 : 16{,}000)) \tag{25}$$

The Δ_α is the difference between (α) particle position and total position can be calculated in the following model.

$$\Delta_\alpha = |A_\alpha . X_a(t) - X_T(t)| \tag{26}$$

where x_T is the averages of total positions, which can be calculated in Eq. (28). A_α is the area of propagation of alpha particle, which is the area of circle and can be calculated as in the following equation:

$$A_\alpha = r^2 \cdot \pi \tag{27}$$

where r is a random number between 0 and 1.

Based on the *Galileo Galilei* equations of motion, we can take the average of total speed of these particles by using the following equation.

$$x_T = \frac{(v_a \cdot v_\beta \cdot v_\gamma)}{3} \tag{28}$$

Finally, the full procedure of CDO can be summarized in the Algorithm 3. The total time complexity of CDO is $O(N \times m)$ time, where N indicates the population size and m represents the problem dimension. Figure 6 illustrates the optimization process of CDO, first the optimization problem should be converted and simplified to mathematical model (optimization model) by determining the limitation, and quantification of the problem. Second, the CDO take the step on the procedure by calculating the gradient descent factor of alpha, beta, and gamma particles in which based on these factors, the algorithm will find the optimal result of the optimization model. Third, based on the results evaluation, the optimization model can be modified to represents the exact real-life problem.

Algorithm 3 *Chernobyl Disaster Optimizer (CDO)*

Begin

Step 1: ***Initialize the particles Xi (i = 1, 2, . . ., N).***

Step 2: while (*the end iteration is not achieved*) **do**

Step 3: *For all particles α,β,γ, calculated the fitness*.

If fitness<gammaScore

    gammaScore ← fitness

    update position of gamma particle.

    EndIf

If fitness<betaScore

    betaScore ← fitness

    update position of beta particle.

    EndIf

If fitness<alphaScore

    alphaScore ← fitness

    update position of alpha particle.

    EndIf

Step 4: For *all particles α,β,γ*, **update position on the Cartesian plan(x, y)**

    Calculate gradient descent factor of gamma particle (γ)

    by Eq.10.

    Calculate gradient descent factor of beta particle (β) by

    Eq.16

    Calculate gradient descent factor of alpha particle (α)

    by Eq.22

Step 5: *Update average of total positions by Eq. (28).*

End.

7 Experimental and result

The efficiency of the proposed "Chernobyl Disaster Optimizer (CDO)" is evaluated using suites of 23 mathematical optimization problems, which are taken form the well-

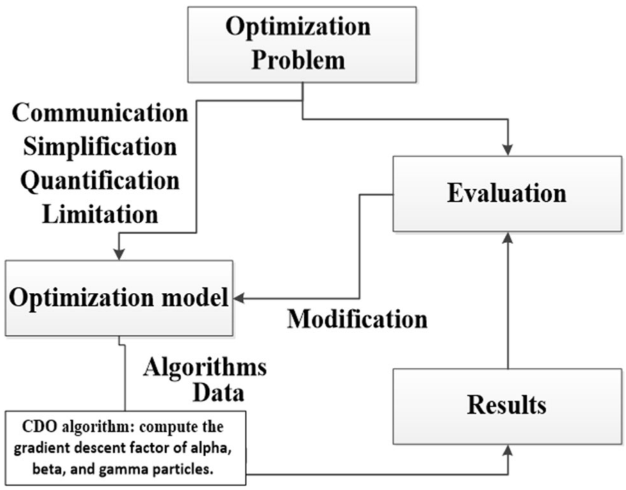


Fig. 6 Optimization process of optimization problem based on CDO

known "Congress on Evolutionary Computation (CEC 2017)" [1]. The mathematical modelling, ranges, and dominations of these test suites are stated in Appendix A. GSA, SSO, and the proposed CDO are coded in "MATLAB R2017a" and run on Intel core i7 CPU, 6 GB RAM utilizing Windows 10.

Overall, in total of (10) time runs are evaluated for every method. The test bed problems in Appendix A are minimization problems, which the optimal values for F_1 to F_{12} expect F_8 are zero. On the other hand, the optimal value of F_8 is $-$ 12,569.5. In addition, the optimal values of F_{13} to F_{23} are $-$ 1.15044, 0.998, 0.0003075, $-$ 1.0316, 0.398, 3, $-$ 3.86, $-$ 3.32, $-$ 10.2, $-$ 10.4, $-$ 10.5, and $-$ 12,569.5. The following criteria are evinced to evaluate the proposed CDO:

- Mean (μ): after a number of generations T, the average of fitness values can be calculated as follows:

$$\text{Mean}(\mu) = \frac{\sum_{t=1}^{n} (f_t)}{T}, \tag{29}$$

- Standard deviation (σ): after executing the approach T times, the standard deviation between individuals can be calculated using Eq. (30). This factor is the variances between values of objective model.

$$\text{Std}(\sigma) = \sqrt{\frac{t}{T-1}\sum_{t=1}^{n} (f_t - \text{mean})^2}, \tag{30}$$

- Optimal value (best fitness): after executing the approach T times, the minimum fitness value can be calculated as follows:

$$\text{Best} = \min_{1 \le i \le T} f_i, \tag{31}$$

The quality of outcome and general performance of the proposed CDO are compared against SSO and GSA methods in terms of mean (μ), standard deviation (σ), and value of optimal (best fitness) which the optimal values are recorded in the last iteration of the procedure. The proposed method, GSA, and SSO have different parameters in which are initialized at the beginning of the procedures. These parameters are listed in Table 2.

