Chapter 3

DWCN-PSO with Genetic Algorithm

In the previous chapter, we have provided a basic overview of Evolutionary Computation (EC) methods and thoroughly reviewed existing approaches. This chapter introduces a novel Directed Weighted Complex Network Particle Swarm Optimization (DWCN-PSO) with Genetic Algorithm and also its application to image security.

3.1 Introduction

The optimization problem has been a fundamental research topic that fascinates a number of research communities from different domains because almost all real-world applications contain optimization problems within themselves. Optimization is the problem of finding the best possible solution while satisfying a set of constraints. Recently, applications of optimization have increased multifold, finding place in mechanics, engineering, image analysis, wireless sensor networks, IoT, ML, cryptography, and cybersecurity. Many of these problems are high dimensional, noisy, non-convex, multimodal, multi-objective, and dynamic in nature. To solve these hard optimization problems, several traditional and EC methods have been proposed. EC techniques prevail

due to their gradient-free approach, improved search ability, simplicity, and usability. Moreover, EC has gained enormous popularity in recent years in addressing problems belonging to NP-hard (NPH) and NP-complete (NPC) complexity classes and for solving optimization tasks whose real solution is unknown. These EC methods fall under population-based strategies that initialize a population of candidate solutions, and the solutions are evolved through interactions between the particles over the iterations. Recent works demonstrate that research groups are constantly striving to develop an effective optimizer either through a hybrid approach or by presenting a new mathematical model based on natural phenomena. In addition, several applications consist of more than one objective function, which are conflicting in nature, such as a low number of attributes versus higher accuracy for classification problems and low cost of the vehicle but more comfortable features, etc. EC is more suitable for such types of problems. One of these well-known methods of our interest is PSO, which works with a population called a swarm. The movement of particles is guided by a set of rules that use local and global information.

The PSO algorithm was first proposed by Kennedy and Eberhart [191], and till now, multiple variants have been available. In PSO, after several iterations, the best solution is found in the problem space by the interaction and exchange of information between candidate solutions, here termed as particles. It is superior to most of the current algorithms because of its simplicity and quicker convergence with a higher probability. But it also has its shortcomings, as once it reaches the optimum local, it converges to the optimum local and thus loses the diversity of the particles. This weakness proves to be very detrimental in multi-modal problems where there are more than one global optima and several local optima. It is evident that in order to find the optimal solution, there must be a good balance between local and global searches throughout the search process, which serves as a control parameter. In addition, the other two control parameters, inertia weight and acceleration coefficients of PSO, have been extensively

tested for better accuracy, and rapid convergence in a number of studies [192, 193].

Recently, several studies have been carried out to overcome these shortcomings, introducing new variants of PSO, such as time-varying acceleration coefficient PSO (PSO-TVAC) [193], and adaptive weighting PSO (AWPSO) [194]. In addition to these works, various topological structures of the particles were added to mitigate premature convergence, and this improved the accuracy to some extent. Still, different new problems were raised due to different topologies [195, 196, 197]. Furthermore, the evolutionary mechanism of the directed dynamic network is proposed, and the characteristics of the scale-free network model, as well as the small world network, have been utilized to design a new adaptive PSO. In this, the particles are connected by a directed weighted complex topology (DWCN-PSO), and the connections between the particles continue to increase after each iteration [198]. This algorithm aims to preserve the diversity of the population and then converge them to a unique optimum by slowly increasing the local neighborhood. It also had its limitations that it became slow due to the repeated formation of the network.

It is also worth mentioning that PSO and its variants are applied in several applications such as parameter optimization, feature selection, clustering, information security, image thresholding, and much more [199, 200]. Recently, data security and its integrity have been the primary concern of almost all fields, such as finance, medical, and the smart digital world. Since there are widespread development and advancement in IoTs [201], sensors and quick connectivity over the digital network provide us with comfort and facilities, particularly in the medical field, for patient data and report transmission. Therefore, in order to promote high-quality healthcare at an affordable cost, IoT-assisted mobile cloud-based e-health services are making huge strides through the use of advanced technologies such as Medi-cloud, Big Data in healthcare, and IoT in healthcare [202]. Further, it is easy to make use of expert suggestions for treatment around the globe these days, but it also poses a threat, like data theft, when we share

patient data and reports over the internet.

Since the main target of cyberattacks is the country's critical national infrastructure, such as banks, hospitals, train systems, electric power grids, etc., which use and rely on the SCADA and industrial control systems to manage their production [203]. Moreover, the risks of intentional attacks such as unauthorized access, manipulation of health records, and alteration of medical data theft are growing widely, rendering data security and privacy one of the challenging areas in smart healthcare that needs to be addressed these days. Hence, researchers addressed this problem and provided various techniques, such as watermarking, encryption, compression, and steganography [204, 133]. In another work, a reversible interpolation-dependent watermarking technique based on GA and PSO is developed and applied to medical as well as standard data [137]. Similarly, Grasshopper optimization with GA is used to create an optimal key that has been applied to both data sanitization, and restoration processes [138]. After analyzing its convergence and comparing it with other methods for medical data, the authors found this to be more effective than others.

Thus, in this chapter, firstly, we addressed the issues of local optima stagnation, slow convergence, and population diversity of DWCN-PSO by incorporating the operators of GA and analyzed its efficacy on standard benchmark functions. Thereafter, it has been modified to deal with multi-objective optimization problems. The proposal has been verified for optimal key generation for medical image security, where for n bits key, we have 2^n possibilities.

3.2 Motivation and Significant Contributions

The major issues with PSO are local optima entrapment and slow convergence. Similarly, DWCN-PSO also suffers from slow convergence due to the repeated formation of a network of particles. However, mutation and crossover operators of GA may solve the problem of local entrapment due to their randomness. Besides all these motiva-

tions, the NFL theorem [205] also states that there is no universal optimizer that gives the best results on all sets of problems. So, there is always a scope to develop new efficient optimizers to solve a challenging problem. By considering these challenges and possibilities, this research work presents an improved directed weighted complex network particle swarm optimization utilizing GA (GDWCN-PSO), which emphasizes faster convergence of the dynamic network without losing the diversity of the population. Since most of the challenging real-world problems have more than one conflicting objective function that needs to be optimized simultaneously; therefore, this work also presents a new multi-objective version of GDWCN-PSO. Besides this, we also addressed one of the challenging applications of medical data encryption and security by using GDWCN-PSO.

The main contributions of this chapter can be summarized as follows:

1. This work presents a hybrid model based on the particle's topological structure. It is biologically inspired by the mutation and crossover of genes and uses them as operators for particle seeking transformation in DWCN-PSO for a better solution with a fast convergence rate.

2. Basic arithmetic recombination is used as the crossover operator.

3. Extensive tests are performed to validate it against standard benchmarks of unimodal functions, multi-modal functions, and fixed dimension multi-modal functions.

4. For comparative analysis, PSO, PSO-GA, and DWCN-PSO are considered. The simulation experiments show that GDWCN-PSO outperforms PSO and DWCN-PSO in terms of accuracy and speed.

5. For wider acceptability and applicability, we also proposed a novel multi-objective version of GDWCN-PSO (MGDWCN-PSO). It is also validated against standard

MOO test suits and evaluated on performance metrics, Inverted Generational Distance (IGD), and Hypervolume (HV).

6. Finally, we have applied GDWCN-PSO to medical image encryption by selecting an optimal key, and results are evaluated using Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR) as well, as Structural Similarity Index (SSIM) evaluation metrics.

3.3 Theoretical Background

3.3.1 Particle Swarm Optimization

PSO is a robust stochastic global optimization method based on the social behavior of animals. PSO is initialized with an initial population of candidate solutions in n-dimensional space with position vector $x_i = (x_{i1}, x_{i2}, ..., x_{in})$ also known as 'particles' for $i = 1, 2, ...N$ where N is the number of particles initialized. The particles search the n-dimensional space following specific trajectories with a velocity $v_i = (v_{i1}, v_{i2}, ..., v_{in})$. Each particle stores the position in the n-dimensional space where it had the optimum value (P_{best}) of the optimizing function, and the best position overall (G_{best}) in the neighborhood is also stored [206]. These two best values influence the trajectory towards themselves by the following equations. The P_{best} position vector is given by the vector $pb_i = (pb_{i1}, pb_{i2}, ..., pb_{in})$. The G_{best}, global position is given by the vector $pg_i = (pg_{i1}, pg_{i2}, ..., pg_{in})$ and velocity and position of particle are updated as:

$$v_i \leftarrow wv_i + c_1 r_1 (pb_i - x_i) + c_2 r_2 (pg_i - x_i) \tag{3.1}$$

$$x_i \leftarrow x_i + v_i \tag{3.2}$$

These updates happen after every iteration. Here w is the weight inertia, which can also be described as the contribution of the previous velocity in determining the new

velocity. Here r_1 and r_2 are randomly generated numbers between $[0, 1]$, c_1 and c_2 are acceleration coefficients also lying between $[0, 2]$.

The neighborhood of the particle is a fascinating issue to be considered. There can be many topologies arising out for the neighborhood of the particles. In the original PSO, the neighborhood is composed of all the particles; thus, the best in the neighborhood is the global best position (G_{best}) of the optimizing function. The information exchange takes place throughout the swarm and gradually iteration by iteration; the diversity of the particles is lost as the swarm converges to a position in the n-dimensional space, which may or may not be the optimal solution. The l_{best} considers the neighborhood to be a set of nodes connected by either topology or dynamic scale-free networks. One such network is the issue under introspection.

3.3.1.1 Limitations of PSO

The primary limitation of PSO is that if a particle gets stuck in local optima, then all the other particles also converge to that local minima through information exchange and thus giving faulty solutions. Thus, there arises a need for retaining the particle diversity before extending the network.

3.3.2 Self-adaptive PSO based on Directed Weighted Complex Networks

First, the neighbor of the swarm of particles is defined by the complex network model given below. Then the interaction between the particles, which creates the dynamic network and the exchange of information, is explained.

