Wireless Personal Communications (2021) 119:585–616
https://doi.org/10.1007/s11277-021-08225-5

Cluster Head Selection in Heterogeneous Wireless Sensor Network Using a New Evolutionary Algorithm

Sumika Chauhan[1] · Manmohan Singh[1] · Ashwani Kumar Aggarwal[1]

Accepted: 3 February 2021 / Published online: 17 February 2021
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC part of Springer Nature 2021

Abstract

In wireless sensor network (WSN), limited energy resources with the nodes is a complex challenge as far as data routing, collecting and aggregating the data is concerned as all these processes are energy demanding. Network lifetime, stability period, and potential of the WSN are some of the parameters which are to be maximized subject to the constraints. The cluster head selection in the heterogeneous wireless sensor network has not been explored much and needs to be improved further to discover the potential of WSN in this area. In this study, optimal cluster head selection in heterogeneous wireless sensor network through Diversity-Driven Multi-Parent Evolutionary Algorithm with Adaptive Non-Uniform Mutation has been suggested. The efficacy of the proposed technique is tested on Classical Benchmark Functions, and obtained results are compared with the state of the art of algorithms. This algorithm is also validated on a heterogeneous wireless sensor network with cluster head selection as a multi-objective optimization problem. The residual energy of sensor node and distance travelled are to be optimized in order to minimize the fitness function. Simulation suggested that the proposed algorithm is found to be reliable and outperforms in terms of remaining energy of nodes, alive nodes versus round, dead nodes versus rounds, the lifespan of network, throughput, and stability period.

Keywords Evolutionary algorithm · Heterogeneous wireless sensor network · Cluster-head selection · Clustering · Fuzzy set theory

✉ Sumika Chauhan
sumika_pei1806@sliet.ac.in; sumi.chauhan2@gmail.com

Manmohan Singh
manmohansingh@sliet.ac.in; singh_manmohan@msn.com

Ashwani Kumar Aggarwal
ashwani.ist@sliet.ac.in; ashwani.ist@gmail.com

[1] Department of Electrical and Instrumentation Engineering, Sant Longowal Institute of Engineering and Technology, Longowal 148106, Punjab, India

1 Introduction

Wireless sensor network (WSN) is the set of teeny, battery-operated sensor nodes in the defined covered area. These sensor nodes have the responsibility to collect, process, and aggregates the data and then forward it to the processing center called a sink. There are many challenges associated with these networks, such as the data acquisition is accomplished in a remote or dense area, which can be inaccessible, limited energy resources, limited onboard processing and computing, and the distributed nature of WSNs. Various energy-aware routing protocols are present in the literature to address these issues [1–3] A WSN has wide scope in many fields such as monitoring, tracking, and surveillance in the military [4, 5], natural calamity conditions [6], health monitoring in biomedical field [7, 8], and in dangerous environment analysis and seismic sensing [9, 10].

The deployment of sensor nodes is accomplished randomly or uniformly. On the basis of deployment WSN can be classified into two classes: structured and unstructured. An unstructured WSN carries a heavy group of sensor nodes. The maintenance of the network is difficult in the unstructured network as it is a challenging task to manage, process, connection, and detection of failures due to a large number of sensor nodes. In the structured WSN, there are few sensor nodes, and deployment of these nodes are pre-planned, so it becomes easy to maintain the network. The lifespan of sensor nodes relies upon the residual energy in the nodes; hence it becomes vital to use the energy-efficient resources in WSNs. The conventional routing protocols include direct-transmission (DT) [11] and minimum-transmission energy (MTE) [12], are not capable of distributing the energy load among nodes in the wireless sensor network. In the direct-transmission, sensors send information directly to the sink, due to this fact, nodes having more distance from the sink would die first. In the MTE nodes, follow the minimum distance path, where cost reflects the transmission power expended. For this problem, the LEACH protocol was proposed in the literature. To enhance the lifetime of a network and to save energy, nodes are grouped and termed as clusters, and a cluster head is selected in each cluster. Remaining sensor nodes behave as cluster members. Each sensor node must relate to only one cluster. The cluster head collects the data from each cluster members and aggregates information and transmits the useful information to the base station (BS) or sink via single-hop or multi-hop communication. The clustering in WSN can be classified in the following: centralized, distributed and hybridized clustering. Also, clustering can be achieved in equal or unequal manner. The equal clustering algorithms are LEACH, HEED etc. and unequal clustering algorithms are ULEACH, UHEED, EEUC, EDUC etc. the hot spot problem is associated with equal clustering due to which unbalanced energy consumption in clusters. The sensor network also categorized as homogeneous WSN and Heterogeneous WSN. If all sensor nodes have the same energy amount, this is called a homogeneous network. In contrast, if some percentage of sensor nodes contained extra energy than other nodes is called a heterogeneous network.

Different type of engineering optimization problems are tackled by numerous optimization algorithms available in the literature including slime mould algorithm (SMA) [13], salp swarm algorithm (SSA) and its variants [14–16], sine cosine algorithm (SCA) [17–19], Harris hawk optimizer [20, 21], grey wolf optimizer (GWO) [22–24], gravitational search algorithm (GSA) [25], Multi-verse optimizer (MVO) [26], whale optimization algorithm (WOA) [27, 28]. The fuzzy theory is also investigated in the literature as in [29, 30].

In this paper, Diversity driven multi-parent evolutionary algorithm-based optimal cluster-head selection in the heterogeneous WSN is proposed. The paper is arranged as follows: the first section described the background and brief introduction to the paper. The second section presents the literature review, and the third section highlights the contributions of the paper. In the fourth section, the proposed algorithm is detailed, and the next fifth section elaborates the problem formulation. The sixth section presents the results and discussion of the proposed work, presenting the competitive performance of the proposed algorithm. Finally, the seventh section concludes the paper.

2 Related Works

In this section, various cluster head selection and energy-efficient techniques are discussed, which are gives the motivation to design the proposed algorithm. The architecture of wireless sensor nodes with the cluster head and sink is given in Fig. 1.

Various energy based routing protocols are explored in the literature such as low energy adaptive clustering hierarchy (LEACH) version of LEACH (VLEACH) [31], LEACH-B protocol [32], hybrid energy-efficient distributed (HEED) protocol [33], adaptive multi clustering algorithm based on fuzzy logic (Adaptive MCFL) [34], and Cluster chain weight metrics approach (CCWM) [35]. The LEACH protocol guarantees that the energy load is well distributed dynamically chosen based on a priori optimal probability. The role of the cluster head is performed through each sensor node by rotating uniformly. Cluster-head.

In the literature, various optimization-based energy-aware routing protocols and cluster head selection techniques are also proposed described in Table 1. The parameters considered and identified problems are summarised in this section. Every method delivers new contributions to energy-efficient routing or cluster head selection.

Ghugar et al. proposed a protocol layer trust-based intrusion detection system (LB-IDS) [44] and physical layer trust-based intrusion detection system (PL-IDS) [45] to provide security for wireless sensor network. A light weight trust management technique based on penalty and reward policy has been proposed by Sahoo et al. to detect malicious, selfish and compromised nodes [46]. Bhoi et al. proposed a density-based

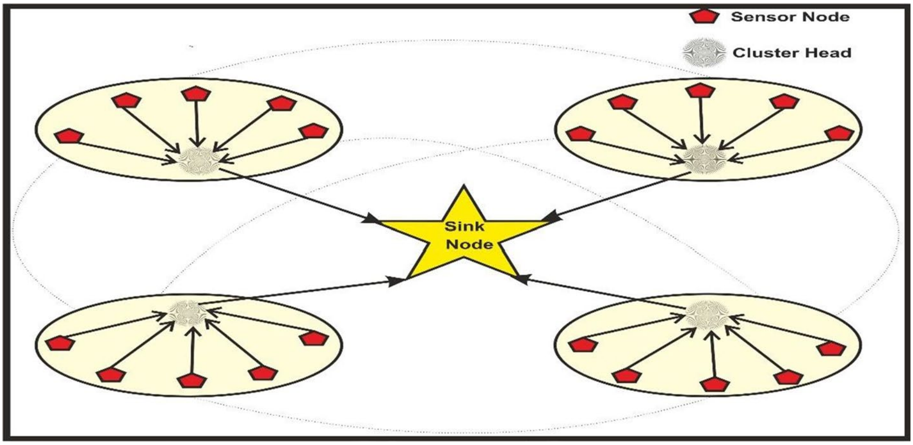

Fig. 1 The Wireless sensor node architecture with cluster heads and sink

Table 1 Literature review

Study reference	Approach	Objectives considered to improve	Problem identified
Shankar et al. [36]	Whale optimization	Residual energy, network lifetime, throughput and stability period	Hot-spot problem is not discussed
Guo et al. [37]	Hybrid genetic Optimization	Energy consumption and network lifetime	Clustering is missing
Kuila et al. [38]	PSO	Network lifetime, energy consumption, dead nodes, delivery of data packets	Do not consider the stability period of network
Zeng et al. [39]	Harmony Search	Network lifetime, energy consumption and path length	CH selection and clustering not discussed
Karaboga et al. [40]	ABC algorithm	Clustering, network lifetime	Network lifetime further improved by the better search process
Chen et al. [41]	Hybrid ABC	Transmission path	No other parameters are discussed
Kumar et al. [42]	FABC algorithm	Delay, distance and energy consumption	Network longevity, throughput, hot-spot problem and clustering is not discussed
Vinodhini et al. [43]	Heuristic approach	Packet delivery ratio, end to end delay, throughput, duty cycle, energy consumption and network lifetime	Clustering and stability period are not discussed
John et al. [2]	Taylor Crow Optimization	Distance between nodes, the energy of nodes, traffic density and delay	Only intra-cluster distance is considered

clustering method to identify the faults in WSNs [47], a software defined network based fault detection in WSN [48] and a geometric constraint based range free localization scheme for WSNs [49]. A fault detection method for heterogeneous WSN is proposed by Swain et al. by incorporating three phases namely; clustering phase, fault detection phase and fault classification phase [50].

3 Contribution of the Paper

The significant contributions of the paper are talked about as follows.

(1) *Proposed new optimization technique* A diversity driven multi-parent evolutionary algorithm is proposed in this paper, which furthers the process of searching the search space by sensing the diversity of the population to avoid local optimum.
(2) *Optimized cluster head selection for the heterogeneous network* The proposed algorithm is modified for the application on heterogeneous wireless sensor node network to choose cluster head in order to enhance the life of the network.
(3) *Use of fuzzy set theory to formulate fitness function and uneven clustering* Fuzzy set theory is used in decision making to choose a solution effectively out of the available non-inferior solutions. Uneven clustering is also performed using the fuzzy theory.

Here a fitness function is designed to evaluate cost function for the selection of cluster head. In this paper, the residual energy of sensor nodes and the distance between nodes and sink is considered in the creation of the objective function. Also, to approve the proposed optimization algorithm, the wireless sensor network is simulated. Also, the presentation of the calculation is widely dissected with different points of view, such as alive nodes, residual energy of nodes and cluster head. The efficacy of the optimization algorithm is tested on benchmark functions, which include unimodal, multi-modal, separable, and non-separable characteristics.

4 Proposed Algorithm

This section is covering the detailed steps of operation performed. The proposed algorithm is named as Diversity-driven multi-parent evolutionary algorithm (DDMPEA) with adaptive non-uniform mutation (ANUM) [51]. In this algorithm, the multi-parent selection strategy is adopted for crossover operation to generate new offspring. Some defined percentages of individuals are selected randomly from the initial population, ensuring that parents are chosen once in an iteration. For mutation, Adaptive non-uniform mutation (ANUM) is applied conditionally based on a probability index calculated from the fitness variance of the population, which signifies the actual aggregation of the population in the search space defined by the problem objective. This mutation helps in diverting the population from local minima as sensed by the search space aggregation. The steps for the proposed algorithm are illustrated in detail as follows:

4.1 Initialization of Population

The initial *NP* numbers of members are randomly generated within the search space using uniform distribution using the following equation

$$x_{ij} = x_j^{\min} + r_{ij}\left(x_j^{\max} - x_j^{\min}\right); \quad (i = 1, 2, \ldots, NP; j = 1, 2, \ldots, D) \tag{1}$$

where r_{ij} the uniform random number for i^{th} member of the population and j^{th} dimension of the variable, *NP* is the size of the population, and *D* is the dimension of the search space. x_j^{min} and x_j^{max} are the minimum and maximum of j^{th} dimension of the control variable.