In Tables from three to eight, we can summarize the experimental results, which the best fitness is summarized in the highlighted background. The results of SSO and GSA are taken from [1]. The experiments are repeated ten-time-runs to ensure the convergence of outcomes. Statistically speaking, for the optimal best values, the CDO is the best on twelve benchmark functions, GSA is the significant on (8) benchmark functions, and SSO is the best on ten benchmark functions. In addition, for the average values, the CDO is the best on five benchmark functions, GSA is the significant on (8) benchmark functions, and SSO is the best on twelve benchmark functions. Moreover, CDO obtains the global minima in all the test suites instead of 5, 6, 7, 8, 10, 12, 13, 14, 21, 22, and 23. CDO outperforms GSA and SSO methods in solving the noisy and rugged problems, such as problems from one to four. From benchmark functions from 16 to 19, the CDO can obtain the global minima as SSO and GSA, which the all have narrow domains. For functions 9, 11, and 16, SSO and CDO perform in the same rank. On the other hand, for the

Table 2 Parameters of the GSA, CDO, and SSO

Parameters	Value
SSO	
Damping factor of velocity (*D*)	Rand (0, 1)
pH	Rand (7, 14)
Temperature	Rand (35.5, 38.5)
Size of population (swarm size)	30
Numbers of iterations/generations	1000
GSA	
α	20
G_0	1
Size of population	30
Numbers of iterations/generations	1000
CDO	
S_γ is the speed of gamma	Rand (1, 300,000) km/s
S_β is the speed of beta	Rand (1, 270,000) km/s
S_α is the speed of alpha	Rand (1, 16,000) km/s
r is the radius of radiations propagation	Rand (0, 1)
Size of population	30
Numbers of iterations/generations	1000

Table 3 CDO, GSA, and SSO numerical results of test suites

Problem number	SSO [1] Best fitness	GSA [1] Best fitness	CDO Best fitness
(1)	7.58E−228	1.01E−16	*2.29E−262*
(2)	1.24E−129	7.53E−08	*2.79E−135*
(3)	5.68E−111	5.79E+02	*1.83E−226*
(4)	1.14E−90	2.470583	*1.52E−126*
(5)	28.08445	*26.83491*	27.23930087
(6)	*4.620426*	297.666	7.5
(7)	*9.24E−06*	0.072594	3.19E−05
(8)	*−5846.46*	−3415.7	−3720.669834

Table 5 CDO, GSA, and SSO numerical results of test suites

Problem number	SSO [1] Best fitness	GSA [1] Best fitness	CDO Best fitness
(9)	*0.0000*	42.78318	*0.0000*
(10)	*8.88E−16*	7.59E−09	4.44E−15
(11)	*0.0000*	6.332879	*0.0000*
(12)	*0.541285*	0.748805	1.106102428
(13)	2.474063	*0.010987*	0.294755937
(14)	*0.998011*	1.995132	2.982107311
(15)	*0.00031*	0.002306	*0.00031*
(16)	*−1.0316*	*−1.0316*	*−1.0316*

test beds 1, 2, 3, and 4 the CDO beats better than GSA and SSO as outlined in the tables listed below. Depends on that it can be concluded that CDO beats better on test bed problems with a domain of wide search space.

This is obvious in the experimental results and rates of convergence of the method as depicted in the aforementioned figures of test suites 1, 2, 3, 4, 9, 11, 15, 16, 17, 18, 19, and 20 in which CDO is superior and faster of reaching and obtaining the global minima of these test beds than the other approaches. Ranking the methods from best to worse achieved fitness value can be listed in Table 9.

8 Discussion

GSA is very efficient approach that is inspired by physical metaphor. However, this approach is very slow, which its affect ability in the area of exploitation is not huge. On the other hand, SSO is a novel and new method that is inspired by swarm metaphor. This approach can be slightly fallen into local minimum, but it has powerful and precise ability of exploitation.

Depends on the aforementioned drawbacks of SSO and GSA, we are motivated to propose new optimization algorithm in this paper. This method is inspired by the nuclear reactor core explosion of Chernobyl. Through the process of nuclear explosion, there are different radiations in which are emitted from nuclei. The most common kinds of these radiations are called, gamma, beta, and alpha particles. These particles fly away from the explosion point (high pressure point) to the low pressure point (the human standing point). These particles can attack human and penetrate their body, which are harmful to the humans. These radiations can make different disease, such as blood cancer, bones cancer, and burn skin. Based on that, in this paper, the main steps of nuclear explosion and attaching human are implemented in which gamma, beta, and alpha particles are involved in this process.

Test suites of 23 problems are selected from CEC 2017 to compare the proposed approach against SSO and GSA. Both experimental numerical and statistical outcomes are calculated in the evaluation, called mean (μ), standard deviation (σ), and optimal value (best fitness). Moreover, the convergence rate has been drawn for each method of each test bed problem. The iteration end criterion for each

Table 4 CDO, GSA, and SSO statistical results of test suites

Problem number	SSO [1]		GSA [1]		CDO	
	μ	σ	μ	σ	μ	σ
(1)	*146.1002*	2452.828	2.11 E+02	2725.425	2.81E+03	9218.187043
(2)	3.78E+08	1.2E+10	*1.13E+07*	3.58E+08	2.14E+10	6.1833E+11
(3)	*486.0306*	4991.478	1.03E+03	4559.733	8.58E+04	100,057.5951
(4)	*0.862714*	6.270643	3.72E+00	5.20553	1.10E+01	23.72986324
(5)	679,417.8	10,241,293	*2.85E+05*	7,657,691	7.46E+06	26,860,036.51
(6)	*1.111483E+2*	1959.556	4.96E+02	3336.817	3.59E+03	11,301.18413
(7)	*0.267079*	5.303608	6.82E+00	22.82138	4.43E+00	16.00126141
(8)	*−5830.63*	113.964	3.41E+03	61.87957	−3.66E+03	109.3767252