3.3.2.1 Directed Weighted Complex Network of the Swarm Particles

Consider a graph $G = \{E, N, W\}$ where E is the set of edges, N is the set of nodes, and W is the weight of the edges between the nodes. In this case, N is the set of the particles initialized. Consider $A = (a_{ij})_{n \times n}$, which is the adjacency matrix denoting whether or not there is an edge between two nodes, node i and node j.

$$Here\, A = \begin{cases} 1, & if\ D(x_i, x_j) \leq R, V(f(x_i), f(x_j)) > 0 \\ 1, & if\ D(x_i, x_j) > R, V(f(x_i), f(x_j)) > 0 \\ 0, & otherwise \end{cases} \tag{3.3}$$

If the distance between two particles is greater than R, then connect the particles at probability p, $0<p<1$. where $D(x_i, x_j)$ denotes the Euclidean distance between the node i and j with $i \neq j$, R is the neighbor threshold distance and $V(f(x_i), f(x_j))$ denotes the difference between the objective functions which need to be optimized [207] [198].

3.3.2.2 Adaption and Learning of Particles

1. **Threshold radius (R) :** Threshold radius is one of the most crucial factors in the information exchange between the particles in the swarm. A particle is connected to another particle by a directed edge in the neighborhood if the distance between the two particles is less than the threshold radius and the difference between the objective function value of the two particles is positive. That is, particle A, which is being connected by an edge from B particle, should have the lesser value of an objective function.

2. **Probabilistic factor (p) :** Here every particle is connected to each other as explained above (lesser than radius R). There is also another criterion for the connection of two particles, which is the probabilistic factor (p). Here a particle A is connected to another particle B, which doesn't lie within the range of the threshold radius with a probability p. The value of the probabilistic factor (p) increases from 0 to 1 as the iterations keep on increasing. The growth can be uniform as well as nonuniform according to the problem to be investigated.

   This increasing probabilistic factor can be seen as a precedent of the divide and conquer principle. First, the swarms of particles are divided into clusters of radius R. Information exchange starts to take place between the particles within the

small cluster, and slowly the cluster becomes equivalent to the local optima of that cluster because it influences most of the particles in that cluster. Then as the probabilistic factor keeps on increasing, the size of clusters keeps on increasing, and effectively now, the exchange of information is taking place between the optima of a larger cluster. In this way, the swarm doesn't converge to any local optima, and the diversity of particles is not lost. In the case of the multi-modal objective function, if we want multiple solutions, we can stop the growth of the probabilistic factor to any value between $[0, 1]$ to get the best particles in the local clusters itself.

3.3.2.3 Limitations of DWCN-PSO

The DWCN-PSO takes a huge number of computations to build the network after every iteration. The number of iterations can be reduced to improve the speed, but we are not sure we will reach the optimal value if the iterations are reduced. The number of particles can be reduced, but this won't ensure whether the complete search space has been searched or not. Thus, we present our novel improved DWCN-PSO to address this shortcoming and to maintain diversity.

3.3.3 PSO with GA

Nowadays, a large number of algorithms are biologically inspired by the concepts of genetics, like concepts of survival of the fittest. Here the offsprings come from the selection of the fittest parents, crossing them and bringing in mutations to increase the diversity of the solutions. These sets of algorithms together are known as the GA. This concept of selection, crossover, and mutation can be applied to iterations of PSO to optimize its convergence rate. The speed-up factor of this paradigm of algorithms comes from the crossing over of the fittest population to create offsprings that are more superior to their parents. The weaker populations are then replaced by these newly generated offspring.

3.4 Proposed Methodology

This section provides a detailed description of our proposed algorithms. Since PSO suffers from slow convergence and local optima stagnation, different PSO variants are proposed, which we have mentioned in Section 3.1. DWCN-PSO is one of these variants, which is briefly presented in subsection 3.3.2 with its limitations. In this work, our major concern is to develop an optimization technique that is suitable for addressing real-world applications, and these applications mostly have multiple objectives. For this, we have considered DWCN-PSO, but due to the aforementioned limitations before proposing a multi-objective version, we have presented the improved version of DWCN-PSO and extended this improved version to design a multi-objective variant.

3.4.1 GDWCN-PSO

The limitation of DWCN-PSO is that it takes more time when the number of nodes of the network is increased. The application of GA to PSO showed improvement in its convergence rate. Thus we can use GA effectively to enhance the convergence rate of DWCN-PSO and thus provide more optimal solutions in a fixed time. The fundamentals of parent selection, crossover, and mutation can be applied after every iteration in DWCN-PSO, but we should take care of the diversity of solutions, which is the key point of applying DWCN. This can be controlled by how frequently we apply crossover in the iterations.

Now the important question which is raised is how the crossover is performed. We have used basic methods of a crossover, like whole arithmetic recombination. This crossover procedure has been performed by selecting two parents from the top-performing population and replacing the particles from the bottom (which have less optimal values) with the crossed over species. This operation is performed after the specified number of iterations so that the diversity is preserved and may not be done after each iteration. Therefore, we have improved DWCN-PSO by applying the specific operators of GA.

Algorithm 3.1: GDWCN-PSO

Input: A population of particles consisting of particles with random initial velocities and random positions, radius R, generations, probability p

```
Compute the fitness value of each particle and get the global best & local best.
while iteration ≤ generations do
    Calculate global best (G_best).
    Connect every particle within the range R to form a local neighborhood.
    Connect particle outside Radius R with probability 'p'.
    for each particle do
        Update velocities and positions in accordance with the local
        neighborhood created above.
        Select the best parent using parent selection techniques.
        Perform crossover in the parents to create X offspring.
        Replace the worst performing X particles with the newly created
        particles.
        Increase the probability 'p'.
        Update iteration= iteration + 1
    end
end
```

The pseudo-code for this is given in Algorithm 3.1. This method is tested against a variety of standard classical test functions with different complexity levels. The basic steps are as follows:

1. **Initialization and parameter setting :**

   Initialize population $P(t)$ of population size N with n dimensions, position vector $\vec{x_i}$ and velocity vector $\vec{v_i}$ of n dimension, P_{best} and G_{best} for the local best position of each particle and global best position of particle respectively. Set parameter for max generation, learning parameter c_1, c_2, probability p of edge connection and inertia weight w_{min}, w_{max}. Once the fitness for each particle has been computed,

the particle's location with the best fitness (minimum value for the minimization problem) is assigned as G_{best}.

2. **Network neighborhood formation for particles:**

Initialize the network neighborhood of the particles by considering the characteristics of the scale-free network model and small network model using Euclidean distance $D(x_i, x_j)$ between two particles x_i and x_j, threshold radius (R) and probability factor p and compute adjacency matrix A using equation (3.3) as described in section 3.3.2 and weight matrix W for network edges by following these equations:

$$w[i][j] = \begin{cases} V(f(x_i), f(x_j)) & a_{ij} \neq 0 \; and \quad \forall a_{ij} \in A \\ 0 & others \end{cases} \tag{3.4}$$

Further edge weight of constructed particle's network is normalized as:

$$W^{normalized} = \frac{w[i][j]}{\sum(w[i][j])} \tag{3.5}$$

$$\forall\, i = \{1, 2, \cdots, N\} \; and \; \forall \; j = \{1, 2, \cdots, N\}$$

$$0 \leq \; w[i][j] \leq \; 1$$

3. **Adaptive learning of particles and modified position updates:**

As particles are connected with neighbors having $D(x_i, x_j)$ within the threshold value of R and also connected randomly with its virtual neighbors at a certain probability p. Virtual neighbors are the particles that lie outside the threshold radius. These virtual neighbors make particle to jump outside to the local optima when it staggers to the local optima. Thus, through this process, each particle learns from its own optimal as well as the neighbor's optimal position. The particle

can effectively jump out of its local position in a situation where it is not able to find the optima within the threshold search space. In order to introduce dynamic learning, velocity and positions are updated by modifying equation (3.1) as:

$$v_i \leftarrow wv_i + c_1 r_1(pb_i - x_i) + c_2 r_2(pl_i - x_i) \tag{3.6}$$

$$x_i \leftarrow x_i + v_i \tag{3.7}$$

Therefore, all the particles optimize according to the above equations if the indegree of network neighborhood obeys power-law distribution, but in the situation where the particle's velocity and the position cross the search bound, it auto-

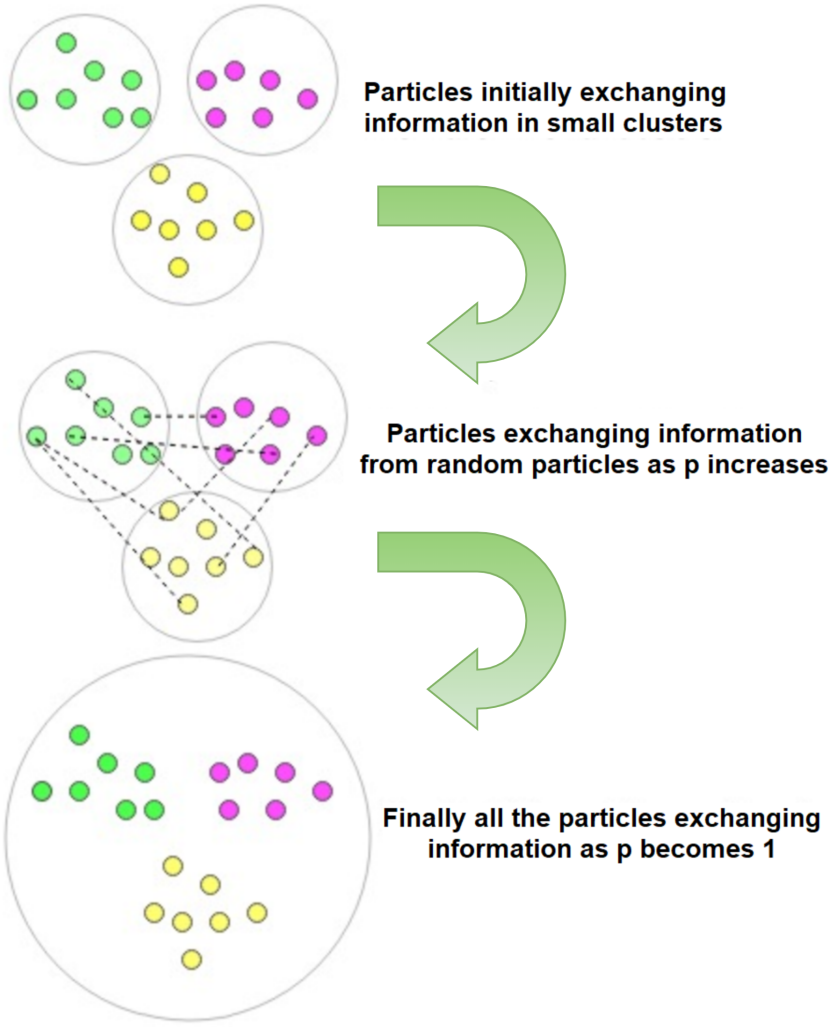


Figure 3.1: Flow diagram representing adaption and learning of particles.

matically selects the boundary values and computes A and W and updates them dynamically. The whole process is described in Figure 3.1.