4.2 Opposition-Based Learning

The optimization algorithms initially randomly chose the members, and to obtain the optimal solution quality of these members is improved. The computational time is calculated using the distance and initial guesses obtained through an optimum solution. By utilizing the opposite solution of initial guesses simultaneously, it can be possible to enhance the search process to obtain better solutions. The initial position, either from the uniformly random distribution or its opposite guess, is chosen. The initial position and its opposite are being used to compute the objective function. Based on this, a decision has been taken for solutions to be considered primarily, which has the capability to accelerate the convergence. The same method can be used not only to initial positions but also continues to each position in the current population [30]. The oppositional learning in population is initialized as

$$x_{i+NP,j}^t = x_j^{\min} + x_j^{\max} - x_{ij}^t(i = 1, 2, \ldots, NP; j = 1, 2, \ldots, D) \tag{2}$$

4.3 Multi Parent DE Crossover

In the multi-parent crossover, three parents are selected randomly for crossover to generate offspring. A pool of mating parents is created by selecting the best P% of the total population. Out of this pool three members x_{r1}, x_{r2} and x_{r3} are randomly chosen for the crossover operation and create an offspring as per the following equation

$$O_{ij}^t = x_{r1,j}^t + \alpha_{ij}\left(x_{r2,j}^t - x_{r3,j}^t\right); \quad \left(i = 1, 2, \ldots, \frac{1}{3} \cdot NP \cdot \frac{P}{100}; \quad j = 1, 2, \ldots, D\right) \tag{3}$$

where α_{ij} is a weighting factor and is a random number from a normal distribution with a mean of 0.7 and a variance of 0.1. *t* represents the index for the current generation. It is taken care that once a member is selected for the crossover is not selected again in the same iteration. The following equation is used to fix the off-limit offspring.

$$O_{ij}^t = \begin{cases} \frac{x_j^{min} + O_{ij}^t}{2}, & \text{if } O_{i,j}^t < X_j^{min} \\ \frac{x_j^{max} + O_{ij}^t}{2}, & \text{if } O_{i,j}^t > X_j^{max} \end{cases} \tag{4}$$

After the multi-parent crossover operation, fitness value is evaluated from newly created offspring by solving the objective definition of the problem at hand [52].

4.4 Adaptive Non-uniform Mutation Strategy

The mutation is an operator used to maintain diversity from one generation to next. Fitness variance of the population is used as a signal of the diminishing diversity of the population. Whenever an algorithm tends to stick in a local minimum, the fitness variance of the population tends to become zero. This condition is used to trigger the adaptive non-uniform mutation to perturb the population from being stuck somewhere in a local optimum. Hence, the proposed algorithm kicks out the population from premature convergence situation by using an adaptive non-uniform mutation operator. The adaptive non-uniform mutation is performed according to Eq. (5).

$$Om_{ij}^{t} = O_{ij}^{t}(1 + 0.5\eta); \quad \left(i = 1, 2, \ldots, \frac{1}{3} \cdot NP \cdot \frac{P}{100}; \quad j = 1, 2, \ldots, D\right) \tag{5}$$

where η stands for the weighting factor from a normal distribution with a value of a mean as 0 and a variance of 1[53]. Om_{ij}^{t} is a mutated version of offspring O_{ij}^{t}. This mutation is carried out if $r_i(0,1) < P_m$, where $r_i(0,1)$ is a uniform random number between 0 and 1 for i^{th} offspring. P_m is the mutation probability and is defined as

$$P_m = \begin{cases} \frac{\exp(-h)}{5.0}, & \sigma^2 < \sigma_1 \\ 0, & \sigma^2 \geq \sigma_1 \end{cases} \tag{6}$$

where σ_1 is the threshold value of variance based on the search space range of the search variables as

$$\sigma_1 = \left(x_j^{max} - x_j^{min}\right)\frac{1}{100} \tag{7}$$

σ^2 is the fitness variance of the population. The degree of diversity, h, in the t^{th} iteration is defined as

$$h = \frac{max_{1 \leq i \leq NP}\left(x_{i,j}^{t} - x_{best2}^{t}\right)}{max_t\left\{max_{1 \leq i \leq NP}\left\{x_{i,j}^{t} - x_{best2}^{t}\right\}\right\}} \tag{8}$$

where x_{best}^{t} is a position of the variable in the search space corresponding to the best fitness value and d_2 is $L_2 - norm$ of d.

The parameter h is the normalized distance of the member of the population from the optimum point, not necessarily the global optimum. The value of h signifies the diversity in the population. When the value of h is high, particles are more dispersed in space and requiring the small value of mutation probability. When the value of h is small, particles are congregated to assemble in the near proximity of a probable local optimum and hence to require significant mutation probability. The essential requirement is to improve the solutions in terms of enhanced ability of exploration is fulfilled by mutation strategy. When the algorithm stuck at local minima, an individual's position and its interaction with the rest of the population are relatively estimated by employing the concept of the fitness variance.

Algorithm-1: Pseudocode of the proposed method for optimization problems

```
• Initialize initial population $x_{ij}$ having NP members using Eq. (1)
• By using opposition based learning get NP members using Eq. (2)
• Evaluate Objective function, $f(x_{ij})$ $\forall(i = 1,2,\ldots,2NP)$
• Select best NP members out of 2NP
• Select member, xbest(j), with best function value and designate as gbest
• For $IT = 1{:}maxIT$
    ○ Sort according to $f(x_{ij})$and select best P% of population to create mating pool
    ○ For $i = 1{:}\frac{1}{3}.NP.\frac{P}{100}$
        ▪ Randomly select three members, $x_{r1}, x_{r2}$ and $x_{r3}$ out of mating pool
        ▪ Perform crossover using Eq. (3) to get offsprings $O_{ij}^{t}$
        ▪ Fix the off − limit offsprings, $O_{ij}^{t}$, using Eq. (4)
    ○ EndFor
    ○ Evaluate Objective function, $f(O_{ij}^{t})$
    ○ Calculate the fitness variance of the population using Eq. (9)
    ○ Calculate degree of diversity, h, using Eq. (8)
    ○ Calculate $P_m$ using eq. (6)
    ○ If $(r(0,1) < P_m)$
        ▪ Mutate the Offsprings, $O_{ij}^{t}$, to get $Om_{ij}^{t}$ using Eq. (5)
    ○ EndIf
    ○ Evaluate Objective function, $f(Om_{ij}^{t})$
    ○ Find oppositionally learned population $Opp_{ij}^{t}$ and $Oppm_{ij}^{t}$ of $O_{ij}^{t}$ and $Om_{ij}^{t}$using Eq. (2)
    ○ Evaluate Objective function $f(O_{ij}^{t})$ and $f(Om_{ij}^{t})$
    ○ Combine main population, $O_{ij}^{t}$, $Om_{ij}^{t}$, $Opp_{ij}^{t}$ and $Oppm_{ij}^{t}$ to create population pool
    ○ Sort members of this population pool and select best NP members to replace
      the main population
    ○ Update the value of xbest(j) and gbest
• EndFor
```

The fitness variance, which is used as an indicator of a stuck population, is calculated using Eq. (9).

$$\sigma^2 = \sum_{i=1}^{NP} \left[\frac{f_i - f_{avg}}{f} \right]^2 \tag{9}$$

where f_{avg} is the average of fitness values of population, f is the returning factor and f_i is the fitness of i^{th} individual. The returning factor is used to control the fitness variance of particles. The value of returning factor is calculated as

$$f = \begin{cases} max\left\{ \left| f_i - f_{avg} \right| \right\}, & \left\{ \left| f_i - f_{avg} \right| \right\} < 1 \\ 1, & otherwise \end{cases} \tag{10}$$

This fitness variance hints about the degree of the congregation of the population. The smaller the value of fitness variance, the nearer the particles are assembling at a point in the search space; otherwise, they are randomly dispersed in the search space [54].

A pool is created involving the present population at time t, newly created offspring from crossover and the mutated offspring when the following steps are performed. This population from this pool is arranged according to their cost evaluations. Finally, the best NP members are selected from this pool to substitute the individuals of the present population. The gradual procedure is elaborated in Algorithm-1. The flow chart of proposed algorithm is given in Fig. 2.

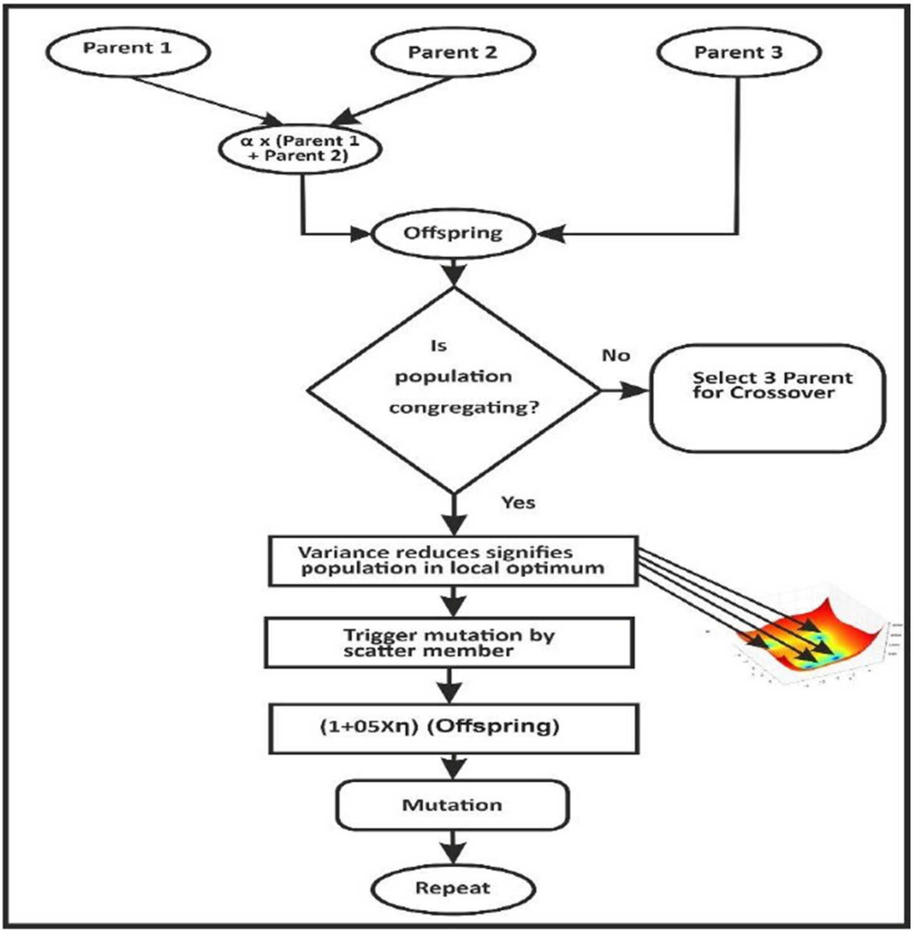
Parent 1
Parent 2
Parent 3
α x (Parent 1 + Parent 2)
Offspring
Is population congregating?
No
Select 3 Parent for Crossover
Yes
Variance reduces signifies population in local optimum
Trigger mutation by scatter member
(1+05Xη) (Offspring)
Mutation
Repeat

Fig. 2 Flow Chart of Diversity driven multi-parent evolutionary algorithm with adaptive non-uniform mutation

5 Problem Formulation for Cluster-Head Selection in Heterogeneous Wireless Sensor Networks

In this section, the formulation of the problem for heterogeneous wireless sensor networks is discussed in which a portion of the nodes has an additional energy source initially. The scenario in which a population percentage of sensor nodes carries additional energy resources compared to other sensor nodes in different proportions is considered. In this structure, the total number of nodes, n, are divided into three categories; advanced nodes, intermediate nodes and normal nodes expressed by N_{ADV}, N_{INT} and N_{NRM} respectively.

$$N_{ADV} = n \times m \tag{11}$$

$$N_{INT} = n \times m_0 \tag{12}$$

$$N_{NRM} = n \times (1 - m - m_0) \tag{13}$$

where m_0 and m are the fractions of intermediate and advanced nodes, respectively. The energy of the advanced nodes and intermediate nodes are α and β times higher in energy as compared to normal nodes, respectively [55]. The energy of normal, intermediated and advanced nodes is as follows

$$E_{ADV} = E_0 \times (1 + \alpha) \times n \times m \tag{14}$$

$$E_{INT} = E_0 \times (1 + \beta) \times n \times m_0 \tag{15}$$

$$E_{NRM} = E_0 \times \left(1 - m - m_0\right) \times n \tag{16}$$

$$E_T = E_{ADV} + E_{INT} + E_{NRM} \tag{17}$$

$$E_T = E_0 \times (1 + \alpha) \times n \times m + E_0 \times (1 + \beta) \times n \times m_0 + E_0 \times \left(1 - m - m_0\right) \times n \tag{18}$$

$$E_T = n \times E_0 \times \left(1 + \beta \times m_0 + m \times \alpha\right) \tag{19}$$

where the total energy of the network is denoted by E_T and E_0 represents the initial energy of the sensor node.