Table 6 CDO, GSA, and SSO statistical results of test suites

Problem number	SSO [1]		GSA [1]		CDO	
	μ	σ	μ	σ	μ	σ
(9)	*3.45444643*	32.17159	6.89E+01	63.1026	1.82E+02	132.4407183
(10)	*0.205431505*	1.46341	6.31E−01	1.850071	2.79E+00	6.369129417
(11)	*1.827365617*	25.06437	1.44E+01	49.52972	3.60E+01	110.5403913
(12)	1,894,427.79	30,251,324	*7.87E+05*	20,086,708	1.33E+07	53,182,172.98
(13)	3.98E+06	59,904,860	*1.64E+06*	40,544,592	2.13E+07	86,557,208.63
(14)	*1.046625709*	0.451234	2.43E+00	13.57987	5.42E+00	0.234957768
(15)	1.42E−03	0.010285	6.21E−03	0.011477	*1.27E−03*	0.025089272
(16)	−1.03E+00	0.07725	−1.02E+00	0.088165	*−9.93E−01*	0.102033521

Table 7 CDO, GSA, and SSO numerical results of test suites

Problem number	SSO [1] Best fitness	GSA [1] Best fitness	CDO Best fitness
(17)	*0.397*	*0.397*	*0.397*
(18)	*3*	*3*	*3*
(19)	*−3.86*	*−3.86*	*−3.86*
(20)	−2.99659	*−3.32*	*−3.32*
(21)	−4.11734	*−10.1532*	−9.336480824
(22)	−3.21942	*−10.4029*	−7.940198741
(23)	−4.27035	*−10.5364*	−8.657742972

method is set to one thousand. To ensure the convergence of the outcomes, the evaluation is repeated ten-time-runs for each test bed problems.

Overall, the proposed CDO outperformed both SSO and GSA methods on the majority of the benchmark problems of narrow and wide domain, which has efficiency and merit in term of exploration. This is very clear in the tables listed below of results and the method algorithm convergence in figures of test bed problems of 9, 11, 15, 16, 17, 18, 19, 20, and especially for noisy and rugged problems, such as problems from one to four. This evinced the superiority of the proposed method in terms of convergence in comparison with the other methods to export the problem domain.

In this study, the test suites may have their drawbacks and limitations in which various tuning parameters in real experiments may available to change the outcomes. Hence, the proposed CDO should be set to test real tasks in the future to ensure its effectiveness, which most of real-life problems can be considered as multi-model functions. These objective functions have a set of local minima/maxima and a single global minima/maxima.

In the future, we will set the proposed method to solve real-life problems, which has the ability and potential to overcome tasks of wide domain. Examples of these tasks are maximizing the propagation of signal in “Radio Networks (RNs)”, and minimizing both attenuation and interference factors in “Wireless Sensor Network (WSN)”, minimizing the path loss and maximizing the power density of the “Underwater Wireless Sensor Networks (UWSNs)” [34–37]. Furthermore, this algorithm will be merged with “Density based Feature Selection (DFS)” to work on feature selection field in the future, which will be incorporated with wrapper method to eradicate redundant or irrelevant features of different datasets in various applications. In addition, this algorithm will be used to predict and classify a set of diseases, such as Parkinson’s disease, heart disease, and kidney disease.

Table 8 CDO, GSA, and SSO statistical results of test suites

Problem number	SSO [1]		GSA [1]		CDO	
	μ	σ	μ	σ	μ	σ
(17)	4.10E−01	0.186291	4.08E−01	0.043303	*4.05E−01*	0.088137775
(18)	*3.30E+00*	4.947175	3.59E+00	2.782326	5.87E+00	2.496927259
(19)	*−3.85E+00*	0.022962	*−3.85E+00*	0.075046	*−3.85E+00*	0.019301841
(20)	−2.83E+00	0.302575	−3.24E+00	0.207765	*−3.26E+00*	0.112827492
(21)	−3.75E+00	0.876071	*−−9.14E+00*	2.654223	−8.43E+00	1.93810293
(22)	−2.94E+00	0.413207	*−9.40E+00*	2.65385	−7.39E+00	1.49386918
(23)	−3.79E+00	0.937543	*−9.38E+00*	2.793826	−7.60E+00	1.408646007