4. **Environmental selection and reproduction:**

To overcome the limitations of DWCN-PSO in high-dimension problems, we hybridized the concept of GA. To achieve better convergence, we sorted the swarm particles according to their fitness function and selected particles with the best potential and performed crossover. Further, we replace the worst swarm particles with these new particles after specific iterations to maintain the diversity of solutions. The whole process is repeated until the stopping criteria are met.

3.4.2 Multi-objective GDWCN-PSO (MGDWCN-PSO)

The proposed MGDWCN-PSO is an extensive multi-objective version of GDWCN-PSO. Population initialization and neighborhood selection are similar to their single-objective counterparts. The whole process is described as follows:

1. **Swarm initialization and parameter setting:**

- Set swarm population size N with particle dimension n corresponds to a dimension of the problem.
- Set the upper and lower bound for the variable x_i. Initialize random positions with random velocities.
- Parameter setting: Set the value of threshold radius R, maximum number of evaluations as Maxgen, learning parameters $c1$, $c2$, inertia weights w_{min}, w_{max}, probability factor p and number of objective functions k.

2. **Network neighborhood formation of particles and adaptive learning:**

It is also constructed in the same way as explained in the single objective version by using the equation (3.3). Adaptive learning in the neighborhood of the particles from its local neighbor and virtual neighbor based on fitness computation

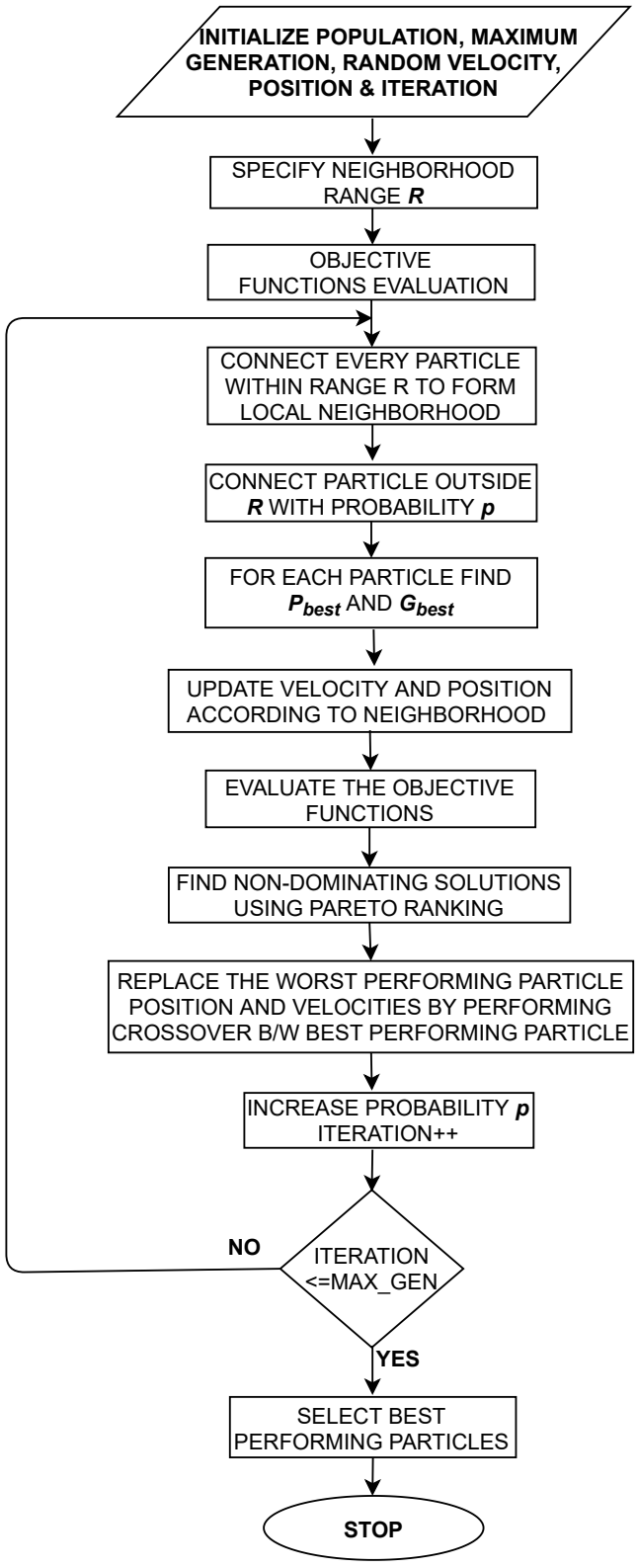


Figure 3.2: Flowchart of multi-objective GDWCN-PSO.

is performed. The velocity, position of the particles, their adjacency matrix, and the weights corresponding to the edges of the particles in the network with their parameters are updated according to their single objective counterparts described in equations (3.4), (3.5) and (3.6).

3. **Enviromental selections and reproduction:**
Objective functions $f_k(x_i)$ are computed by each particle, and the best fittest values are selected using Pareto-based ranking with crowding distance[208]. Further, two non-dominated solutions are selected as parents, and one point crossover is applied to generate offspring. Then we replaced the worst swarms in the population with these newly generated competitive swarms after a specific iteration and generated a new population. Finally, we updated the position and velocity of the particles and then followed step 2 and 3 until the termination criteria were met.

The flowchart of the whole algorithm is described in Figure 3.2.

3.5 Application of GDWCN-PSO for Optimal Key-based Image Encryption

Automated computer-assisted systems are the current need for healthcare systems, and it is highly beneficial for accurate, fast as well as efficient diagnosis of medical images. Nowadays, the Internet is used to share valuable information more easily, and due to the amount of medical data being exchanged over the internet and the number of digital records being generated as well as maintained and give rise to remote healthcare applications where patients and doctors can perform their duties from different geographic locations. Expert suggestions are taken by doctors as well as patients for correct diagnoses across the country. Moreover, these advanced facilities need online digital data and image transmission through IoT. So currently, the healthcare system is integrated with IoTs and giving rise to innovations of different sensors for healthcare.

Hence, it requires a new secure technique for medical image transmission to maintain the privacy, integrity, and authenticity of patients' sensitive data because small changes in data may lead to the wrong diagnosis, and it is perilous for the patient. Therefore, it becomes essential to provide solutions for the most challenging problem of data security in the ever-blooming IoTs and cloud computing of the modern era. Since the encryption techniques belong to time-honored cryptography are used extensively to provide robust protection of sensitive data like medical data and reports etc. Image encryption is a useful technique of image content protection, which is very useful for medical data security. Medical image encryption plays a key role in healthcare applications.

Image encryption is used for the secure sharing of the image over the network. Over the last decade, much attention has been paid to medical images, and a number of image encryption algorithms have been developed. Also, it is worth mentioning that due to some of the inherent characteristics of images, such as huge data capacity and high data consistency, image encryption is different from text; thus, it is difficult to use conventional encryption methods. Therefore, we have proposed a random optimal key-based encryption technique for medical images, as shown in Figure 3.3. To accomplish

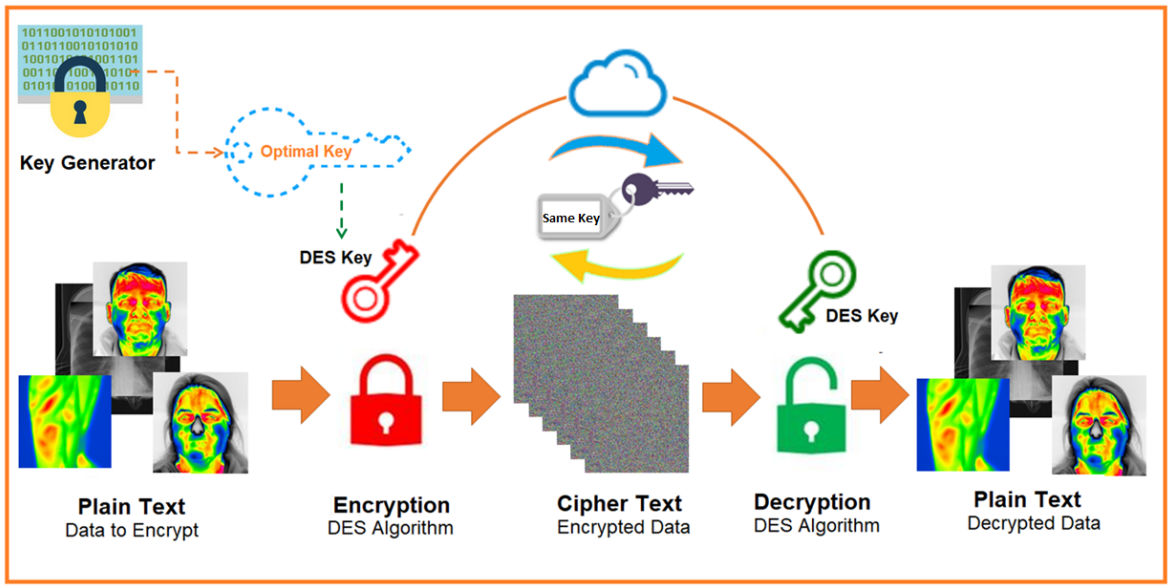


Figure 3.3: Proposed framework for medical data security.