The problem of cluster head selection in a heterogeneous wireless sensor network is created as a multi-objective optimization problem. The two objectives which are considered are residual energy of the node and the distances between cluster heads and base station and cluster head and other nodes. The objectives considered are conflicting in nature in the sense that when distance objective is decreasing, the residual energy of the nodes increases as with lesser distances consumed energy decreases. Search space is searched for non-inferior solutions by employing the proposed algorithm DDMPEA-ANUM. Fuzzy set theory is used for decision making to find the best solution.

5.1 System Energy Dissipation Model

The calculation of energy dissipation for data transmission in WSN is carried out by comparing the distance travelled like the distance between the transmitter and the receiver, and the critical value $d_0 = \sqrt{\frac{E_{efs}}{E_{amp}}}$. In transmission of each packet energy loss occurs which follows a free-space and multi-path fading model. A model for radio transmission and reception is shown in Fig. 3.

The energy required in the transmission of k-bit data at the distance d, is given by $E_{dis}(l, d)$ and calculated as

$$E_{dis}(k, d) = k \times E_{elec} + k \times E_{efs} \times d^2 \ \forall d \le d_0 \tag{20}$$

$$E_{dis}(k, d) = k \times E_{elec} + k \times E_{amp} \times d^4 \ \forall d > d_0 \tag{21}$$

where E_{efs} is for free space energy model, E_{amp} is an energy consumed in the power amplifier and E_{elec} represents the electronic energy of the circuit given as

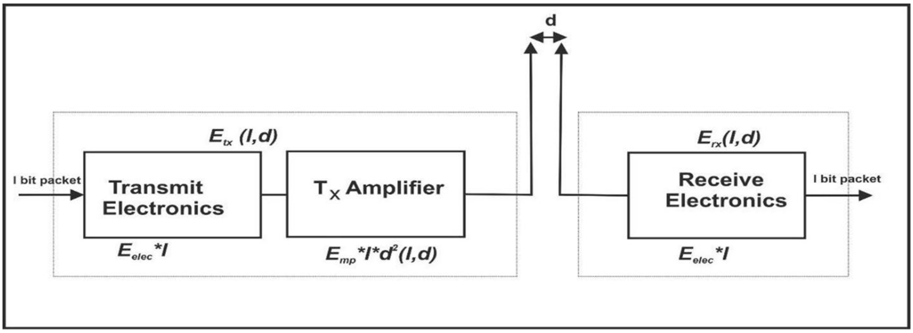

Fig. 3 Radio energy dissipation model

$$E_{elec} = E_{TX} + E_{DA} \tag{22}$$

where E_{TX} is transmitter circuit's energy and E_{DA} is energy required in data aggregation. d is the distance between the sender node and the receiver node. k represents the packet size of the data transmitted. The energy consumed during the reception of data per bit is given by

$$E_{dis}(l) = E_{elec} \times k \tag{23}$$

Hence the total energy consumed by cluster head nodes is given by

$$E_C^{t_r}(i) = N_{CH} \times \left(E_{dis}(k) + E_{elec}\right) \tag{24}$$

and the total energy dissipated by other nodes is given by

$$E_C^{t_r}(i) = E_{dis}(k, d) \tag{25}$$

5.2 Residual Energy of the NODE

The energy estimation of each node is revise subsequently to sending or getting k. bytes of information. This procedure is rehashed until each node is said to be a dead node. A node is said to be dead node when the leftover energy gets negative or zero. The residual energy of the node in the current round is considered as the first objective for evaluating the optimum cluster. The cluster head consumes more energy as compared to other sensor nodes as it has the responsibility of transmitting, receiving, aggregating, and forwarding the data. After every round of communication the cluster head decreases its energy more rapidly than the other nodes. Therefore, the re-establishment of cluster heads in each round on the basis of the residual energy of a node becomes obligatory. The residual energy of the node is represented mathematically as

$$E_R^{t_r+1}(i) = E_R^{t_r}(i) - E_C^{t_r}(i); \quad (i = 1, 2, \dots, n) \tag{26}$$

where $E_R^{t_r}$ is the residual energy of i^{th} node after t_r number of rounds and $E_C^{t_r}$ is the energy consumed in the t_r^{th} round. It is clear that for each type of heterogeneous nodes, the value of $E_R^{t_r}$ will be different according to the type of node. For the first round $\left(t_r = 1\right)$, $E_R^{t_r}$ would be

E_{ADV}, E_{INT} and E_{NRM} for advanced, intermediate, and normal nodes, respectively. It is obvious that the node with the higher value of energy wins the selection of a node to be cluster head.

5.3 Distance Between Node and Sink

Any communication that takes place between nodes and cluster head or between nodes and sink consumes some energy according to the role performed by the node. Higher the distance between the nodes while communicating; higher energy will be required and vice-versa. The distance between sensor nodes to sink is calculated as

$$D_i = \sqrt{\left(x_{BS} - x_i\right)^2 + \left(y_{BS} - y_i\right)^2}; \quad (i = 1, 2, \dots, n) \tag{27}$$

This is the second fitness objective for the multi-objective problem at hand. The distance between sensor nodes and cluster heads is calculated as

$$D_{ij} = \sqrt{\left(x_j - x_i\right)^2 + \left(y_j - y_i\right)^2}; \quad \left(i = 1, 2, \dots, n; \quad j = 1, 2, \dots, N_{CH}\right) \tag{28}$$

where N_{CH} are the number of cluster heads specified.

5.4 Decision Making

In the problem at hand, two objectives are considered, which are of conflicting nature as residual energy is to be maximized, and the distances between the cluster heads and base station are bound to be reduced. Hence, to find the best-compromised solution, the fuzzy set theory is employed. In general, the decision-maker's assessment is of an imprecise type, and it is worth thinking that the DM might have blurred or imprecise objectives for each objective feature. The Fuzzy Sets are described as membership functions by equations. These functions reflect a membership degree in some fuzzy sets by using values from 0 to 1. The membership function is defined by assuming $\mu\left(F_i^k\right)$ is a decreasing and continuous function that is strictly monotonous. In fuzzy set theory membership values are calculated for both fitness parameters; residual energy of each node (considered as F_i^1) and distance between sink and nodes (considered as F_i^2) as given below

$$\mu\left(F_i^k\right) = \begin{cases} 1; & F_i^k \le F_k^{min} \\ \frac{F_k^{max} - F_i^k}{F_k^{max} - F_i^{min}}; & F_k^{min} < F_i^k < F_k^{max} \ (i = 1, 2, \dots, n; k = 1, 2) \\ 0; & F_i^k \ge F_k^{max} \end{cases} \tag{29}$$

The satisfaction level of F_i^k objective of non-inferior solution arranged in range of [0,1], is reflected by the value of membership function. The accomplishment of k non dominated solutions [56] given below

$$\mu_D^i = \frac{\left[\sum_{k=1}^{2} \mu\left(F_i^k\right)\right]}{\left[\sum_{i=1}^{n} \sum_{k=1}^{2} \mu\left(F_i^k\right)\right]} \tag{30}$$

The solution corresponding to the maximum membership μ_D^i chosen as the best solution. Hence the problem statement becomes

$$\begin{aligned} &Max\left\{\mu_D^i \,:\, i = 1, 2, \ldots, n\right\} \\ &\text{subject to } x_j^{min} < x_{ij} < x_j^{max} \end{aligned} \tag{31}$$

In the proposed work, the clusters are re-elected after each round, and as a result, the load is very much conveyed and adjusted among the nodes of the network. After cluster head selection, uneven clustering is accomplished using fuzzy theory based on the distance between nodes and sink. The working procedure for optimum clustering is given in Algorithm-2.

Algorithm-2: Pseudocode of the proposed method for clustering

```
• $\boldsymbol{For\ i = 1 : NP}$
    ○ $Compute\ the\ residual\ energy\ of\ all\ nodes\ after\ each\ round\ using\ Eq.(26)$
    ○ $Calculate\ the\ distance\ between\ cluster-head\ and\ all\ sensor\ nodes$
    ○ $Evaluate\ the\ membership\ function\ value\ for\ residual\ energy\ matrix$
      $\mu(F_i^1) = \begin{cases} 1 & ; E_R^{t_r}(i) \le E_R^{min} \\ \dfrac{E_R^{max} - E_R^{t_r}(i)}{E_R^{max} - E_R^{min}} & ; E_R^{min} < E_R^{t_r}(i) < E_R^{max} \\ 0 & ; E_R^{t_r}(i) \ge E_R^{max} \end{cases}$
    ○ $Evaluate\ the\ membership\ function\ value\ for\ distance\ objective$
      $\mu(F_i^2) = \begin{cases} 1 & ; D_i \le D^{min} \\ \dfrac{D^{max} - D_i}{D^{max} - D^{min}} & ; D^{min} < D_i < D^{max} \\ 0 & ; D_i \ge D^{max} \end{cases}$
• $\boldsymbol{EndFor}$
• $Find\ out\ the\ accomplishment\ of\ the\ each\ non-dominated\ solution\ using\ Eq.(29)$
• $On\ the\ basis\ of\ highest\ cardial\ ranking\ priority\ choose\ the\ \mu_D^i\ using\ Eq.(30)$
• $Choose\ the\ cluster\ heads\ according\ to\ maximum\ value\ of\ \mu_D^i$
• $Evaluate\ distance\ of\ each\ node\ from\ each\ cluster\ head\ using\ Eq.(28)$
• $Evaluate\ membership\ function\ of\ each\ node\ for\ each\ cluster\ head$
  $\mu(D_{ij}) = \begin{cases} 1 & ; D_{ij} \le D^{min} \\ \dfrac{D^{max} - D_{ij}}{D^{max} - D^{min}} & ; D^{min} < D_{ij} < D^{max}\ ;(i = 1,2,\ldots,n; j = 1,2,\ldots,N_{CH}) \\ 0 & ; D_{ij} \ge D^{max} \end{cases}$
• $Assign\ the\ members\ for\ respective\ cluster\ heads\ on\ the\ basis\ of\ highest\ cardial$
  $priority\ ranking$
```

5.5 Implementation

The proposed protocol works in two stages, namely, setup stage and steady stage (Fig. 4). In this step, the wireless sensor network is formulated by the uniformly random deployment of heterogenous energy sensor nodes termed as advanced, intermediate, and normal nodes at the start. From the second iteration onwards, the locations of these nodes are modified employing the proposed algorithm in the crossover and mutation steps. Then, the decision maker chooses the configuration to be updated to further the process of the algorithm. The base station or sink is placed in the center of the formulated network. After the network formulation, cluster heads are selected, and nodes are assigned to their cluster heads on the basis of the distance between sensor nodes and the sink and the residual energy of nodes. Cluster head selection operation is performed by employing the proposed