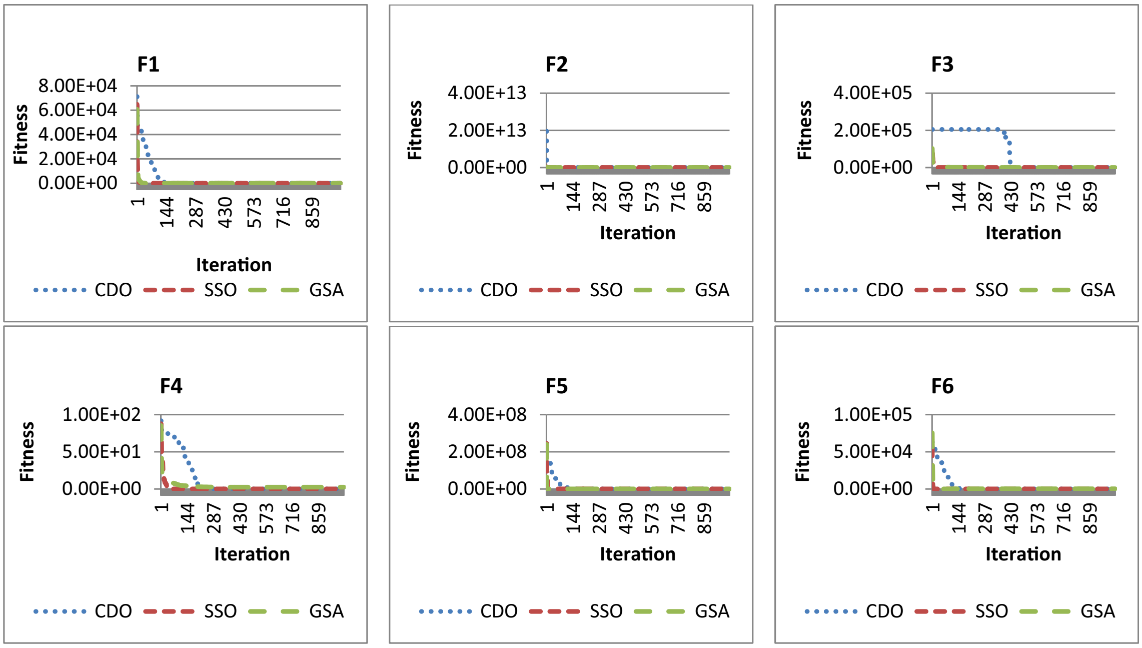


Fig. 7 Comparison of convergence rate between CDO, GSA, and SSO for the functions 1 to 6

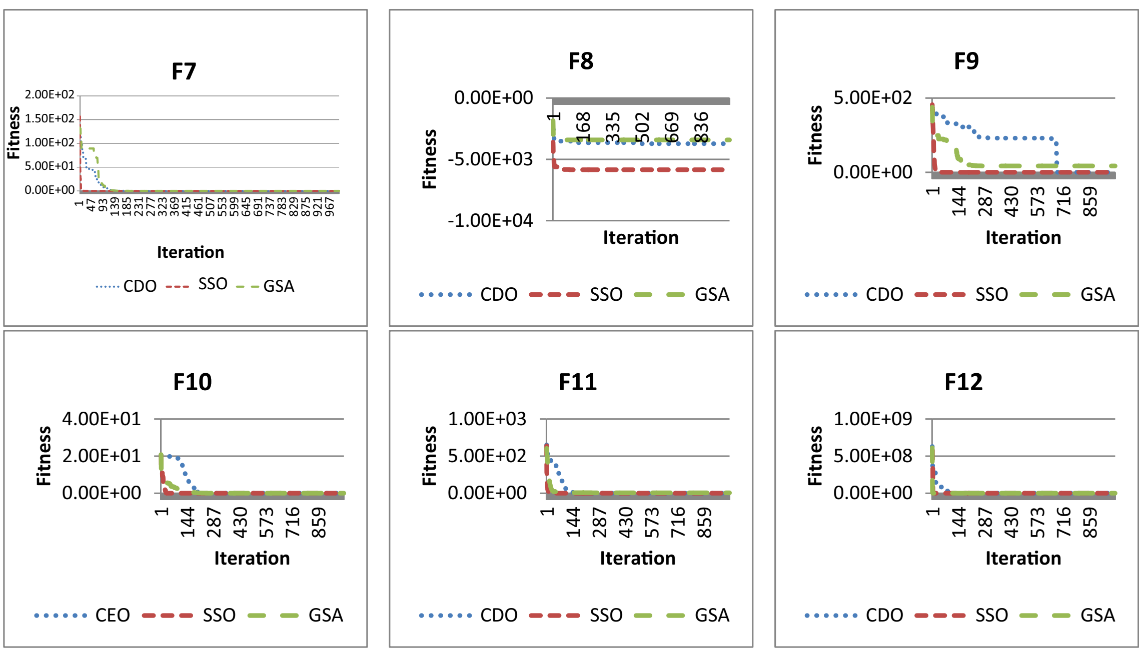


Fig. 8 Comparison of convergence rate between CDO, GSA, and SSO for the functions 7 to 12