Algorithm 3.2: GDWCN-PSO for Optimal Key Generation

Input: A population of particles with random initial position and random velocity, radius R, generations, probability p

Calculate the fitness value for each particle and get G_{best} & local best.
while *iteration* $\leq$ *generations* **do**
    Calculate G_{best}.
    Connect every particle within the range R to form a local neighborhood.
    Connect particle outside Radius R with probability $'p'$.
    for *i in range (num-of-particle)* **do**
        Update velocities and positions in accordance with the local neighborhood created above.
        Select the best parent using parent selection techniques.
        Perform crossover in the parents to create X offspring.
        Replace the worst performing X particles with the newly created particles.
        Increase the probability $'p'$.
        Iteration ++
    end
    def key-function (x)
    Initialize array a[x] = [0].
    for *i in range (len[x])* **do**
        if (x[i] $\geq$ 0.5)
        a[i] = 1
    end
    Perform **gaptest( )**
end
def gaptest (a)
Find the maximum distance between two different digits of a key and set it to *max_distance*.
return *max_distance*

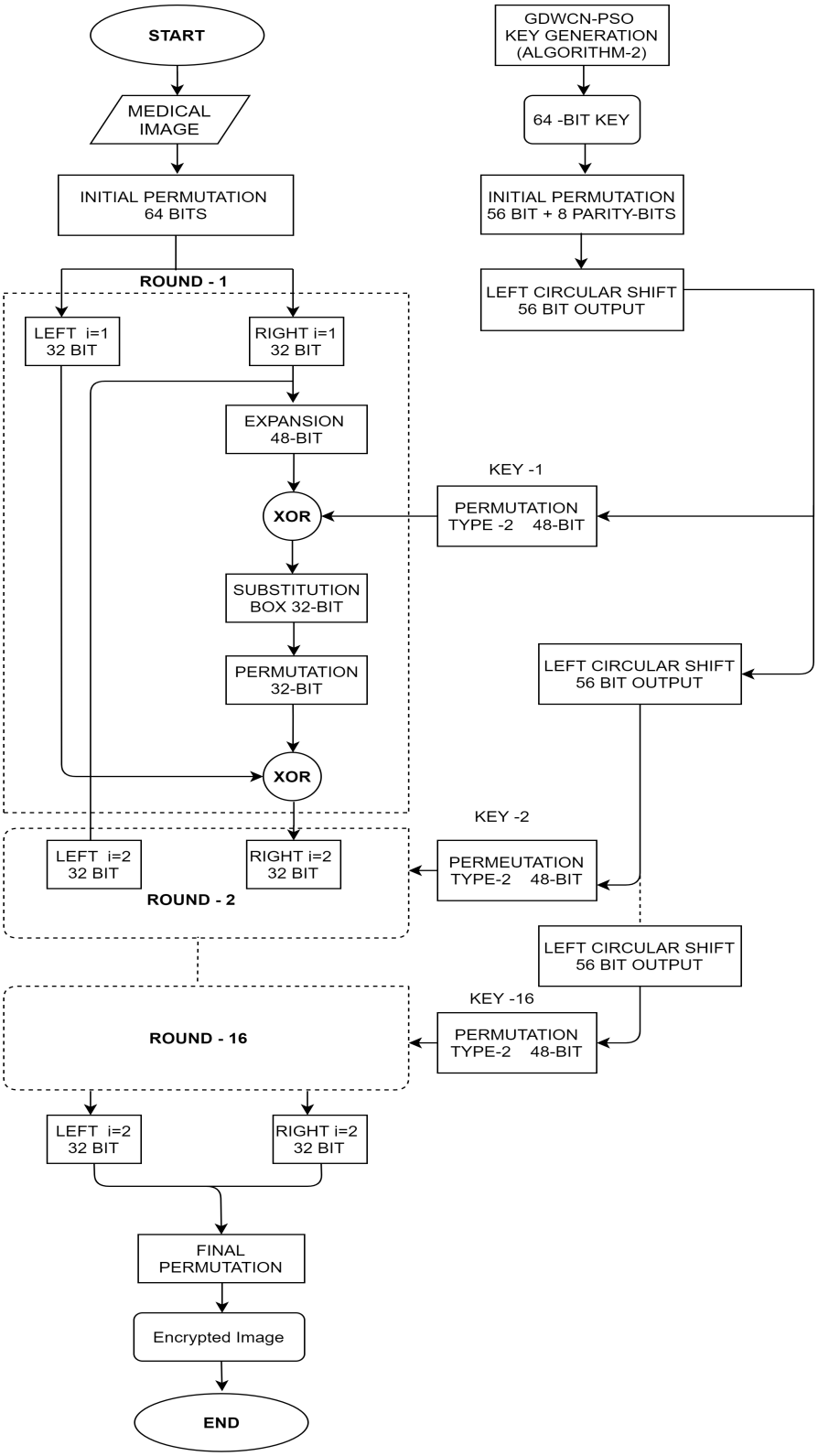


Figure 3.4: Flowchart of medical image encryption.

this goal, GDWCN-PSO is used for generating an optimal key to encrypt the image. For this, we considered the gap test [209] [210] as an objective function $F(k)$. This optimization problem is a minimization problem shown in equation (3.8).

$$F(k) = min(gaptest) \tag{3.8}$$

The key generation process is described in Algorithm 3.2. For encryption, we have selected a symmetric cryptography algorithm, Data Encryption Standard (DES) [140]. DES, based on a block cipher, is designed by utilizing the generated key from the optimizer. It uses the same key for encryption and decryption. For encryption, it uses 64-bit blocks of plaintext to generate a ciphertext of 64 bits. The complete procedure for this involves 16 steps referred to as the round key. Encryption involves two permutations (P-boxes) that are initial and final permutations, along with 16 Feistel rounds. The key size is 64 bits, of which 56 bits are used for encryption, and the rest are used as parity bits. A similar operation is performed during decryption to retrieve the original data at the receiver site. The complete process of medical image encryption is described in Figure 3.4.

3.6 Results and Discussions

In this section, we have presented an analysis on standard test functions belonging to single-objective optimization and multi-objective test suits for showing the effectiveness of the proposed GDWCN-PSO and MGDWCN-PSO, respectively. Later in this section, we provide a detailed analysis of the application of GDWCN-PSO in the area of medical data security in IoTs by using an optimal key-based symmetric cryptography algorithm. The simulation experiments are run on a computer with Intel Core i7-7500U @ 2.70GHz and 8 GB of memory.

3.6.1 Benchmark Functions

To investigate the performance of the proposed GDWCN-PSO method, we have used different classical benchmark functions with different configurations in terms of dimensions, search range, and modality [211]. We have considered 18 classical benchmarks among these functions, f_1 -f_{15} are the problem with five dimensions, and the remaining functions are of two dimensions. We have also considered unimodal benchmark functions presented in Table 3.1, multi-modal benchmarks in Table 3.2, and Fixed dimension multi-modal benchmarks detailed in Table 3.3. For performing experiments, we have taken N as 30, maxgen is 200, 0.4, and 0.9 as w_{min} and w_{max} respectively. Learning factors c_1 and c_2 are considered in [0,2], and p is taken in the interval [0,1]. The complete simulation is executed 30 times, and the mean fitness is compiled for each function and is reported in Table 3.4.

GDWCN-PSO is also compared with PSO, PSO-GA, and DWCN-PSO. As shown in Table 3.4, the values for GDWCN-PSO are much better than that of PSO and DWCN-PSO for a fixed set of iterations. For better visualization, we have considered two unimodal functions f_1 and f_5 as well as two multi-modal functions f_9 and f_{10}. The surface plot, convergence plot, and contour plots of these 4 functions are presented in Figure 3.5, Figure 3.6, Figure 3.7, and Figure 3.8, respectively. It is clearly visible that

Table 3.1: Unimodal-benchmark test functions.

Test Function	**Variable Bounds**	$\boldsymbol{f_{min}}$
$f_1(x) = \sum_{i=1}^{n} x_i^2$	$[-100, 100]$	0
$f_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	$[-10, 10]$	0
$f_3(x) = \sum_{i=1}^{n} (\sum_{j-1}^{i} x_j)^2$	$[-100, 100]$	0
$f_4(x) = max_{i=1}^{n} \{\lvert x_i \rvert\}$	$[-100, 100]$	0
$f_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$	$[-30, 30]$	0
$f_6(x) = \sum_{i=1}^{n} ([x_i + 0.5])^2$	$[-100, 100]$	0
$f_7(x) = \sum_{i=1}^{n} i x_i^4 + random[0, 1)$	$[-1.28, 1.28]$	0

Table 3.2: Multi-modal-benchmark test functions.