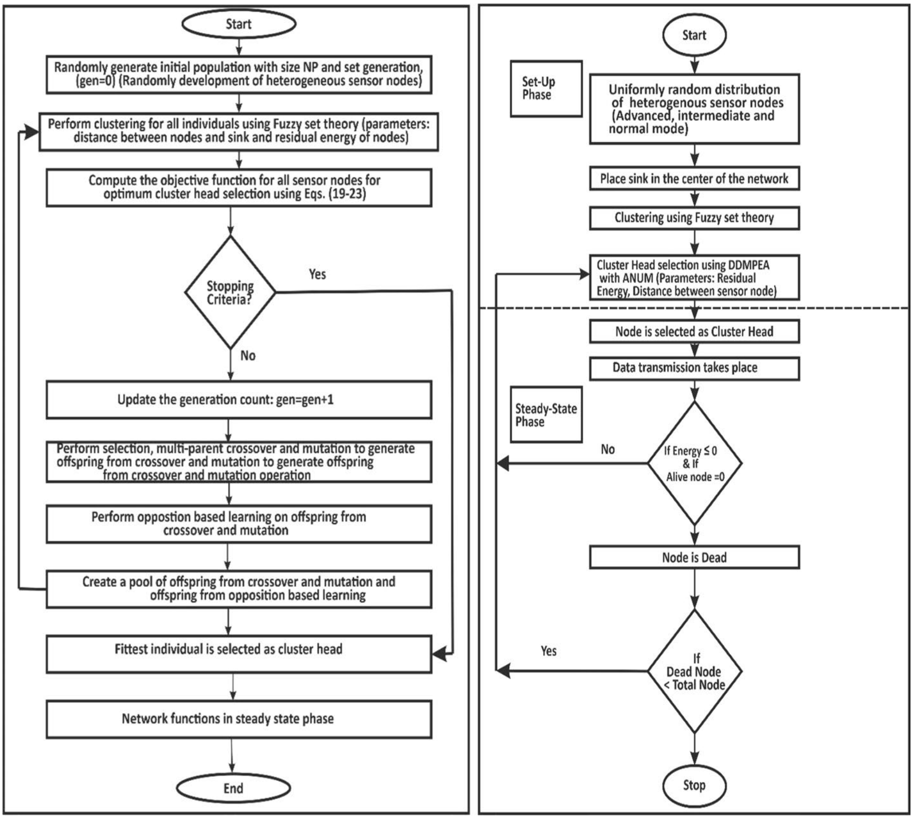
Start
Randomly generate initial population with size NP and set generation, (gen=0) (Randomly development of heterogeneous sensor nodes)
Perform clustering for all individuals using Fuzzy set theory (parameters: distance between nodes and sink and residual energy of nodes)
Compute the objective function for all sensor nodes for optimum cluster head selection using Eqs. (19-23)
Stopping Criteria?
Yes
No
Update the generation count: gen=gen+1
Perform selection, multi-parent crossover and mutation to generate offspring from crossover and mutation to generate offspring from crossover and mutation operation
Perform oppostion based learning on offspring from crossover and mutation
Create a pool of offspring from crossover and mutation and offspring from opposition based learning
Fittest individual is selected as cluster head
Network functions in steady state phase
End
Start
Set-Up Phase
Uniformly random distribution of heterogenous sensor nodes (Advanced, intermediate and normal mode)
Place sink in the center of the network
Clustering using Fuzzy set theory
Cluster Head selection using DDMPEA with ANUM (Parameters: Residual Energy, Distance between sensor node)
Node is selected as Cluster Head
Data transmission takes place
Steady-State Phase
No
If Energy ≤ 0 & If Alive node =0
Node is Dead
Yes
If Dead Node < Total Node
Stop

Fig. 4 Flow chart of DDMPEA with ANUM for CH selection and working of proposed protocol

optimization technique. The parameters utilized in the formulation of fitness function are residual energy of each node and distance between nodes and sink.

The second stage, that is, the steady-state stage, indicates the initialization of data transmission. It is run for some specified number of rounds in every iteration. For communication to take place, the threshold value of the distance is compared with the distance between sensor nodes and the sink. This helps to make the decision that communication occurs via single hop or multiple hops. When data is received by the cluster head, it collects, processed the data and forwards it to the sink. Also, the energy of each node is analysed if it is equal or less than zero if it happens so, the node is flagged as the dead node. This dead node does not participate in the process further after it is flagged as so.

As mentioned in Eq. (9) of the proposed algorithm, the fitness variance is calculated using the objective function value of each member. But in the case of the problem at hand, there is only one objective value collectively for the whole NP population, i.e., the number of live nodes after a certain specified number of data transmission rounds executed. Hence, this parameter is molded according to the problem at hand to the location variance of sensor nodes. The advantage of this modification is that whenever nodes tend to aggregate at a particular location, the mutation is performed. The two-dimensional spatial variance of the sensor node locations is calculated as follows

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\left(x_i-\bar{x}\right)^2+\left(y_i-\bar{y}\right)^2} \tag{32}$$

where $\left(\bar{x},\bar{y}\right)$ is the mean of all the sensor node locations. The value of mutation probability is modified accordingly as

$$P_m = \begin{cases} \frac{\exp\left(-\sigma^2\right)}{5.0}, & \sigma^2 < \sigma_1 \\ 0, & \sigma^2 \geq \sigma_1 \end{cases} \tag{33}$$

The whole working process of DDMPEA with ANUM applied in this work is described in Algorithm-3 as follows.

Algorithm-3: Pseudocode of the proposed method for optimum cluster head selection

```
• Initialize initial population $x_{ij}$ having NP members of sensor nodes using Eqs. (1)
• For $l = 1{:}t_r$
    o Evaluate $\mu_D^i$ using Eqs. (26 − 30) corresponding to present configuration
    o Select cluster heads and assign members employing Algorithm − 2
    o Data transmission takes place (steady state phase)
    o Update residual energies using Eq. (26)
• EndFor
• Save the present configuration and sum of residual energies of all the nodes as gbest
• For IT = 1: maxIT
    o For $i = 1{:}\frac{1}{3}.NP.\frac{P}{100}$
        ▪ Randomly select three members, $x_{r1}, x_{r2}$ and $x_{r3}$ out of mating pool
        ▪ Perform crossover using Eq. (3) to get offspring $O_{ij}^t$
        ▪ Fix the off − limit offspring, $O_{ij}^t$, using Eq. (4)
    o EndFor
    o Calculate the fitness variance of the population using Eq. (32)
    o Calculate $P_m$ using eq. (33)
    o If$(r(0,1) > P_m)$
        ▪ Mutate the Offspring, $O_{ij}^t$, to get $Om_{ij}^t$ using Eq. (4)
    o EndIf
    o Find oppositionally learned population $Opp_{ij}^t$ and $Oppm_{ij}^t$ of $O_{ij}^t$ and $Om_{ij}^t$using Eq. (2)
    o For $l = 1{:}t_r$
        ▪ Evaluate $\mu_D^i$ using Eqs. (26 − 30) corresponding to present configuration
        ▪ Select cluster heads and assign members employing Algorithm − 2
        ▪ Data transmission takes place (steady state phase)
        ▪ Update residual energies using Eq. (26).
        ▪ If energy of any node becomes zero or negative, flag the node as dead node
    o EndFor
    o Combine $O_{ij}^t$, $Om_{ij}^t$, $Opp_{ij}^t$ and $Oppm_{ij}^t$ to create population pool
    o Evaluate total sum of residual energies of all nodes ($O_{ij}^t$, $Om_{ij}^t$, $Opp_{ij}^t$ and $Oppm_{ij}^t$)
    o If total sum of residual energies is greater than gbest, update the nodes with $O_{ij}^t$,
      $Om_{ij}^t$, $Opp_{ij}^t$ and $Oppm_{ij}^t$
• EndFor
```

6 Results and Discussion

The efficacy of DDMPEA with ANUM is tested in two different scenarios. First, on the basic benchmark optimization problems and second on the cluster head selection in case of a heterogeneous wireless sensor network. The simulation is carried out in MATLAB-2015b platform running under the windows-10 operating system with the Intel Core i3 processor @ 1.70 GHz frequency with 4 GB RAM memory. The obtained results are tabulated and discussed in this section.

6.1 Basic Benchmark Functions

This section discusses the performance of the proposed algorithm tested on benchmark functions. The functions $f_1 - f_7$ are unimodal while functions $f_8 - f_{23}$ are multi-modal [18]. The results are obtained and tabulated in terms of mean, median, standard deviation, and worst values in Table 2. The population size is taken as 50, and maximum iterations are set to 1000 for comparison. The other optimization algorithms considered for comparison are SCA [17], PSO[57], TLBO[58], MFO[59], DE[60], WOA [27], GWO[22], CSA[61] and SCA-PSO[18] as available in the literature.

For functions f_1,f_2,f_3 andf_4, the results obtained from the proposed algorithm are competitive to the results obtained from other algorithms. Functions f_8 to f_{13} are multi-modal, separable and non-separable, high dimensional benchmark functions which are usually used to benchmark the ability of an algorithm to tackle with the premature convergence problem. From the results obtained, it is observed that functions f_9, f_{10} and f_{11} gives competitive results as compared to the results from other algorithms in terms of mean, median, standard deviation and worst value. Hence, it can be inferred that the proposed algorithm works effectively to solve the premature convergence problem. This observation shows the better exploration ability of the proposed algorithm. SCA perform better for function f_8. For functions f_{12} and f_{13} classical DE performs better compared to results from other algorithms.

Functions f_{14} to f_{23} are the multi-modal low dimensional functions. It can be observed from Table 1, the proposed algorithm works effectively for these functions too. Functions f_{16}, f_{18}, f_{19}, f_{20}, f_{22} and f_{23} gives optimum global values in terms of mean, median, standard deviation and worst value. The convergence behaviour of benchmark functions and their box plots are given in Figs. 5 and 6., respectively. From the box plots, it is observed that proposed algorithm gives results with minimum standard deviation in case of almost all considered functions.

6.1.1 Statistical Analysis of Benchmark Functions

The comparison of results obtained by the proposed algorithm with the results available in the literature from other algorithms is tabulated based on the mean, standard deviation, median, and worst values from 30 independent runs. For the comparison of results from each run, a nonparametric test, the Wilcoxon rank-sum test is conducted with a 5% significance level, and obtained p values are shown in Table 3. The null hypothesis is rejected if the p-values are less than 0.05, which represents that the better objective functions values are obtained by the proposed algorithm in each case are statistically significant and have not occurred by chance. For this analysis, the best value is considered of which has the smallest mean value. In case if there exists more than one mean value of compared algorithms, then algorithm which have the smallest standard deviation is selected. Not applicable (N/A) is written for the best algorithm because the best algorithm cannot be compared with itself.

From Table 3, it is clear that DDMPEA with ANUM obtained best results for function f_5,f_6,$f_9 - f_{11}$,$f_{16} - f_{19}$ and $f_{21} - f_{23}$. For function f_8 CSA gives the best values. For