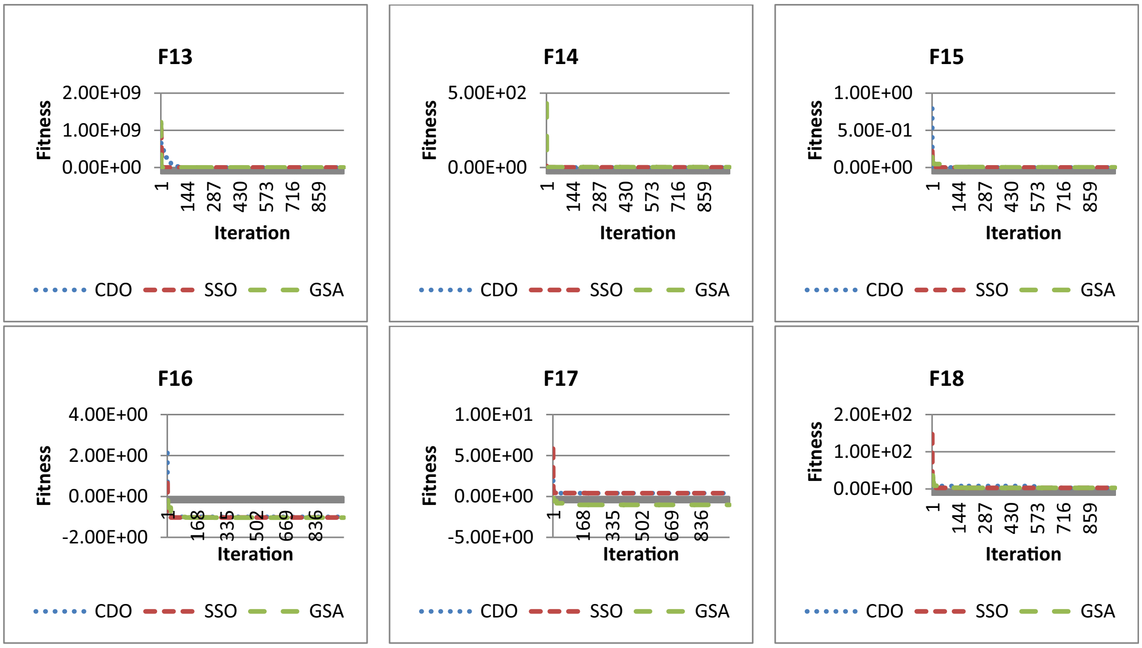


Fig. 9 Comparison of convergence rate between CDO, GSA, and SSO for the functions 13 to 18

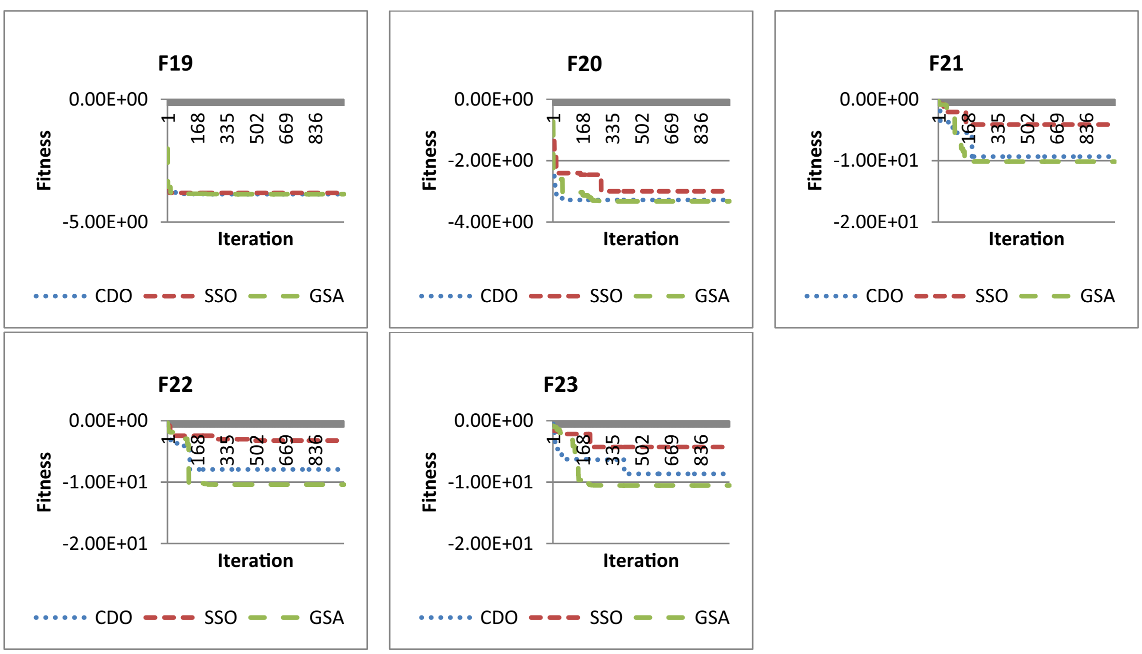


Fig. 10 Comparison of convergence rate between CDO, GSA, and SSO for the functions 19 to 23

Table 9 Ranking the approaches from best to worse obtained fitness value

Function	Based on the fitness values, we can rank the methods from best to worse as follows
(1)	CDO, SSO, GSA
(2)	CDO, SSO, GSA
(3)	CDO, SSO, GSA
(4)	CDO, SSO, GSA
(5)	GSA, CDO, SSO
(6)	SSO, CDO, GSA
(7)	SSO, CDO, GSA
(8)	SSO, CDO, GSA
(9)	CDO and SSO in the first rank, followed by GSA
(10)	SSO, CDO, GSA
(11)	CDO and SSO in the first rank, followed by GSA
(12)	SSO, GSA, CDO
(13)	GSA, CDO, SSO
(14)	SSO, GSA, CDO
(15)	CDO and SSO in the first rank, followed by GSA
(16)	Same rank for all
(17)	Same rank for all
(18)	Same rank for all
(19)	Same rank for all
(20)	CDO and GSA in the first rank, followed by SSO
(21)	GSA, CDO, SSO
(22)	GSA, CDO, SSO
(23)	GSA, CDO, SSO

9 Conclusion

In this article, a novel optimization method, called "Chernobyl Disaster Optimizer (CDO)", has been proposed. The main concept of this approach is inspired by the nuclear reactor core explosion of Chernobyl where there are three types of radiations are emitted from nuclei. These radiations (particles) are alpha, beta, and gamma, which attack humans and harm them. Well-known test bed of twenty-three problems has been selected in order to compare the proposed approach against standard SSO and GSA. Those methods are compared based on couple of techniques, called quantitative and qualitative studies. For the quantitative technique, mean (μ), standard deviation (σ), and optimal value (best fitness) are calculated. On the other hand, for the qualitative technique, the convergence rate is drown for each test bed problem based on each the outcome of each method. Depends on the results in Tables from 3, 4, 5, 6, 7, 8 and Figs. from 7, 8, 9 and 10, CDO outperformed the other methods in solving test bed problems of 9, 11, 15, 16, 17, 18, 19, 20, and especially for noisy and rugged problems, such as problems from 1 to 4. Moreover, the convergence rate of the aforementioned figures evinces the speed rate of the CDO. This is clear in the convergence curve of test bed problems 1, 2, 3, 4, 9, 11, 15, 20. We can conclude that the CDO is a feasible approach and alternative since it outperformed GSA and SSO in most test bed problems of a noisy and rugged domain of search space.