Test Function	**Variable Bounds**	f_{min}
$f_8(x) = \sum_{i=1}^{n} -x_i sin(\sqrt{\lvert x_i \rvert})$	$[-500, 500]$	-2094.914
$f_9(x) = \sum_{i=1}^{n} (10 + x_i^2 - 10cos(2\pi x_i))$	$[-5.12, 5.12]$	0
$f_{10}(x) = -20exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}) - exp(\frac{1}{n}\sum_{i=1}^{n} cos(2\pi x_i)) + 20 + e$	$[-32, 32]$	0
$f_{11}(x) = 1 + \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} cos(\frac{x_i}{\sqrt{i}})$	$[-600, 600]$	0
$f_{12}(x) = \frac{\pi}{n}(10sin^2(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10sin^2(\pi y_{i+1})] + (y_n - 1)^2) + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$ $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	$[-50, 50]$	0
$f_{13}(x) = 0.1(sin^2(3\pi x_1) + \sum_{i=1}^{n}(x_i - 1)^2[1 + sin^2(3\pi x + 1)] + (x_n - 1)^2[1 + sin^2(2\pi x_n)]) + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	$[-50, 50]$	0
$f_{14}(x) = -\sum_{i=1}^{n} sin(x_i) \times (sin(\frac{ix_i^2}{\pi}))^{2m}, m = 10$	$[0, \pi]$	-4.687
$f_{15}(x) = ((\sum_{i=1}^{n} sin^2(x_i)) + -exp(-\sum_{i=1}^{n} x_i^2)) \times exp(-\sum_{i=1}^{n} \sqrt{\lvert x_i \rvert})$	$[-10, 10]$	-1

Table 3.3: Fixed dimension multi-modal-benchmark test functions.

Test Function	**Variable Bounds**	f_{min}
$f_{16}(x) = 4x_1^2 + -2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	$x \in [-5, 5]$	-1.0316
$f_{17}(x_1, x_2) = (x_2 + \frac{5}{4\pi^2}x_i^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})cosx_1 + 10$	$x_1 \in [-5, 10], x_2 \in [0, 15]$	0.398
$f_{18}(x) = (1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)) \times (30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2))$	$[-2, 2]$	3

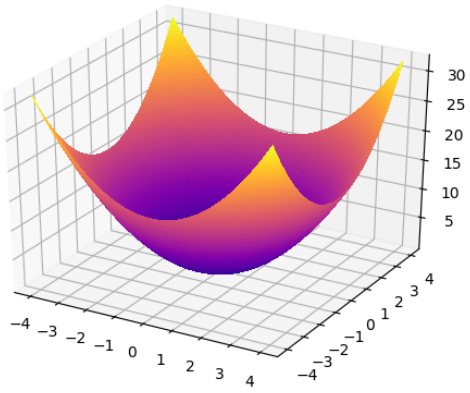
the proposed algorithm is able to converge to a minimum value in all four functions specified above. For function f_1, Figure 3.5 (a) shows the surface plot, and Figure 3.5 (b), demonstrates the convergence rate of the GDWCN-PSO algorithm as the iterations increase. In Figure 3.5 (c), the trajectory of the first particle in a single dimension is traced out across the iterations for the function f_1. This curve clearly shows that during the initial iterations, the particle is heavily influenced by factors like local neighborhood maximum, and thus the positions keep on changing, but once iterations increase and the local maxima's are found, then the positions stop varying much and after some time it becomes constant. In the relative convergence curve, our proposed novel algorithm outperforms PSO and DWCN-PSO as it converges to a lesser value and it converges in

Table 3.4: Mean values of PSO, DWCN-PSO, and GDWCN-PSO.

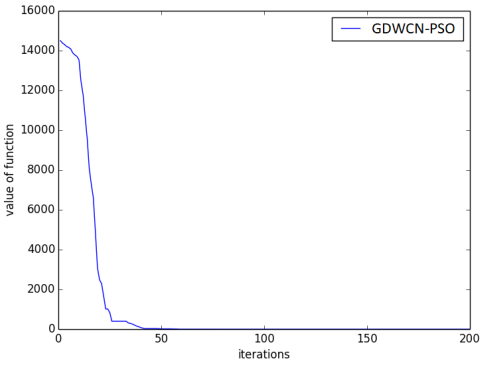
Function	PSO	PSO-GA	DWCN-PSO	GDWCN-PSO
f_1	7.5747E-09	1.7768E+02	1.0927E-03	**1.1013E-10**
f_2	4.7530E+00	5.2788E+00	3.4453E+00	**6.6816E-01**
f_3	9.8519E+02	1.9908E+03	7.2088E+02	**3.3333E+02**
f_4	1.4081E+01	2.2820E+01	7.1054E-06	**1.2607E-12**
f_5	**4.7387E+01**	1.3926E+02	1.0389E+02	1.0431E+02
f_6	1.8611E+00	2.9620E+02	5.6004E-02	**1.0487E-08**
f_7	2.4350E-02	3.8990E-02	3.3220E-02	**1.5596E-02**
f_8	-1.1074E+03	-1.0391E+03	-1.4338E+03	**-1.5159E+03**
f_9	4.1238E+01	3.8919E+01	3.3269E+01	**1.0844E+01**
f_{10}	1.9077E+01	1.9277E+01	1.4295E+01	**3.9767E+00**
f_{11}	6.2840E+01	6.9174E+01	6.1416E+00	**3.1269E-01**
f_{12}	6.2923E+00	1.3732E+01	-5.7290E+00	**-5.7185E+00**
f_{13}	2.7420E+01	6.1743E+01	5.6881E-02	**7.7134E-03**
f_{14}	-3.6107E+00	-3.3133E+00	-2.8166E+00	**-3.7158E+00**
f_{15}	7.1033E-04	**-2.5901E-16**	2.4163E-03	1.0963E-05
f_{16}	-8.4119E-01	-7.8678E-01	-1.0044E+00	**-1.0316E+00**
f_{17}	6.1350E-01	3.9789E-01	6.1350E-01	**3.9789E-01**
f_{18}	1.8821E+01	2.1000E+01	1.1450E+01	**3.2808E+00**

a lesser number of iterations.

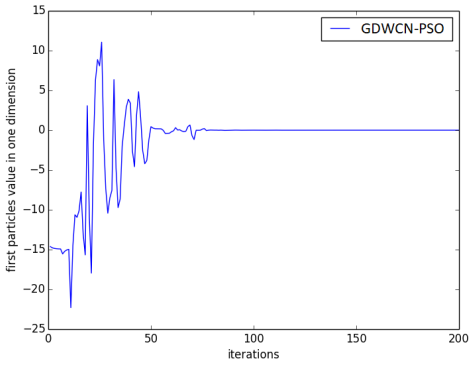
Furthermore, Figure 3.5 (d), shows the relative performance of PSO, DWCN-PSO, and GDWCN-PSO on function f_1 of dimension 5. This figure shows the superiority of GDWCN-PSO over PSO and DWCN-PSO. The reason for the superiority of GDWCN-PSO over DWCN-PSO lies in the point that the searching power being the same, but it converges to the minima faster than that of DWCN-PSO and superiority over PSO lies, in fact, because of superior searching capabilities of GDWCN-PSO. It is also noticed that during the initial iteration, PSO performs better than GDWCN-PSO



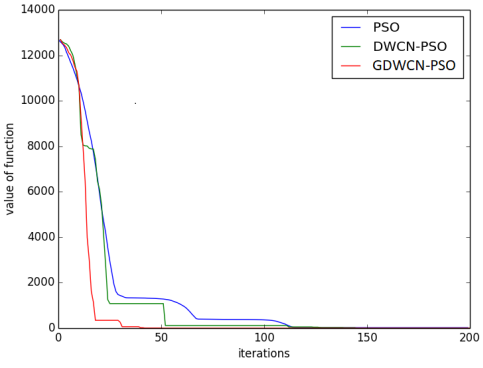
(a) Surface plot



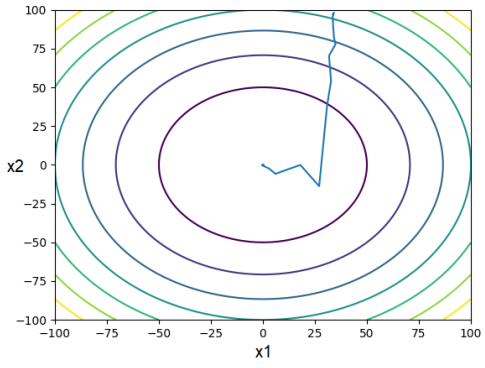
(b) Convergence curve



(c) Co-ordinate of a particle in one dimension

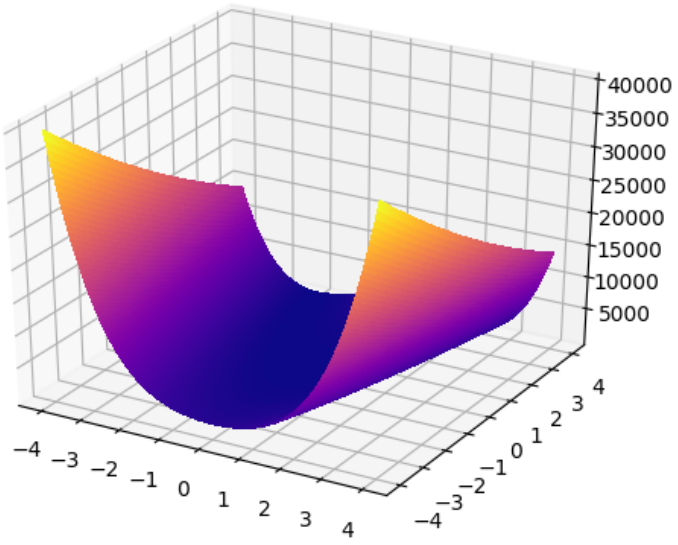


(d) Performance of GDWCN-PSO in comparison to other algorithms

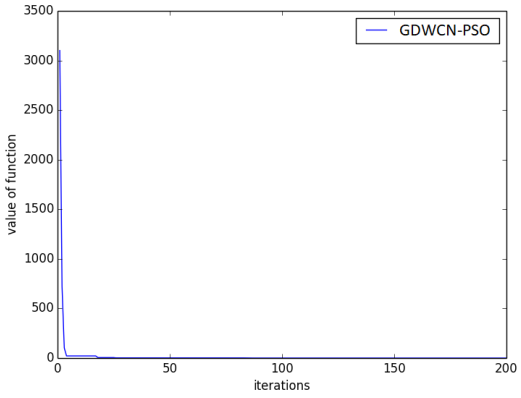


(e) Contour plot for the convergence

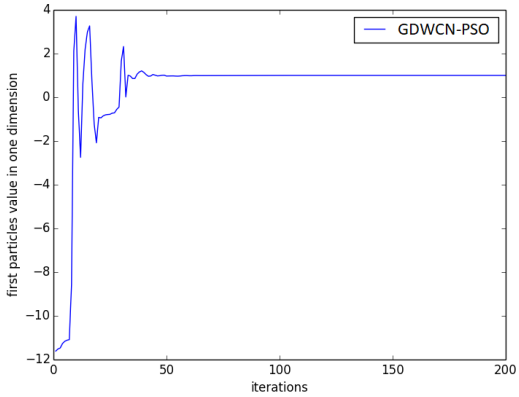
Figure 3.5: Results for $f_1(x)$.