Table 2 Results of 23 benchmark functions for different algorithms with initial population NP = 50

f_n		DDMPEA	SCA-PSO	SCA	PSO	DE	TLBO	CSA	MFO	WOA[27]	GWO
f_1	Mean	7.92×10^{-108}	0.00	3.49×10^{-32}	9.03×10^{-01}	9.69×10^{-42}	2.99×10^{-219}	8.61×10^{-04}	2.08×10^{-31}	2.05×10^{-179}	2.57×10^{-70}
	Std	5.65×10^{-108}	0.00	6.90×10^{-32}	1.1512	1.08×10^{-41}	0.00	1.52×10^{-03}	5.07×10^{-31}	0.00	4.70×10^{-70}
	Median	1.057×10^{-125}	0.00	6.19×10^{-33}	2.75×10^{-01}	2.39×10^{-42}	5.79×10^{-220}	2.00×10^{-04}	1.38×10^{-32}	6.55×10^{-181}	5.62×10^{-71}
	Max	3.96×10^{-106}	0.00	2.16×10^{-31}	3.3649	3.37×10^{-41}	1.72×10^{-218}	4.68×10^{-03}	1.61×10^{-30}	1.90×10^{-178}	1.52×10^{-69}
f_2	Mean	1.39×10^{-222}	0.00	3.41×10^{-21}	3.67×10^{-01}	1.63×10^{-24}	1.26×10^{-109}	5.75×10^{-03}	1.00	2.27×10^{-109}	4.86×10^{-41}
	Std	0.00	0.00	9.41×10^{-21}	2.75×10^{-01}	9.17×10^{-25}	1.14×10^{-109}	3.89×10^{-03}	3.16	6.02×10^{-109}	6.61×10^{-41}
	Median	6.84×10^{-221}	0.00	1.64×10^{-22}	2.53×10^{-01}	1.37×10^{-24}	1.00×10^{-109}	5.79×10^{-03}	1.57×10^{-19}	3.23×10^{-114}	2.47×10^{-41}
	Max	4.58×10^{-221}	0.00	1.64×10^{-22}	9.3×10^{-01}	3.64×10^{-24}	3.41×10^{-109}	1.33×10^{-02}	1.00	1.91×10^{-108}	2.25×10^{-40}
f_3	Mean	1.55×10^{-115}	0.00	7.20×10^{-15}	$3.72\times10^{+01}$	2.29×10^{-01}	1.93×10^{-96}	5.13×10^{-03}	$5.00\times10^{+02}$	$1.25\times10^{+04}$	1.97×10^{-20}
	Std	1.09×10^{-114}	0.00	1.29×10^{-14}	$3.35\times10^{+01}$	1.11×10^{-01}	4.90×10^{-96}	8.15×10^{-03}	$1.58\times10^{+03}$	$8.98\times10^{+03}$	3.39×10^{-20}
	Median	1.13×10^{-130}	0.00	1.00×10^{-15}	$2.87\times10^{+01}$	2.23×10^{-01}	9.79×10^{-98}	1.48×10^{-03}	4.60×10^{-10}	$1.36\times10^{+04}$	2.08×10^{-21}
	Max	7.68×10^{-114}	0.00	3.25×10^{-14}	1.15×10^{-01}	4.23×10^{-01}	1.58×10^{-95}	2.64×10^{-02}	$5.00\times10^{+03}$	$2.64\times10^{+04}$	9.91×10^{-20}
f_4	Mean	2.83×10^{-221}	0.00	3.95×10^{-11}	3.5424	2.26×10^{-07}	3.00×10^{-92}	3.60×10^{-03}	9.72×10^{-03}	$2.55\times10^{+01}$	3.22×10^{-18}
	Std	0.00	0.00	4.50×10^{-11}	1.8715	1.16×10^{-07}	3.26×10^{-92}	3.86×10^{-03}	3.07×10^{-03}	$2.58\times10^{+01}$	2.98×10^{-18}
	Median	1.70×10^{-227}	0.00	2.39×10^{-11}	3.2528	2.06×10^{-07}	1.61×10^{-91}	2.34×10^{-03}	1.41×10^{-06}	$1.77\times10^{+01}$	2.21×10^{-18}
	Max	1.38×10^{-219}	0.00	1.42×10^{-10}	6.6687	4.40×10^{-07}	1.05×10^{-91}	1.30×10^{-02}	9.72×10^{-02}	$8.26\times10^{+01}$	1.00×10^{-17}
f_5	Mean	0.00	5.9438	6.7985	$9.15\times10^{+01}$	1.5134	2.85×10^{-02}	1.53×10^{-03}	$7.21\times10^{+01}$	$2.67\times10^{+01}$	$2.62\times10^{+01}$
	Std	0.00	3.587×10^{-01}	3.62×10^{-01}	$7.45\times10^{+01}$	1.5438	7.78×10^{-02}	2.74×10^{-03}	$1.11\times10^{+02}$	3.96×10^{-01}	7.04×10^{-01}
	Median	0.00	6.1431	6.8806	$7.54\times10^{+01}$	8.67×10^{-01}	1.92×10^{-03}	4.88×10^{-04}	5.0679	$2.66\times10^{+01}$	2.62×10^{-01}
	Max	0.00	6.2947	7.2247	$2.11\times10^{+02}$	4.7443	2.49×10^{-01}	8.35×10^{-03}	$2.86\times10^{+02}$	$2.76\times10^{+01}$	$2.71\times10^{+01}$
f_6	Mean	0.00	0.00	3.25×10^{-01}	1.0488	6.88×10^{-23}	6.16×10^{-34}	4.34×10^{-04}	1.79×10^{-31}	2.85×10^{-03}	3.00×10^{-01}
	Std	0.00	0.00	1.75×10^{-01}	1.0618	8.17×10^{-22}	1.30×10^{-33}	7.83×10^{-04}	2.64×10^{-31}	1.08×10^{-03}	2.59×10^{-01}
	Median	0.00	0.00	2.57×10^{-01}	7.45×10^{-01}	3.14×10^{-22}	0.00	1.49×10^{-04}	9.24×10^{-32}	2.74×10^{-03}	2.50×10^{-01}
	Max	0.00	0.00	6.17×10^{-01}	3.464	2.35×10^{-21}	3.08×10^{-33}	2.55×10^{-03}	8.35×10^{-31}	4.75×10^{-03}	7.50×10^{-01}
f_7	Mean	8.63×10^{-05}	1.5×10^{-04}	6.83×10^{-04}	5.53×10^{-03}	2.78×10^{-03}	2.19×10^{-03}	7.67×10^{-04}	2.345×10^{-03}	8.79×10^{-04}	6.56×10^{-04}
	Std	7.12×10^{-05}	$6.40\times10^{+01}$	5.61×10^{-04}	6.05×10^{-03}	8.25×10^{-04}	1.10×10^{-03}	6.38×10^{-04}	1.38×10^{-03}	9.04×10^{-04}	3.98×10^{-04}
	Median	6.63×10^{-05}	1.46×10^{-04}	5.27×10^{-04}	3.334×10^{-03}	2.77×10^{-03}	2.75×10^{-03}	5.78×10^{-04}	2.25×10^{-03}	6.23×10^{-04}	4.93×10^{-04}
	Max	3.40×10^{-04}	3.08×10^{-04}	1.89×10^{-03}	2.04×10^{-02}	4.44×10^{-03}	5.73×10^{-03}	2.19×10^{-03}	4.71×10^{-03}	2.60×10^{-03}	1.32×10^{-03}

Table 2 (continued)

f_n		DDMPEA	SCA-PSO	SCA	PSO	DE	TLBO	CSA	MFO	WOA[27]	GWO
f_8	Mean	$-1.30\times10^{+08}$	$-2.62\times10^{+03}$	$-2.30\times10^{+03}$	$-3.98\times10^{+03}$	$-4.18\times10^{+03}$	$-3.68\times10^{+03}$	$-4.18\times10^{+03}$	$-3.21\times10^{+03}$	$-1.25\times10^{+05}$	$-6.02\times10^{+03}$
	Std	$5.33\times10^{+08}$	$1.40\times10^{+02}$	$9.39\times10^{+01}$	$1.68\times10^{+03}$	0.00	$1.80\times10^{+02}$	1.91×10^{-03}	$3.41\times10^{+02}$	$7.07\times10^{+02}$	$4.62\times10^{+02}$
	Median	$-1.67\times10^{+06}$	$-2.63\times10^{+03}$	$--2.28\times10^{+03}$	$-3.81\times10^{+03}$	4.44×10^{-15}	$-3.73\times10^{+03}$	$-4.18\times10^{+03}$	$-3.18\times10^{+03}$	$-1.24\times10^{+04}$	$-6.08\times10^{+03}$
	Max	$-2.01\times10^{+05}$	$-2.38\times10^{+03}$	$-2.12\times10^{+03}$	$-2.36\times10^{+03}$	4.44×10^{-15}	$-3.36\times10^{+03}$	$-4.18\times10^{+03}$	$-2.76\times10^{+03}$	$-1.09\times10^{+04}$	$-5.17\times10^{+03}$
f_9	Mean	0.00	0.00	2.88×10^{-04}	$2.24\times10^{+01}$	0.00	1.99	9.12×10^{-05}	$1.83\times10^{+01}$	0.00	0.00
	Std	0.00	0.00	9.12×10^{-04}	$1.15\times10^{+01}$	0.00	2.2494	9.95×10^{-05}	$1.34\times10^{+01}$	0.00	0.00
	Median	0.00	0.00	0.00	$2.06\times10^{+01}$	0.00	1.4924	6.92×10^{-05}	$1.54\times10^{+01}$	0.00	0.00
	Max	0.00	0.00	0.002883	$4.77\times10^{+01}$	0.00	5.9698	2.59×10^{-04}	$4.37\times10^{+01}$	0.00	0.00
f_{10}	Mean	8.88×10^{-16}	8.88×10^{-16}	4.80×10^{-15}	3.0004	4.44×10^{-15}	4.44×10^{-15}	3.64×10^{-03}	4.44×10^{-15}	3.73×10^{-15}	1.44×10^{-14}
	Std	0.00	0.00	1.12×10^{-15}	1.1084	0.00	0.00	3.63×10^{-03}	0.00	2.25×10^{-15}	1.50×10^{-15}
	Median	8.88×10^{-16}	8.88×10^{-16}	4.44×10^{-15}	3.1716	4.44×10^{-15}	4.44×10^{-15}	2.50×10^{-03}	4.44×10^{-15}	4.44×10^{-15}	1.51×10^{-14}
	Max	8.88×10^{-16}	8.88×10^{-16}	7.99×10^{-15}	4.0685	4.44×10^{-15}	4.44×10^{-15}	1.29×10^{-02}	4.44×10^{-15}	7.99×10^{-15}	1.51×10^{-14}
f_{11}	Mean	0.00	0.00	4.17×10^{-10}	6.78×10^{-01}	0.00	2.50×10^{-03}	5.03×10^{-04}	1.63×10^{-01}	2.50×10^{-03}	0.00
	Std	0.00	0.00	1.32×10^{-09}	1.69×10^{-01}	0.00	7.75×10^{-03}	5.02×10^{-04}	1.18×10^{-01}	7.92×10^{-03}	0.00
	Median	0.00	0.00	0.00	6.59×10^{-01}	0.00	4.17×10^{-12}	3.05×10^{-04}	1.40×10^{-01}	0.00	0.00
	Max	0.00	0.00	4.17×10^{-09}	9.91×10^{-01}	0.00	2.45×10^{-02}	1.35×10^{-03}	4.43×10^{-01}	2.50×10^{-02}	0.00
f_{12}	Mean	3.22×10^{-02}	2.24×10^{-02}	6.21×10^{-02}	2.4396	4.71×10^{-32}	4.76×10^{-32}	5.98×10^{-06}	6.22×10^{-02}	5.45×10^{-04}	2.09×10^{-02}
	Std	2.51×10^{-02}	7.24×10^{-03}	1.99×10^{-02}	1.1245	1.15×10^{-47}	1.21×10^{-33}	1.11×10^{-05}	1.31×10^{-01}	3.12×10^{-04}	1.15×10^{-02}
	Median	2.72×10^{-02}	2.24×10^{-02}	6.25×10^{-02}	2.2663	4.71×10^{-32}	4.71×10^{-32}	1.14×10^{-06}	6.96×10^{-32}	3.96×10^{-04}	2.28×10^{-02}
	Max	1.26×10^{-02}	3.64×10^{-03}	8.84×10^{-02}	4.1762	4.71×10^{-32}	5.10×10^{-32}	3.45×10^{-05}	3.11×10^{-01}	1.02×10^{-03}	3.35×10^{-02}
f_{13}	Mean	1.11	5.72×10^{-02}	2.02×10^{-01}	3.18×10^{-01}	1.35×10^{-32}	1.35×10^{-32}	2.62×10^{-05}	4.39×10^{-03}	5.34×10^{-02}	3.50×10^{-01}
	Std	1.25	4.25×10^{-02}	8.05×10^{-02}	3.46×10^{-01}	2.89×10^{-48}	2.89×10^{-48}	4.63×10^{-05}	5.67×10^{-03}	7.07×10^{-02}	1.21×10^{-01}
	Median	3.59×10^{-01}	4.51×10^{-02}	2.00×10^{-01}	1.74×10^{-01}	1.35×10^{-32}	1.35×10^{-32}	8.71×10^{-06}	1.25×10^{-30}	1.54×10^{-02}	3.06×10^{-01}
	Max	2.97	1.73×10^{-01}	3.15×10^{-01}	1.0777	1.35×10^{-32}	1.35×10^{-32}	1.49×10^{-04}	1.09×10^{-02}	1.88×10^{-01}	5.13×10^{-01}