Appendix A

Table 10 shows the mathematical formulation of 23 test bed problems of a well-known "Congress on Evolutionary Computation (CEC 2017)" test bed suites.

Table 10 Problems of CEC 2017 test bed suites

Function	Dim	Range
$F_1(z) = \sum_{i=1}^{d} z_i^2,$	30	[−100, 100]
$F_2(z) = \sum_{i=1}^{z} \lvert z_i \rvert + \prod_{i=1}^{d} \lvert z_i \rvert,$	30	[−10, 10]
$F_3(z) = \sum_{i=1}^{d} \left(\sum_{j-1}^{i} z_j \right)^2,$	30	[−100, 100]
$F_4(z) = \max_i \{ \lvert z_i \rvert, 1 \leq i \leq d \},$	30	[−100, 100]
$F_5(z) = \sum_{i=1}^{d-1} \left[100\left(z_{i+1} - z_i^2\right)^2 + (z_i - 1)^2 \right],$	30	[−30, 30]
$F_6(z) = \sum_{i=1}^{d} ([z_i + 0.5])^2,$	30	[−100, 100]
$F_7(z) = \sum_{i=1}^{d} i z_i^4 + random[0, 1),$	30	[−1.28, 1.28]
$F_8(z) = \sum_{i=1}^{d} -z \sin(\sqrt{\lvert z_i \rvert}),$	30	[−500, 500]
$F_9(z) = \sum_{i=1}^{d} [z_i^2 - 10\cos(2\pi z_i) + 10],$	30	[−5.12, 5.12]
$F_{10}(z) = -20 \exp\left(-0.2 \sqrt{\frac{i}{z} \sum_{i=1}^{d} z_i^2} \right) - \exp\left(\frac{i}{z} \sum_{i=1}^{d} \cos(2\pi z_i) \right) + 20 + e,$	30	[−32, 32]
$F_{11}(z) = \frac{i}{4000} \sum_{i=1}^{d} z_i^2 - \prod_{i=1}^{d} \cos\left(\frac{z_i}{\sqrt{i}} \right) + 1,$	30	[−600, 600]
$F_{12}(z) = \frac{\pi}{d} \left\{ 10\sin(\pi z_1) + \sum_{i=1}^{d-1} (v_i - 1)^2 [1 + 10\sin^2(\pi v_{i+1})] + (v_z - 1)^2 \right\} + \sum_{i=1}^{z} u(z_i, 10, 100, 4),$	30	[−50, 50]
$F_{13}(z) = 0.1 \left\{ \begin{array}{r} \sin^2(3\pi z_1) + \\ \sum_{i=1}^{d} (z_i - 1)^2 [1 + \sin^2(3\pi z_i + 1)] + (z_d - 1)^2 [1 + \sin^2(2\pi z_d)] + \sum_{i=1}^{z} u(z_i, 5, 100, 4) \end{array} \right\}$	30	[−50, 50]
$F_{14}(z) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2} (z_i - a_{ij})^6} \right)^{-1},$	2	[−62.536, 65.536]
$F_{15}(z) = \sum_{i=1}^{11} \left[a_i - \frac{y_{1(b_i^2 + b_i z_2)}}{b_i^2 + b_i z_3 + z_4} \right]^2,$	4	[−5, 5]
$F_{16}(z) = 4z_1^2 - 2.1z_1^4 + \frac{1}{3} z_1^6 + z_1 z_2 - 4z_2^2 + 4z_2^4,$	2	[−5, 5]
$F_{17}(z) = \left(z_2 - \frac{5.1}{4\pi^2} z_1^2 + \frac{5}{\pi} z_1 - 6 \right)^2 + 10\left(1 - \frac{1}{8\pi}\right) \cos z_1 + 10,$	2	[−5, 5]
$F_{18}(z) = [1 + (z_1 + z_2 + 1)^2 (19 - 14z_1 + 3z_1^2 - 14z_2 + 6z_1 z_2 + 3z_2^2)] \times [30 + (2z_1 - 3z_2)^2 \times (18 - 32z_1 + 12z_1^2 + 48z_2 - 36z_1 z_2 + 27z_2^2)],$	2	[−2,2]
$F_{19}(z) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{3} a_{ij} (z_j - p_{ij})^2 \right),$	3	[1,3]
$F_{20}(z) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{6} a_{ij} (z_j - p_{ij})^2 \right),$	6	[0,1]
$F_{21}(z) = -\sum_{i=1}^{5} \left[(z - a_i)(z - a_i)^T + c_i \right]^{-1},$	4	[0,10]
$F_{22}(z) = -\sum_{i=1}^{7} \left[(z - a_i)(z - a_i)^T + c_i \right]^{-1},$	4	[0,10]
	4	[0,10]

Acknowledgements There is no any fund for this research.

Declarations

Conflict of interest The authors declare that there are no conflicts of interest regarding the publication of this paper.