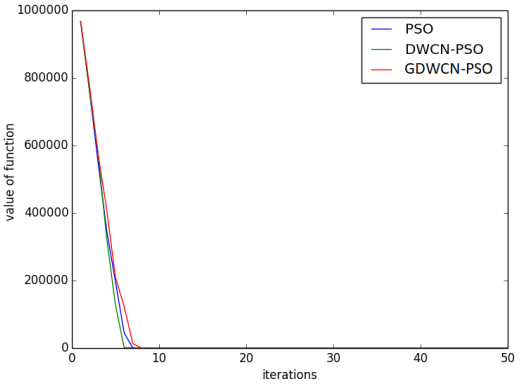
(a) Surface plot



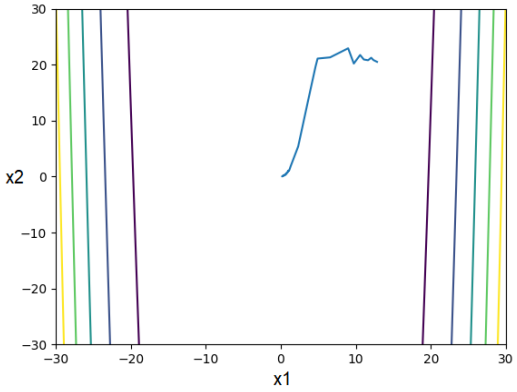
(b) Convergence curve



(c) Co-ordinate of a particle in one dimension

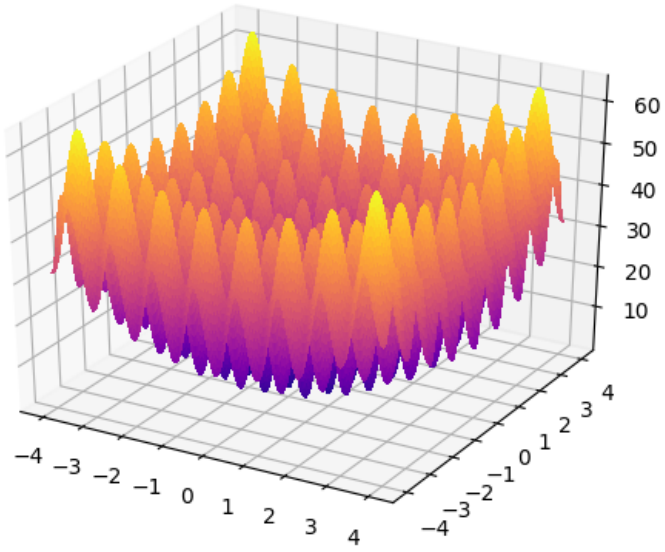


(d) Performance of GDWCN-PSO in comparison to other algorithms

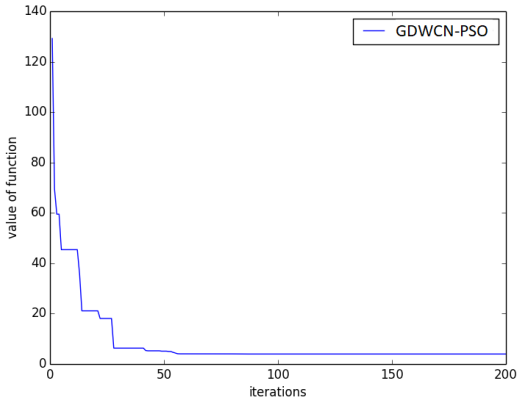


(e) Contour plot for the convergence

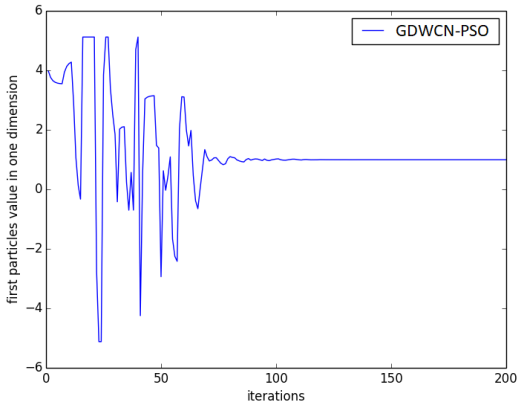
Figure 3.6: Results for $f_5(x)$.



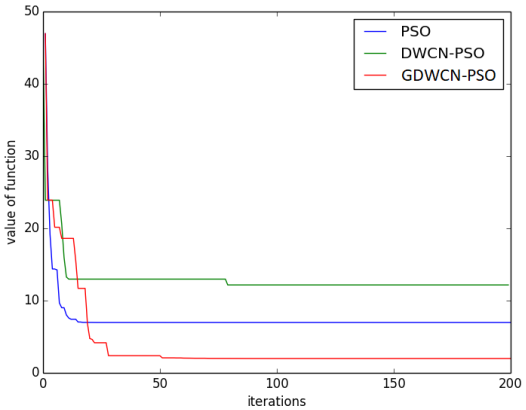
(a) Surface plot



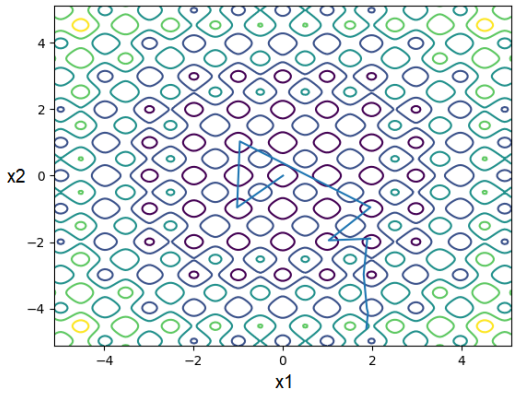
(b) Convergence curve



(c) Co-ordinate of a particle in one dimension

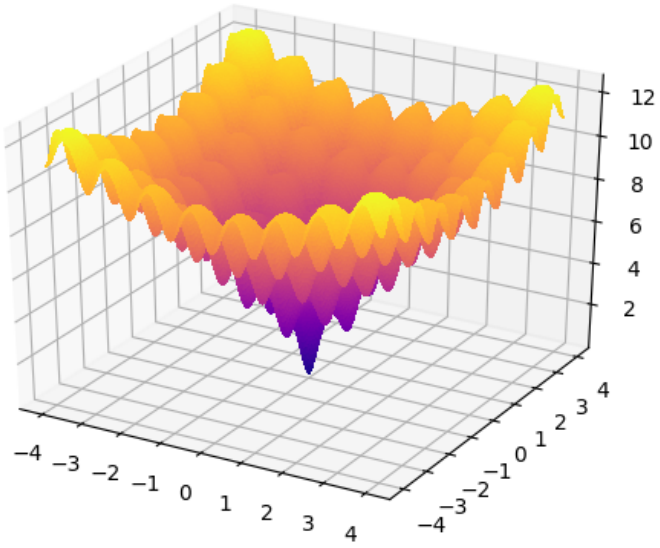


(d) Performance of GDWCN-PSO in comparison to other algorithms

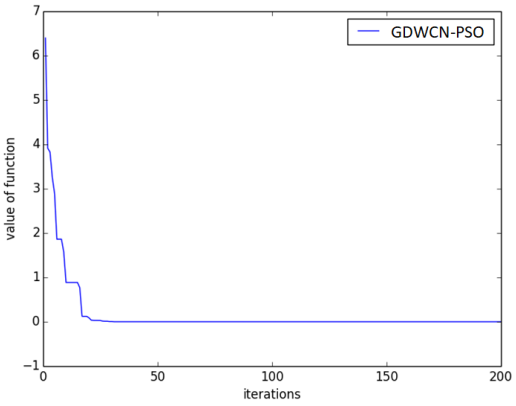


(e) Contour plot for the convergence

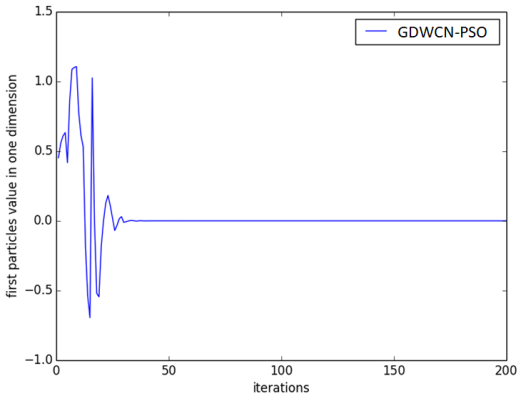
Figure 3.7: Results for $f_9(x)$.