Table 2 (continued)

f_n		DDMPEA	SCA-PSO	SCA	PSO	DE	TLBO	CSA	MFO	WOA[27]	GWO
f_{14}	Mean	1.4491	9.98×10^{-01}	1.3935	1.1964	9.98×10^{-01}	9.98×10^{-01}	9.98×10^{-01}	2.5786	1.1964	4.5151
	Std	6.81×10^{-01}	0.00	7.58×10^{-01}	6.27×10^{-01}	0.000	0.00	7.92×10^{-11}	2.0922	6.27×10^{-01}	4.8435
	Median	9.98×10^{-01}	9.98×10^{-01}	7.98×10^{-01}	9.98×10^{-01}	9.98×10^{-01}	9.98×10^{-01}	9.98×10^{-01}	9.98×10^{-01}	9.98×10^{-01}	1.9901
	Max	2.9821	9.98×10^{-01}	1.9821	2.9821	9.98×10^{-01}	9.98×10^{-01}	9.98×10^{-01}	5.9288	2.9821	$1.26\times10^{+01}$
f_{15}	Mean	7.64×10^{-04}	3.83×10^{-04}	7.92×10^{-04}	6.47×10^{-04}	6.05×10^{-04}	3.08×10^{-04}	5.53×10^{-04}	9.68×10^{-04}	6.81×10^{-04}	2.31×10^{-03}
	Std	6.79×10^{-04}	7.75×10^{-05}	4.68×10^{-04}	4.88×10^{-04}	7.84×10^{-05}	2.38×10^{-04}	1.69×10^{-04}	3.49×10^{-04}	4.54×10^{-04}	6.34×10^{-03}
	Median	3.07×10^{-04}	3.07×10^{-03}	6.08×10^{-04}	3.67×10^{-04}	5.8×10^{-04}	3.07×10^{-04}	5.32×10^{-04}	7.83×10^{-04}	5.47×10^{-04}	3.07×10^{-04}
	Max	5.08×10^{-03}	1.06×10^{-03}	1.54×10^{-04}	1.59×10^{-03}	7.38×10^{-04}	3.12×10^{-04}	8.68×10^{-04}	1.48×10^{-03}	1.49×10^{-03}	2.03×10^{-02}
f_{16}	Mean	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.03	− 1.0316	− 1.0316	− 1.0316
	Std	6.3034×10^{-06}	0.00	1.25×10^{-05}	0.00	0.00	0.00	1.33×10^{-03}	0.00	1.32×10^{-12}	5.62×10^{-09}
	Median	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0303	− 1.0316	− 1.0316	− 1.0316
	Max	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.0316	− 1.027	− 1.0316	− 1.0316	− 1.0316
f_{17}	Mean	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}
	Std	0.00	0.00	4.1×10^{-04}	0.00	0.00	0.00	5.96×10^{-04}	0.00	5.53×10^{-08}	1.35×10^{-07}
	Median	3.97×10^{-01}	3.97×10^{-01}	3.98×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.98×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}
	Max	3.97×10^{-01}	3.97×10^{-01}	3.99×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.99×10^{-01}	3.97×10^{-01}	3.97×10^{-01}	3.97×10^{-01}
f_{18}	Mean	3.00	3.00	3.00	3.00	3.00	3.00	3.2643	3.00	3.00	3.00
	Std	0.00	0.00	1.91×10^{-06}	3.63×10^{-16}	6.10×10^{-16}	2.96×10^{-16}	1.87×10^{-01}	1.52×10^{-01}	4.09×10^{-06}	3.32×10^{-06}
	Median	3.00	3.00	3.00	3.00	3.00	3.00	3.2619	3.00	3.00	3.00
	Max	3.00	3.00	3.00	3.00	3.00	3.00	3.5859	3.00	3.00	3.00
f_{19}	Mean	− 3.8628	− 3.8628	− 3.8546	− 3.8628	− 3.8628	− 3.8628	− 3.845	− .8628	− 3.8614	− 3.862
	Std	0.00	0.00	2.91×10^{-04}	8.50×10^{-16}	9.36×10^{-16}	9.36×10^{-16}	1.43×10^{-02}	9.36×10^{-16}	2.09×10^{-03}	2.06×10^{-03}
	Median	− 3.8628	− 3.8628	− 3.8547	− 3.8628	− 3.8628	− 3.8628	− 3.8487	− 3.8628	− 3.8622	− 3.8628
	Max	− 3.8628	− 3.8628	− 3.8539	− 3.8628	− 3.8628	− 3.8628	− 3.8248	− 3.8628	− 3.8563	− 3.8563

Table 2 (continued)

f_n		DDMPEA	SCA-PSO	SCA	PSO	DE	TLBO	CSA	MFO	WOA[27]	GWO
f_{20}	Mean	− 3.2031	− 3.1266	− 3.0678	− 3.2982	− 3.322	− 3.2483	− 3.0003	− 3.2084	− 3.2499	− 3.2606
	Std	0.00	9.25×10^{-02}	6.04×10^{-02}	5.01×10^{-02}	1.39×10^{-03}	8.56×10^{-02}	1.25×10^{-01}	4.48×10^{-02}	8.05×10^{-02}	6.47×10^{-02}
	Median	− 3.2031	− 3.111	− 3.071	− 3.322	− 3.322	− 3.2623	− 3.0584	− 3.2031	− 3.2616	− 3.2625
	Max	− 3.2031	− 3.0124	− 2.9989	− 3.2031	− 3.3217	− 3.0779	− 2.7147	− 3.1376	− 3.1068	− 3.1974
f_{21}	Mean	− 9.24	− 10.1532	− 3.5548	− 5.8911	− 10.1532	− 10.1532	− 10.1532	− 4.8925	− 10.1528	− 9.6477
	Std	1.9944	0.00	2.4856	3.1247	3.08×10^{-15}	1.18×10^{-15}	1.53×10^{-05}	2.9835	4.7×10^{-04}	1.5977
	Median	− 10.1532	− 10.1532	− 4.5015	− 5.1008	− 10.1532	− 10.1532	− 10.1532	− 3.869	− 10.153	− 10.1529
	Max	− 5.06	− 10.1532	-4.96×10^{-01}	− 2.6829	− 10.1532	− 10.1532	− 10.1532	− 2.6305	− 10.1517	− 5.1005
f_{22}	Mean	$-1.01\times10^{+01}$	− 10.4029	− 3.7603	− 5.9614	− 10.4029	− 10.4029	− 104,028	− 8.4398	− 8.5758	− 10.4026
	Std	1.2874	1.18×10^{-15}	2.5092	3.9106	2.37×10^{-15}	1.32×10^{-15}	4.18×10^{-05}	3.2084	3.0082	1.82×10^{-04}
	Median	− 10.4029	− 10.4029	− 4.9814	− 3.9474	− 10.4029	− 10.4029	− 10.4028	− 10.4029	− 10.4022	− 10.4027
	Max	− 5.09	− 10.4029	− 0.90776	− 1.8376	− 10.4029	− 10.4029	− 10.4027	− 2.7659	− 2.7659	− 10.4023
f_{23}	Mean	$-1.02\times10^{+01}$	− 10.5364	− 4.6381	− 5.9094	− 10.5364	− 105,364	− 10.5363	− 7.0895	− 9.9942	− 10.5361
	Std	1.3099	1.78×10^{-15}	7.79×10^{-01}	3.3467	1.79×10^{-15}	2.13×10^{-15}	5.53×10^{-05}	3.6447	1.71	1.91×10^{-04}
	Median	− 0.5364	− 10.5364	− 4.8828	− 5.1521	− 10.5364	− 105,364	− 10.5363	− 7.4859	− 10.5356	− 10.5361
	Max	− 5.13	− 10.5364	− 2.9115	− 2.4273	− 10.5364	− 105,364	− 10.5361	− 2.8711	− 5.1274	− 10.5358