Consent and data availability statement: We give our consent for the publication of identifiable details, which can include photograph(s) and/or videos and/or case history and/or details within the text ("Material") to be published in the above Journal and Article. We confirm that we have seen and been given the opportunity to read both the Material and the Article (as attached) to be published by your journal. In Addition, a sample of data of this paper will be available upon request. The code of our algorithm, namely, SSO is available via the following link: https://www.mathworks.com/matlabcentral/fileexchange/92150-sperm-swarm-optimization-sso, https://www.mathworks.com/matlabcentral/fileexchange/92130-hssogsa, https://www.springerprofessional.de/en/a-hybrid-sperm-swarm-optimization-and-gravitational-search-algor/18968734.

References

1. Shehadeh HA (2021) A hybrid sperm swarm optimization and gravitational search algorithm (HSSOGSA) for global optimization. Neural Comput Appl 33(18):11739–11752. https://doi.org/10.1007/s00521-021-05880-4
2. Shehadeh HA, Idris MYI, Ahmedy I (2017) Multi-objective optimization algorithm based on sperm fertilization procedure (MOSFP). Symmetry 9(10):241. https://doi.org/10.3390/sym9100241
3. Devaraj R, Mahalingam SK, Esakki B, Astarita A, Mirjalili S (2022) A hybrid GA-ANFIS and F-race tuned harmony search algorithm for multi-response optimization of non-Traditional machining process. Expert Syst Appl 199:116965. https://doi.org/10.1016/j.eswa.2022.116965
4. Ji J, Xiao H, Yang C (2021) HFADE-FMD: a hybrid approach of fireworks algorithm and differential evolution strategies for functional module detection in protein-protein interaction networks. Appl Intell 51(2):1118–1132. https://doi.org/10.1007/s10489-020-01791-4
5. Shehadeh HA, Ahmedy I, Idris MYI (2018) Sperm swarm optimization algorithm for optimizing wireless sensor network challenges. In: Proceedings of the ACM international conference on communications and broadband networking (ICCBN), Singapore, ACM, pp 53–59.https://doi.org/10.1145/3193092.3193100
6. Shehadeh HA, Ahmedy I, Idris MYI (2018) Empirical study of sperm swarm optimization algorithm. In: Arai K, Kapoor S, Bhatia R (eds) Book: volume 869 of the advances in intelligent systems and computing series. In Proceedings of SAI intelligent systems conference. Springer, Cham, pp 1082–1104. https://doi.org/10.1007/978-3-030-01057-7_80
7. Mittal H, Tripathi A, Pandey AC, Pal R (2021) Gravitational search algorithm: a comprehensive analysis of recent variants. Multimedia Tools Appl 80(5):7581–7608. https://doi.org/10.1007/s11042-020-09831-4
8. Shehadeh HA, Mustafa HM, Tubishat M (2022) A Hybrid genetic algorithm and sperm swarm optimization (HGASSO) for multimodal functions. Int J Appl Metaheur Comput (IJAMC) 13(1):1–33. https://doi.org/10.4018/ijamc.292507
9. Shehadeh HA, Shagari NM (2022) A hybrid grey wolf optimizer and sperm swarm optimization for global optimization. In: Manshahia MS, Kharchenko V, Munapo E, Thomas JJ, Vasant P (eds) Handbook of intelligent computing and optimization for sustainable development, vol 1, pp 487–507.https://doi.org/10.1002/9781119792642.ch24
10. Khajehzadeh M, Kalhor A, Tehrani MS, Jebeli M (2022) Optimum design of retaining structures under seismic loading using adaptive sperm swarm optimization. Struct Eng Mech 81:93–102. https://doi.org/10.12989/sem.2022.81.1.093
11. Sundararaju N, Vinayagam A, Veerasamy V, Subramaniam G (2022) A Chaotic search-based hybrid optimization technique for automatic load frequency control of a renewable energy integrated power system. Sustainability 14(9):5668. https://doi.org/10.3390/su14095668
12. Khajehzadeh M (2022) Earth slope stability evaluation subjected to earthquake loading using chaotic sperm swarm optimization. Arab J Geosci 15(15):1–13. https://doi.org/10.1007/s12517-022-10633-1
13. Concepcion R, Janairo AG, Baun JJ, Cuello J, Dadios E, Vicerra RR, Bandala A (2022) Differential effects of potassium chloride on vascular tissues, morphological traits and germination of tomato with sperm swarm-based nutrient optimization. Trends Sci 19(14):1990. https://doi.org/10.48048/tis.2022.1993
14. Cvetkovski G, Petkovska L (2022) Optimal solution of PM synchronous motor obtained by gravitational search algorithm. Int J Appl Electromag Mech 69(2):149–167. https://doi.org/10.3233/JAE-210178
15. Li C, An X, Li R (2015) A chaos embedded GSA-SVM hybrid system for classification. Neural Comput Appl 26:713–721. https://doi.org/10.1007/s00521-014-1757-z
16. Kumar V, Kumar D (2019) Automatic clustering and feature selection using gravitational search algorithm and its application to microarray data analysis. Neural Comput Appl 31:3647–3663. https://doi.org/10.1007/s00521-017-3321-0
17. Taradeh M, Mafarja M, Heidari AA, Faris H, Aljarah I, Mirjalili S, Fujita H (2019) An evolutionary gravitational search-based feature selection. Inf Sci 497:219–239. https://doi.org/10.1016/j.ins.2019.05.038
18. Shilaja C, Arunprasath T (2019) Optimal power flow using moth swarm algorithm with gravitational search algorithm considering wind power. Future Gener Comput Syst 98:708–715. https://doi.org/10.1016/j.future.2018.12.046
19. Hu H, Cui X, Bai Y (2017) Two kinds of classifications based on improved gravitational search algorithm and particle swarm optimization algorithm. Adv Math Phys 2017:1–8. https://doi.org/10.1155/2017/2131862
20. Zhao F, Xue F, Zhang Y, Ma W, Zhang C, Song H (2019) A discrete gravitational search algorithm for the blocking flow shop problem with total flow time minimization. Appl Intell 49(9):3362–3382. https://doi.org/10.1007/s10489-019-01457-w
21. Wijaya ABM, Maedjaja F (2019) Adapted gravitational search algorithm using multiple populations to solve exam timetable scheduling problems. In: IEEE 2019 international congress on applied information technology (AIT), Yogyakarta, Indonesia, IEEE, pp 1–6. https://doi.org/10.1109/ait49014.2019.9144908
22. Ghorbani MA, Deo RC, Karimi V, Kashani MH, Ghorbani S (2019) Design and implementation of a hybrid MLP-GSA model with multi-layer perceptron-gravitational search algorithm for monthly lake water level forecasting. Stoch Env Res Risk Assess 33(1):125–147. https://doi.org/10.1007/s00477-018-1630-1
23. Xu BC, Zhang YY (2014) An improved gravitational search algorithm for dynamic neural network identification. Int J Autom Comput 11(4):434–440. https://doi.org/10.1007/s11633-014-0810-9
24. Shehadeh HA, Idna Idris MY, Ahmedy I, Ramli R, Mohamed Noor N (2018) The multi-objective optimization algorithm based