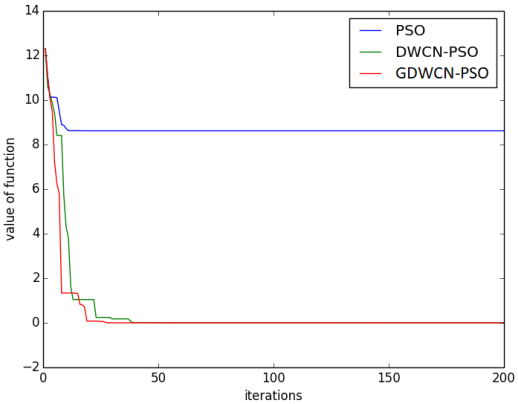
(a) Surface plot



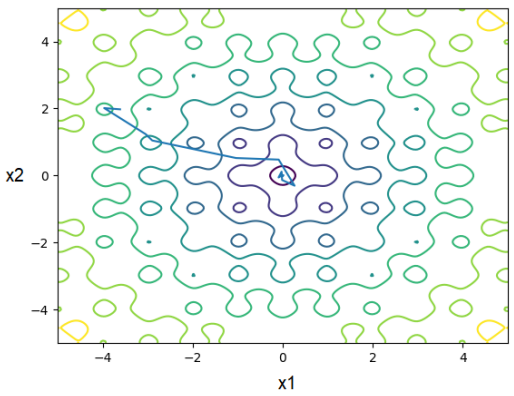
(b) Convergence curve



(c) Co-ordinate of a particle in one dimension



(d) Performance of GDWCN-PSO in comparison to other algorithms



(e) Contour plot for the convergence

Figure 3.8: Results for $f_{10}(x)$.

because initially, the networks are small, and the complete exchange of information is not possible. Figure 3.5 (e), presents contour plots of the function f_1, which shows that it converges to the optimum value of the function. Similar plots are also presented for f_5, f_9 & f_{10}, in Figure 3.6, Figure 3.7, and Figure 3.8 respectively. It has been

Table 3.5: Multi-objective benchmark test functions.

Problem Name	**Objective Functions**	**Variable Bounds**	**Remarks**
ZDT1	$f^1(x) = x_1$ $f^2(x) = h(x)\left(1 - \sqrt{\frac{x_1}{h(x)}}\right)$ $h(x) = 1 + \frac{9}{29}\sum_{i=2}^{30} x_i$	[0,1]	Convex
ZDT2	$f^1(x) = x_1$ $f^2(x) = h(x)\left[1 - \left(\frac{x_1}{h(x)}\right)^2\right]$ $h(x) = 1 + \frac{9}{29}\sum_{i=2}^{30} x_i$	[0,1]	Nonconvex
ZDT3	$f^1(x) = x_1$ $f^2(x) = h(x)\left(1 - \sqrt{\frac{x_1}{h(x)}} - \frac{x_1}{h(x)}sin(10\pi x_1)\right)$ $h(x) = 1 + \frac{9}{29}\sum_{i=2}^{30} x_i$	[0,1]	Convex disconnected
ZDT6	$f^1(x) = 1 - e^{(-4x_1)}sin^6(6\pi x_1)$ $f^2(x) = h(x)\left[1 - \left(\frac{f^1(x)}{h(x)}\right)^2\right]$ $h(x) = 1 + 9\left[\frac{(\sum_{i=2}^{n} x_i)}{(n-1)}\right]^{0.25}$	[0,1]	Nonconvex Not uniformly spaced
Poloni	$f^1(x,y) = [1 + (A_1 - B_1(x,y))^2 + (A_2 - B_2(x,y))^2]$ $f^2(x,y) = (x+3)^2 + (y+1)^2$ *where* $A_1 = 0.5sin1 - 2cos1 + sin2 - 1.5cos2$ $A_2 = 1.5sin1 - cos1 + 2sin2 - 0.5cos2$ $B_1(x,y) = 0.5sin(x) - 2cos(x) + sin(y) - 1.5cos(y)$ $B_2(x,y) = 1.5sin(x) - cos(x) + 2sin(y) - 0.5cos(y)$	$[-\pi, \pi]$	Disconnected
Fonseca-Fleming	$f^1(x) = 1 - e^{-\sum_{i=1}^{n}\left(x_i - \frac{1}{\sqrt{n}}\right)^2}$ $f^2(x) = 1 - e^{-\sum_{i=1}^{n}\left(x_i + \frac{1}{\sqrt{n}}\right)^2}$	[-4,4]	Nonconvex
Schaffer N.1	$f^1(x) = x^2$ $f^2(x) = (x-2)^2$	$[-10, 10^5]$	Convex
Schaffer N.2	$f^1(x) = \begin{cases} -x, & if\, x \le 1 \\ x-2, & if\, 1 < x \le 3 \\ 4-x, & if\, 3 < x \le 4 \\ x-4, & if\, x > 4 \end{cases}$ $f^2(x) = (x-5)^2$	[-5,10]	Disconnected

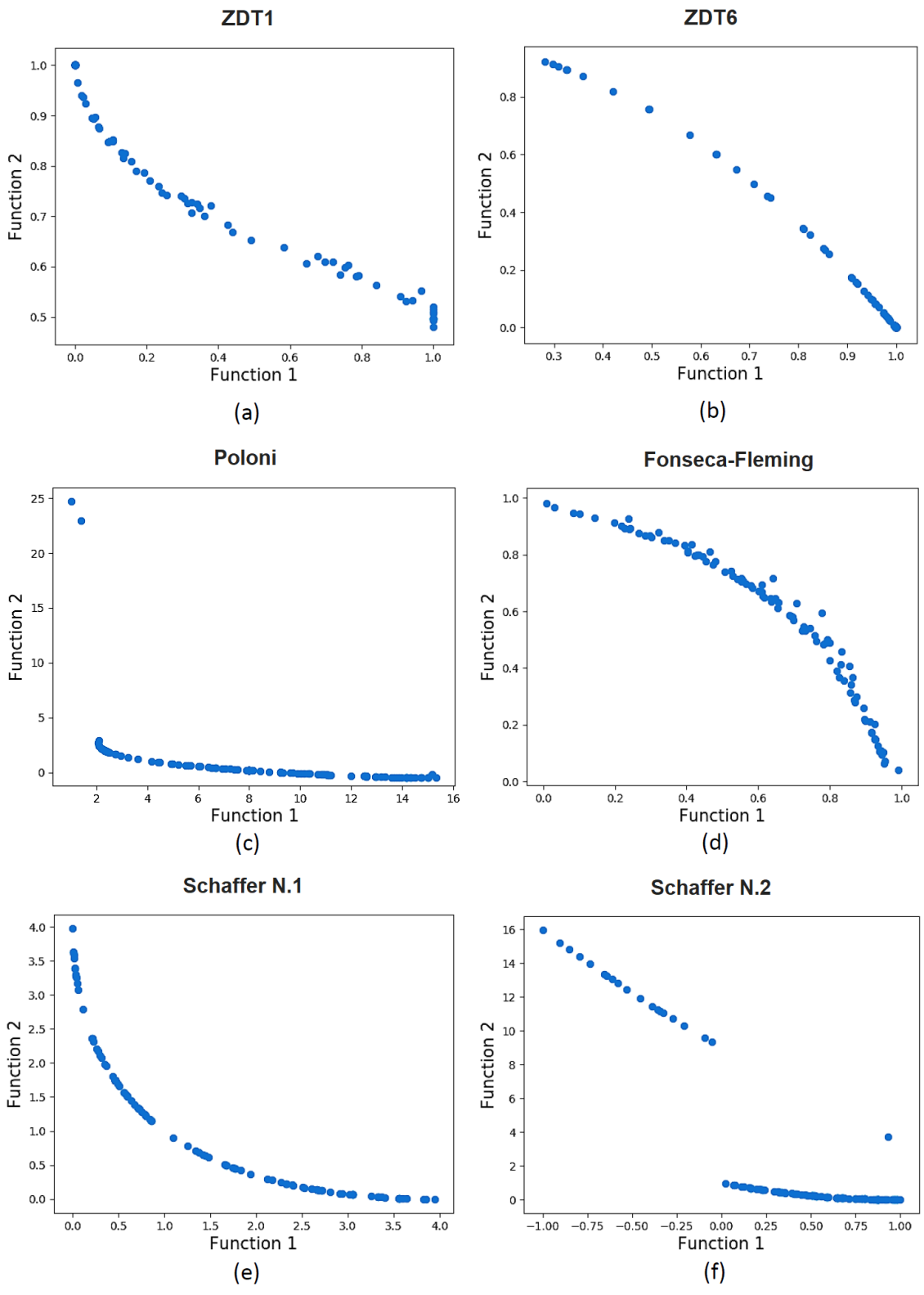


Figure 3.9: Pareto fronts for multi-objective benchmarks.

Table 3.6: Performance metrics for multi-objective problems

Test Functions	HV Mean	IGD Mean
ZDT1	117.052	2.942
ZDT2	106.586	4.165
ZDT3	71.523	2.893
ZDT6	107.625	7.617
Poloni	88.128	NA
Fonseca-Fleming	120.098	NA
Schaffer N.1	118.198	NA
Schaffer N.2	120.741	NA

observed that in almost all benchmarks, GDWCN-PSO outperforms as compared to PSO, PSO-GA, and DWCN-PSO. In order to verify the significant differences between the results of the proposed GDWCN-PSO and the other optimizer, the Wilcoxon rank-sum test with 5% degree of the significance level is meticulously performed here [212]. Differences in results are observed to be statistically meaningful in most cases.

Since we have also proposed a multi-objective version of GDWCN-PSO, therefore the effectiveness of MGDWCN-PSO is tested on standard MOP benchmarks Poloni, Fonseca-Fleming, Schaffer, ZDT test suits [213]. These test suits for the MOO problem are presented in Table 3.5. Further, the Pareto fronts obtained from MGDWCN-PSO on these test suits are shown in Figure 3.9. It is also validated on two performance metrics, IGD and HV, as shown in Table 3.6. IGD and HV are the unary performance metrics used in [214] for measuring the convergence and diversity of the obtained nondominated solutions of MOP. These metrics are widely used for evaluating the performance of NIAs for MOO. For computing IGD value, information related to the true Pareto front is necessary, while HV can be computed without any prior information of the true Pareto front. High HV values and low IGD values are preferable for the better performance of any optimizer. For details, readers are referred to [214].