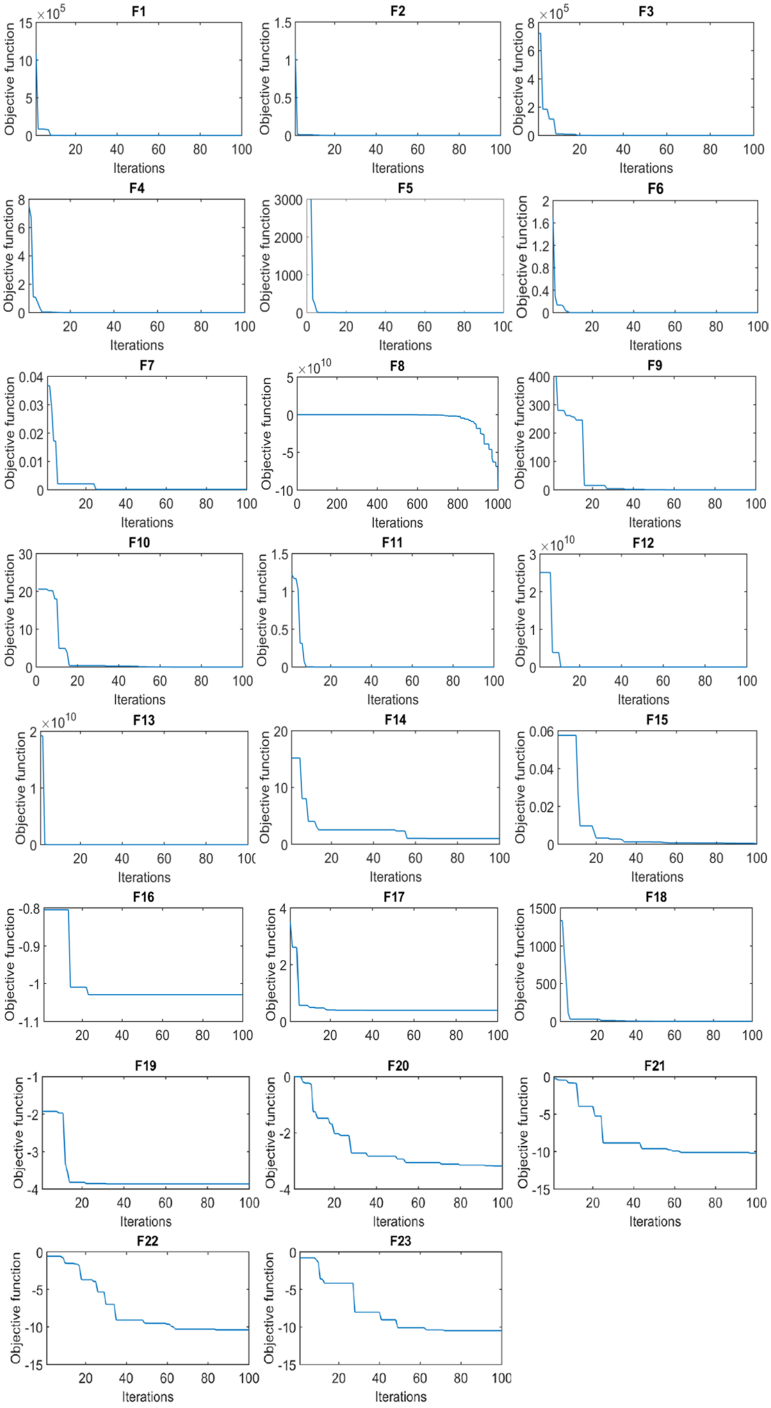

Fig. 5 Convergence behaviour for the results of DDMPEA with ANUM algorithm

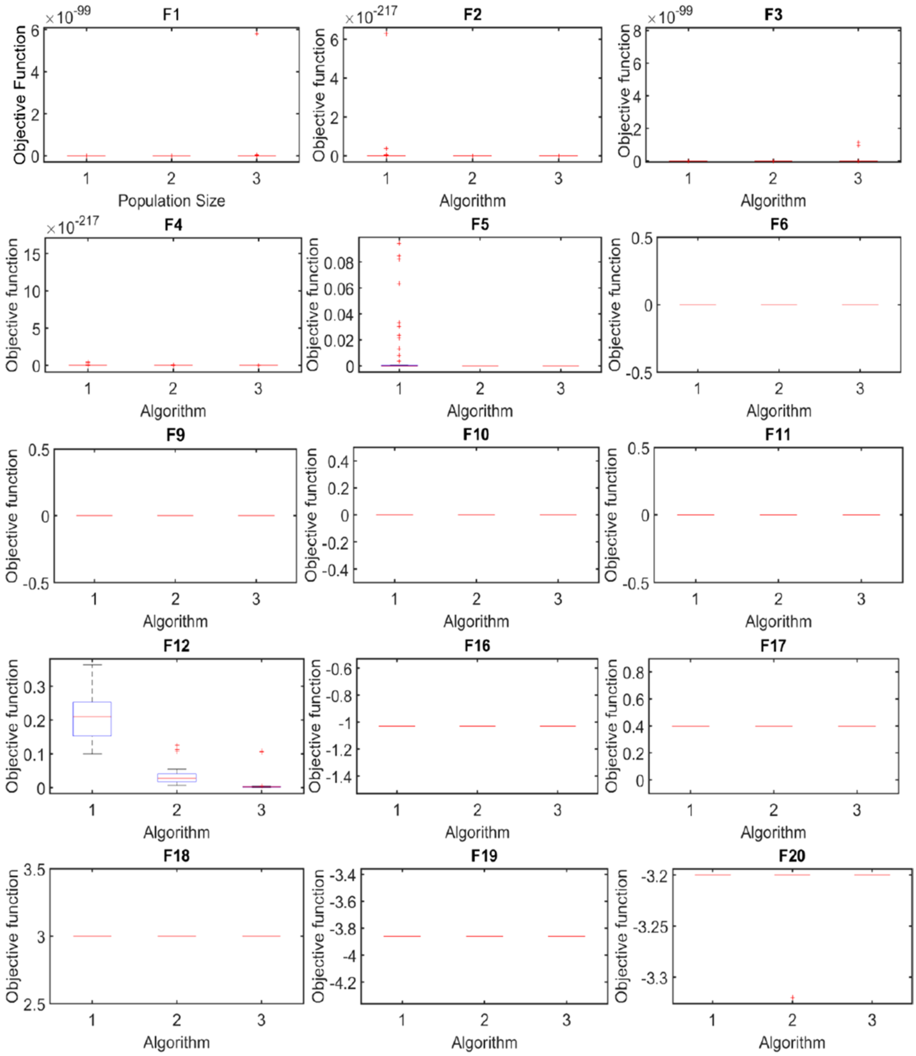

Fig. 6 Box plots of the results of proposed algorithm for function f_1–f_6, f_9–f_{12}, f_{16}–f_{20}

remaining functions, the proposed algorithm gives competitive results as compared to other optimization algorithms.

6.2 Heterogeneous Wireless Sensor Network

Validating the proposed algorithm on benchmark problems, the application of heterogeneous wireless sensor networks is investigated. The parameters considered for simulation of heterogeneous WSN are given in Table 4. The heterogeneous energy sensor nodes are deployed randomly in area 100 × 100 m^2, and the sink is located in the middle of the network.

Table 3 p-values calculated for the Wilcoxon rank-sum test (with significance level 0.05) for basic benchmark functions

f_n	DDMPEA	SCA-PSO[18]	SCA[18]	PSO[18]	DE[18]	TLBO[18]	CSA[18]	MFO[18]	WOA[18]	GWO[18]
f_1	6.39×10^{-05}	N/A	6.39×10^{-05}	$6.39\times10–^{05}$	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}
f_2	6.39×10^{-05}	N/A	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}
f_3	6.39×10^{-05}	N/A	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}
f_4	6.39×10^{-05}	N/A	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}
f_5	N/A	1.82×10^{-04}	1.83×10^{-04}	1.83×10^{-04}	2.46×10^{-04}	4.40×10^{-04}	N/A	1.83×10^{-04}	1.83×10^{-04}	1.83×10^{-04}
f_6	N/A	N/A	6.39×10^{-05}	6.39×10^{-05}	N/A	1.47×10^{-02}	6.39×10^{-05}	2.31×10^{-04}	6.39×10^{-05}	6.39×10^{-05}
f_7	7.69×10^{-04}	N/A	7.69×10^{-04}	1.83×10^{-04}	1.83×10^{-04}	1.83×10^{-04}	1.70×10^{-03}	1.83×10^{-04}	1.13×10^{-02}	1.70×10^{-03}
f_8	1.68×10^{-01}	1.83×10^{-04}	1.83×10^{-04}	1.83×10^{-04}	1.10×10^{-04}	1.82×10^{-04}	N/A	1.83×10^{-04}	1.83×10^{-04}	1.83×10^{-04}
f_9	N/A	N/A	1.68×10^{-01}	6.39×10^{-05}	N/A	6.25×10^{-05}	6.39×10^{-05}	6.34×10^{-05}	1.68×10^{-01}	1.68×10^{-01}
f_{10}	N/A	N/A	5.51×10^{-05}	6.39×10^{-05}	1.59×10^{-05}	1.59×10^{-05}	6.39×10^{-05}	3.32×10^{-05}	6.04×10^{-04}	5.51×10^{-05}
f_{11}	N/A	N/A	1.68×10^{-01}	6.39×10^{-05}	N/A	3.50×10^{-02}	6.39×10^{-05}	6.39×10^{-05}	N/A	1.68×10^{-01}
f_{12}	8.75×10^{-05}	8.75×10^{-05}	8.75×10^{-05}	8.75×10^{-05}	N/A	1.68×10^{-02}	8.75×10^{-05}	1.00×10^{-03}	8.75×10^{-05}	8.75×10^{-05}
f_{13}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	N/A	5.90×10^{-03}	6.39×10^{-05}	5.51×10^{-05}	6.39×10^{-05}	6.39×10^{-05}
f_{14}	N/A	N/A	6.34×10^{-05}	1.59×10^{-04}	N/A	N/A	N/A	5.76×10^{-05}	4.88×10^{-05}	5.98×10^{-05}
f_{15}	8.75×10^{-05}	N/A	8.75×10^{-05}	7.81×10^{-01}	8.75×10^{-05}	3.18×10^{-01}	8.75×10^{-05}	8.69×10^{-05}	1.30×10^{-03}	8.84×10^{-02}
f_{16}	N/A	N/A	1.40×10^{-03}	7.56×10^{-04}	9.26×10^{-04}	N/A	1.15×10^{-01}	N/A	7.56×10^{-04}	1.70×10^{-02}
f_{17}	N/A	N/A	1.72×10^{-02}	2.43×10^{-05}	N/A	N/A	6.39×10^{-05}	N/A	5.94×10^{-05}	6.39×10^{-05}
f_{18}	N/A	N/A	6.39×10^{-05}	1.59×10^{-05}	1.59×10^{-05}	1.59×10^{-05}	6.39×10^{-05}	3.68×10^{-01}	1.68×10^{-01}	1.68×10^{-01}
f_{19}	N/A	N/A	6.39×10^{-05}	1.59×10^{-05}	1.59×10^{-05}	1.59×10^{-05}	6.39×10^{-05}	1.59×10^{-05}	6.39×10^{-05}	6.39×10^{-05}
f_{20}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	6.39×10^{-05}	N/A	4.88×10^{-05}	6.39×10^{-05}	5.76×10^{-05}	6.39×10^{-05}	6.39×10^{-05}
f_{21}	N/A	N/A	8.75×10^{-05}	2.09×10^{-04}	1.33×10^{-04}	7.95×10^{-05}	7.57×10^{-04}	1.54×10^{-04}	8.75×10^{-05}	1.21×10^{-04}
f_{22}	N/A	N/A	1.29×10^{-04}	1.30×10^{-01}	1.90×10^{-03}	1.03×10^{-04}	2.75×10^{-04}	9.30×10^{-03}	1.29×10^{-04}	1.29×10^{-04}
f_{23}	N/A	N/A	1.11×10^{-04}	4.55×10^{-01}	1.93×10^{-02}	4.50×10^{-01}	1.20×10^{-03}	1.70×10^{-02}	2.07×10^{-04}	2.07×10^{-04}

Table 4 Parameters considered for Heterogeneous wireless sensor network

The area considered for network	100 × 100 m^2
Number of sensor nodes	100
The initial energy of nodes (Joules) (E_0)	0.5
Heterogeneity in terms of energy for nodes	Normal, intermediate and advanced nodes
Energy fraction of intermediate nodes (β) and advanced nodes (α)	$\beta = 1, \alpha = 2$
Number of intermediate (m) and advanced nodes (m$_0$)	$m = 0.1, m_0 = 0.2$
The energy required for running transmitter and receiver E_{elec}	50 nJ/bit
Threshold distance (d$_0$)	87 m
Amplification energy required for smaller distance $d \leq d_0 \left(E_{efs}\right)$	10 pJ/bit/m^2
Amplification energy required for larger distance $d > d_0 \left(E_{mp}\right)$	0.0013 pJ/bit/m4
Energy consumption incurred while data aggregation (E_da)	5 nJ/bit/signal
Data packet size	2000 bits

The 0.5 J is taken as initial energy of normal nodes, and the intermediate nodes have energy two times of normal nodes, while advanced nodes have three times more energy of normal nodes. The number of intermediate nodes is ten percent of total sensor nodes in the considered network, and the number of advanced nodes is 20 percent of total nodes, and energy fraction is given in Table 4. The assumptions considered for the proposed protocol.

A few assumptions are considered about the qualities for a portion of the sensor nodes enlisted bellow.

- In the network, all sensor nodes are stationary after the deployment, including the sink. They have enough capacity for forwarding and receiving data packets from other nodes and sink, covering their sensing range. Also, the coverage area is taken in meter square.
- The deployed nodes are heterogeneous in terms of their initial energy, i.e., some of the nodes have more energy resources compared to other nodes. Hence three types of nodes are considered advanced, intermediate, and normal nodes.
- The sink is viewed as having an infinite energy source.
- Once the sensor nodes are depleting their energies, they cannot be recharged again. Also, the sensor nodes are not aware of their location.
- The different components, for example, signal blurring, happen during transmission and gathering, impedance, and sign misfortune because of different natural elements, and physical deterrents are not thought of.

6.2.1 Performance Measures

The analysis of the proposed protocol is carried out using measures such as network lifetime, residual energy of the network, stability period, and the number of data packets transmitted to the sink (network throughput).

The lifetime of a network The number of alive nodes in the network represents the lifetime of the network, i.e. the active nodes having sufficient energy for transmission of data packets.

residual energy of the network As the data transmission takes place in the wireless sensor network, the sum of residual energy of sensor nodes decreases gradually due to the fact that nodes squander their energy during the data transmission with other sensor nodes and the sink. This performance measure is obtained by adding residual energy of all nodes after each round of data transmission.

Stability Period The stability period of WSN is defined in terms of numbers of the round until the first sensor node consumes its total energy in the process of data transmission. This first dead sensor node can be advanced, intermediate, or normal.

Network throughput The number of data packets satisfyingly transmitted over time is termed as network throughput. This performance metric plays a vital role as it ensures the reliability of the network to obtain the best network performance.

6.2.2 Simulation Analysis

The simulation analysis of the proposed protocol is carried out on the basis of performance measures discussed in Sect. 6.2.1. The working process is described in Sect. 5.5 for a heterogeneous wireless sensor network. The obtained results of the proposed protocol are compared with other optimization algorithms based protocols such as GAOC, GADA_LEACH, TEDRP, and DCH_GA from genetic algorithm based paper [55].

Network Lifetime From Fig. 7., it is evident that in the proposed protocol, sensor nodes die after 21,000 rounds whereas GAOC, GADA-LEACH, and DCH-GA cover 19,648, 17,113, 18,498 and 14,729 rounds respectively. The improvement in network lifetime by proposed protocols as compared to other protocols, in terms of percentage is 6.88, 22.71, 13.52, and 42.57% compared to GAOC, GADA-LEACH, TEDRP, and DCH-GA respectively. The alive nodes vs. rounds given in Fig. 8.

The dead nodes vs. rounds given in Fig. 9. In the proposed work, half sensor node dead (HND) after 10,627 rounds of data transmission in the wireless sensor network. The HND for GAOC, GADA-LEACH, TEDRP, and DCH-GA are 10,674, 7722, 9086, and 7111, respectively.