on sperm fertilization procedure (MOSFP) method for solving wireless sensor networks optimization problems in smart grid applications. Energies 11(1):97. https://doi.org/10.3390/en11010097
25. Mehic A (2020) The Electoral consequences of nuclear fallout: evidence from chernobyl. Department of Economics, School of Economics and Management, Lund University. https://ideas.repec.org/p/hhs/lunewp/2020_023.html
26. Ray K, Stick M (2015) Chapter 32—radiation and health effects. In: Gupta RC (ed) Handbook of toxicology of chemical warfare agents, vol 2. Academic Press, New York, pp 431–446. https://doi.org/10.1016/B978-0-12-800159-2.00032-4
27. L'Annunziata MF (2020) Handbook of radioactivity analysis, 4th edn. Academic Press, New York. https://doi.org/10.1016/c2016-0-04811-8
28. Almayahi B (2019) Use of gamma radiation techniques in peaceful applications. BoD Books Demand 1:1–260. https://doi.org/10.5772/intechopen.78481
29. McIntire M, Luczaj J (2019) Chernobyl's lesser known design flaw: the chernobyl liquidator medal—an educational essay. J Multidiscip Sci J MDPI 2(3):340–351. https://doi.org/10.3390/j2030023
30. Santos PP, Sillero N, Boratyński Z, Teodoro AC (2019) Landscape changes at chernobyl. In: Proceedings of remote sensing for agriculture, ecosystems, and hydrology XXI; 111491X, event: SPIE remote sensing, 2019, Strasbourg, France, pp 1–18. https://doi.org/10.1117/12.2532564
31. United States Department of Energy Office of Unclear Energy, Science and Technology, Washington, DC (2000) 64 FR 53669—programmatic environmental impact statement for accomplishing expanded civilian nuclear energy research and development and isotope production missions in the united states including the role of the fast flux test facility (DOE/EIS-0310), Publisher: Office of the Federal Register, National Archives and Records Administration,2:1−8https://www.federalregister.gov/documents/1999/09/15/99-24086/programmatic-environmental-impact-statement-for-accomplishing-expanded-civilian-nuclear-energy
32. Glikson AY (2017) The Plutocene: blueprints for a post-anthropocene greenhouse earth. Springer, Cham, pp 1–154. https://doi.org/10.1007/978-3-319-57237-6
33. Strath SJ, Swartz AM, Parker SJ, Miller NE, Grimm EK, Cashin SE (2011) A pilot randomized controlled trial evaluating motivationally matched pedometer feedback to increase physical activity behavior in older adults. J Phys Act Health 8(s2):S267–S274. https://doi.org/10.1123/jpah.8.s2.s267
34. Hamdan M, Yassein MB, Shehadeh HA (2015) Multi-objective optimization modeling of interference in home health care sensors. In: IEEE 11th international conference on innovations in information technology (IIT), Dubai, UAE, IEEE, pp 219–224. https://doi.org/10.1109/innovations.2015.7381543
35. Hamdan M, Bani-Yaseen M, Shehadeh HA (2018) Multi-objective optimization modeling for the impacts of 2.4-GHz ISM band interference on IEEE 802.15. 4 health sensors. In: Ismail L, Zhang L (eds) Information innovation technology in smart cities. Springer, Singapore, pp 317–330. https://doi.org/10.1007/978-981-10-1741-4_21
36. Shehadeh HA, Idris MYI, Ahmedy I, Hassen HR (2020) Optimal placement of near ground VHF/UHF radio communication network as a multi objective problem. Wirel Pers Commun 110:1169–1197. https://doi.org/10.1007/s11277-019-06780-6
37. Shehadeh HA, Jebril IH, Wang X, Chu SC, Idris MYI (2022) Optimal topology planning of electromagnetic waves communication network for underwater sensors using multi-objective optimization algorithms (MOOAs). Automatika 2022:1–12. https://doi.org/10.1080/00051144.2022.2123761