3.6.2 Performance Evaluation of Medical Image Encryption

In order to validate our improved GDWCN-PSO algorithm for medical image encryption, we created a small dataset of ten medical images, of which five are infrared thermal images captured in IIT(BHU), Varanasi, and five are radiology images taken from Sir Sunderlal Hospital, Varanasi. Further, we have taken three images of each infrared thermal image and radiology images from those images as test images. Figure 3.10 presents these test images and their detailed information. We applied our proposed encryption scheme to these images and found that all encrypted images are properly decrypted with no visual loss.

Performance of the whole process is evaluated on the basis of PSNR, MSE, and SSIM [215]. These metrics are calculated as follows:

$$PSNR = 10 \times \log_{10}(\frac{Q^2}{MSE}) \tag{3.9}$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(A_i - B_i)^2 \tag{3.10}$$

$$\text{SSIM}(\mathbf{A}, \mathbf{B}) = \frac{(2\mu_A\mu_B + C_1)(2\sigma_{AB} + C_2)}{(\mu_A^2 + \mu_B^2 + C_1)\,(\sigma_A^2 + \sigma_B^2 + C_2)} \tag{3.11}$$

where $Q \in [0, 255]$, N is the pixel count, A_i and B_i are the i^{th} pixel in the original and processed image. C_1 and C_2 are the regularize parameters.

Result analysis after encryption and decryption using optimal key-based symmetric cryptography algorithm DES is presented in Figure 3.10. The overall performance of GDWCN-PSO is evaluated on different medical images and also compared with other existing methods. We have calculated MSE, PSNR, and SSIM between the original and encrypted as well as the original and decrypted image. Results for evaluation metrics between original and encrypted are presented in Table 3.7. We see that with optimal key

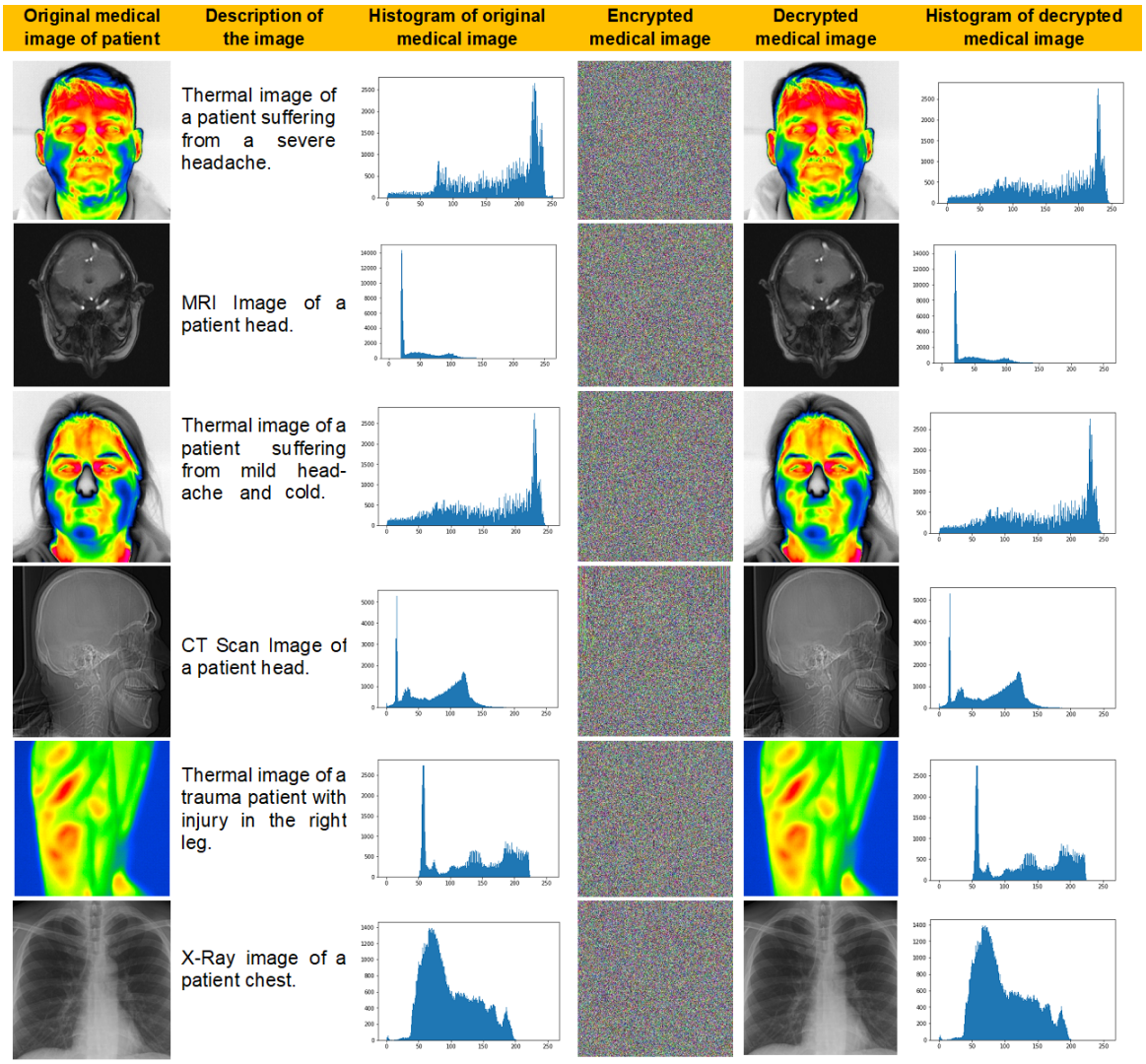
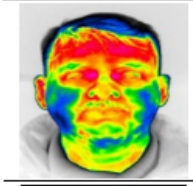
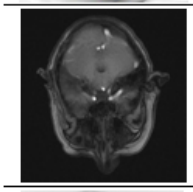
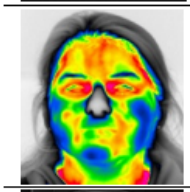
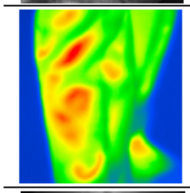


Figure 3.10: Result analysis after encryption and decryption using optimal key-based DES cryptography.

selection using GDWCN-PSO gives maximum MSE and lowest PSNR between original and encrypted images. Also, the encrypted image exhibits the lowest SSIM with the original image.

Furthermore, it is worth mentioning that PSNR and SSIM should be as low as possible for a better encryption scheme, and MSE should be high between the original and the encrypted image. Similarly, for lossless decryption, SSIM is 1, and PSNR value is infinity. It is clearly visible from the results that the proposed method got high MSE,

Table 3.7: Result showing performance of the proposed algorithm with other algorithms.

IMAGE	ALGORITHM	MSE	PSNR	SSIM
	PSO	15356.08331	6.267999014	0.006746183
	PSO-GA	15333.38956	6.274421913	0.007741397
	DWCN-PSO	15281.14934	6.289243408	0.006918166
	PROPOSED	**15377.83723**	**6.261851011**	**0.006345432**
	PSO	13381.12486	6.865877377	0.006025195
	PSO-GA	13445.14439	6.845148904	0.005720998
	DWCN-PSO	13490.02205	6.830677013	0.005875338
	PROPOSED	**13542.09543**	**6.813944909**	**0.005364083**
	PSO	14328.69639	6.568736802	0.007262553
	PSO-GA	14303.87766	6.576265758	0.007558277
	DWCN-PSO	14299.20357	6.577685117	0.007108916
	PROPOSED	**14364.21586**	**6.557984379**	**0.006809591**
	PSO	9462.181618	8.370890812	0.008388564
	PSO-GA	9492.701318	8.356905446	0.008650164
	DWCN-PSO	9551.032946	8.330300177	0.008418722
	PROPOSED	**9567.199984**	**8.322955087**	**0.008279732**
	PSO	16744.25643	5.892144946	0.005556172
	PSO-GA	16724.28948	5.897326848	0.005473739
	DWCN-PSO	16689.81471	5.906288458	0.005932591
	PROPOSED	**16892.73676**	**5.853803464**	**0.005293374**
	PSO	7971.552565	9.115374466	0.009257608
	PSO-GA	7934.729378	9.135482416	0.009928906
	DWCN-PSO	7968.350065	9.117119556	0.009143673
	PROPOSED	**7981.036598**	**9.110210585**	**0.009135273**

low PSNR, and low SSIM in the case of all original-encrypted images. Moreover, we also got SSIM Value 1, $MSE = 0$, and $PSNR = infinity$ between all original-decrypted images. Thus, the optimal key selected gives minimum information to the adversary while maintaining the quality of the decrypted images. This shows the consistency of our proposed approach.

3.7 Summary

The advancement of the IoT is anticipated to transform the healthcare sector and may lead to the growth of the Internet of Medical Things (IMoT). Today's IoT revolution augments human services with promising financial and social perspectives. However, this technology poses new challenges for medical data security. This study examined medical data security through the use of an innovative optimal cryptographic key generation model. In this study, an enhanced version of DWCN-PSO was proposed based on the concepts of selection, crossover, and mutation just after the update process. This increases the speed of convergence without losing the diversity of particles and thus improves the performance of DWCN-PSO. It retains the diversity of the particles by the mutating capabilities and the divide and conquer nature of the algorithm. GDWCN-PSO performs better than DWCN-PSO and PSO because of the aforesaid reasons. In a modern hospital, patient information is stored on a cloud server, thereby opening up security challenges. In our proposal, an optimal key will be chosen using GDWCN-PSO to optimize the security of the encryption and decryption process. Moreover, we have also introduced MGDWCN-PSO to address real-life problems with more than one objective function.