The residual energy of the network The occurrence of the data transmission process in heterogeneous wireless sensor networks costs the energy loss of sensor nodes. The

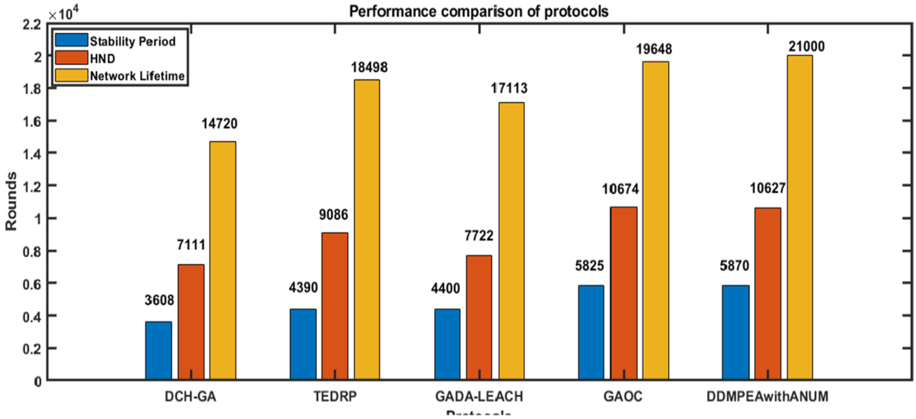

Fig. 7 Comparison of DDMPEA with ANUM with other protocols in terms of stability period, HND and network lifetime

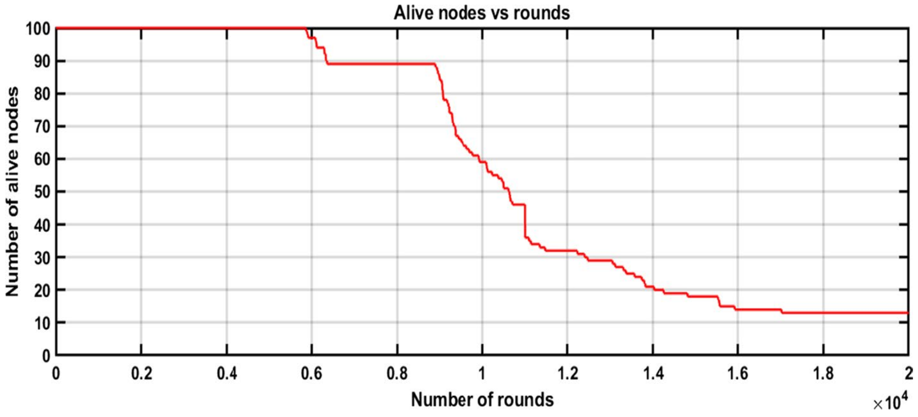

Fig. 8 Alive nodes vs. rounds for proposed algorithm

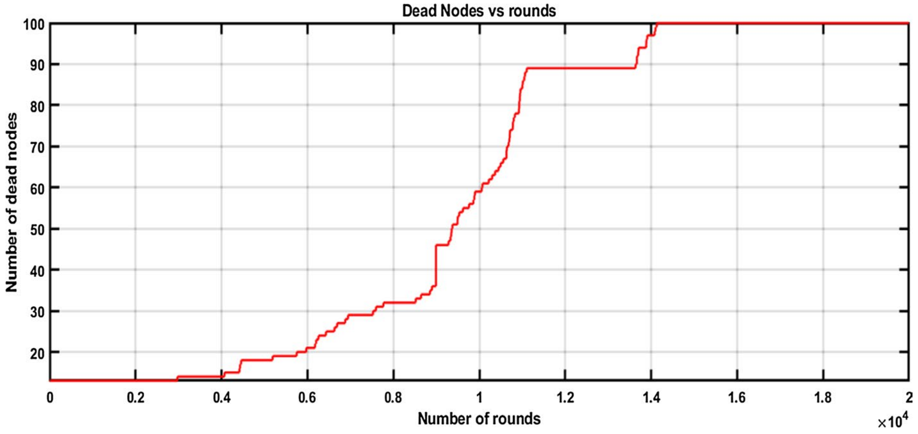

Fig. 9 Dead nodes vs rounds for proposed protocol

remaining energy in WSN is observed after each round of data transmission. Figure 10. presents energy consumption per round in the data communication process. The figure is given for 100 sensor nodes, which are used to cover the 100 × 100 m^2 area. The total energy for this configuration is 70 J.

Stability period From Figs. 7. and 8., it is clear that the proposed algorithm shows improvement in the stability period of WSN in percentage as 0.77, 33.41, 33.71 and 62.69% as compared to GAOC, GADA-LEACH, TEDRP and DCH-GA respectively. This improvement ensures reliable data transmission in the network.

Network throughput This parameter is related to the number of data packets transmitted successfully to the base station. The proposed algorithm transmits 1,680,000 data bits to the sink from the cluster heads. This performance metric plays a significant role in acquiring a higher network lifespan.

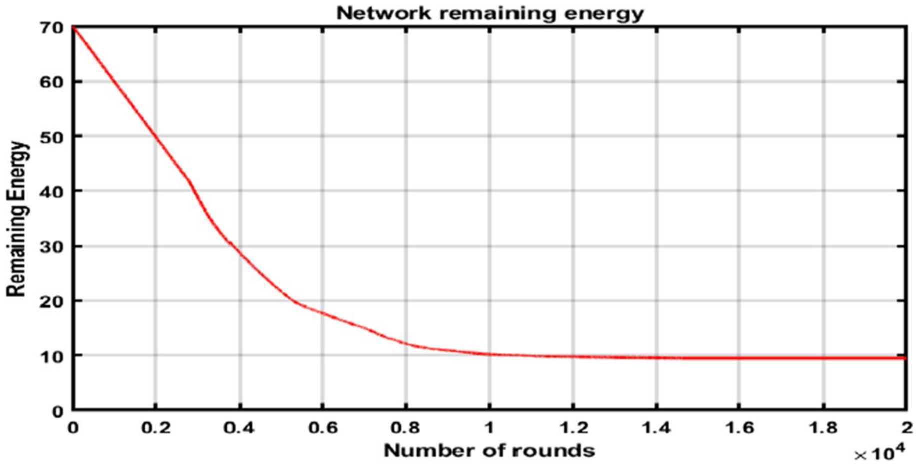

Fig. 10 Residual energy of heterogeneous WSN for 100 sensor nodes

6.2.3 Summarized Analysis of Proposed Protocol

From the obtained results, it is clear that the proposed method not only enhances the reliability but also improves the stability period of the network. The comparative analysis and percentage improvement in performance metrics are shown in Tables 5 and 6.

7 Conclusion

In a given study, a new technique named 'Diversity driven multi-parent evolutionary algorithm with adaptive non-uniform mutation' has been suggested for cluster head selection in heterogeneous wireless sensor networks. Residual energy and distance travelled are two objective functions that are to be optimized simultaneously in order to minimize the overall fitness function. Fuzzy set theory is utilized to evaluate membership function for both objective functions, i.e. residual energy and distance travelled. The best value of a membership function is selected for cluster head selection based on higher cardinal ranking.

Table 5 Comparison of proposed protocol with other protocols in terms of performance metrics

Value of advanced fractions and quantity fractions of the node $\left(m = 0.1, m_0 = 0.2, \beta = 1, \alpha = 2\right)$

Protocols	No. of data sink	Stability Period of HWSN (rounds)	Number of Half node dead (rounds)	Network lifetime of HWSN (rounds)
DDMPEA with ANUM-OC	1	5870	10,627	21,000
GAOC	1	5825	10,674	19,648
GADA-LEACH	1	4400	7722	17,113
TEDRP	1	4390	9086	18,498
DCH-GA	1	3608	7111	14,720

Table 6 Comparison in terms of percentage improvement

Percentage (%) improvement by DDMPEA with ANUM-OC protocol			
Protocols	Stability Period of HWSN	Number of Half node dead	Network lifetime of HWSN
GAOC	0.77	0.00	6.88
DCH-GA	62.91	49.44	42.57
TEDRP	33.71	16.96	13.52
GADA-LEACH	33.41	37.62	22.71

The clustering of nodes is achieved utilizing fuzzy theory based on the remaining energy of nodes and distance between sensor nodes and base station. The proposed algorithm proved to be beneficial as it shows the improvement in the stability period, alive nodes, and network lifetime in comparison with other optimization-based methods. The percentage improvement in the stability period is 0.77, 62.91, 33.71, and 33.41 percentage as compared to GAOC, DCH-GA, TEDRP, and GADA-LEACH, respectively. In the same manner, the percentage improvement in network lifetime is 6.88, 42.51, 13.52, and 22.71% as compared to these protocols. The efficacy of the proposed algorithm is also validated on benchmark functions, and improvement is observed in terms of mean, worst, median, best, and standard deviation.

For future work, we are going to investigate the following issues: First, the proposed DDMPEA with ANUM can be applied to other challenging real-world problems like signal processing and fault diagnosis. Second, it could be interesting to incorporate DDMPEA with ANUM with some effective mechanisms of other metaheuristics, such as SCA, and SSA.

References

1. Potthuri, S., Shankar, T., & Rajesh, A. (2018). Lifetime improvement in wireless sensor networks using hybrid differential evolution and simulated annealing (DESA). *Ain Shams Engineering Journal, 9*(4), 655–663. https://doi.org/10.1016/j.asej.2016.03.004.
2. John, J., & Rodrigues, P. (2019). MOTCO: Multi-objective Taylor crow optimization algorithm for cluster head selection in energy aware wireless sensor network. *Mobile Networks and Applications, 24*(5), 1509–1525. https://doi.org/10.1007/s11036-019-01271-1.
3. Kumar, D. (2013). Performance analysis of energy efficient clustering protocols for maximising lifetime of wireless sensor networks. *IET Wireless Sensor System, 4*(1), 9–16. https://doi.org/10.1049/iet-wss.2012.0150.
4. Simon, G., et al. (2004). Sensor network-based countersniper system. In *Proceedings of second international conference embeded networked sensor systems (Sensys)*, Balt. MD.
5. Yick, J., Mukherjee, B., & Ghosal, D. (2005). Analysis of a prediction-based mobility adaptive tracking algorithm. In *2nd international conference broadband networks, BROADNETS* (vol. 2005, pp. 809–816). https://doi.org/10.1109/ICBN.2005.1589681.
6. Castillo-Effen, M., Quintela, D. H., Jordan, R., Westhoff, W., & Moreno, W. (2004). Wireless sensor networks for flash-flood alerting. In *Proceedings of IEEE international caracas conference devices, circuits system ICCDCS* (pp. 142–146). https://doi.org/10.1109/iccdcs.2004.1393370.
7. Gao, T., Greenspan, D., Welsh, M., Juang, R. R., & Alm, A. (2005). Vital signs monitoring and patient tracking over a wireless network. In *Annual international conference ieee engineering in medicine and biology proceedings* (vol. 7, pp. 102–105).
8. Lorincz, K., et al. (2004). Sensor networks for emergency response: Challenges and opportunities. In *IEEE pervasive computing pervasive computing first response (Special Issue).*

9. Werner-Allen, G., et al. (2006). Deploying a wireless sensor network on an active volcano. *IEEE Internet Computing, 10*(2), 18–25. https://doi.org/10.1109/MIC.2006.26.
10. Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. *Computer Networks, 52*(12), 2292–2330. https://doi.org/10.1016/j.comnet.2008.04.002.
11. Bagci, F. (xxxx). Energy-efficient communication protocol for wireless microsensor networks. In *Proceeding of 33rd Hawai international conference system science*
12. Shepard, T. J. (xxxx). A Channel access scheme for large dense packet radio networks. In *Proceeding of ACM SIGCOMM* (pp. 219–230). https://doi.org/10.1145/248157.248176.
13. Li, S., Chen, H., Wang, M., Heidari, A. A., & Mirjalili, S. (2020). Slime mould algorithm: A new method for Stochastic optimization. *Future Generation Computer Systems, 2,* 13.
14. Mirjalili, S., Gandomi, A. H., Zahra, S., & Saremi, S. (2017). Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. *Advances in Engineering Software, 114,* 163–191. https://doi.org/10.1016/j.advengsoft.2017.07.002.
15. Faris, H., Mirjalili, S., Aljarah, I., Mafarja, M., & Heidari, A. A. (2020). Salp Swarm algorithm: Theory, literature review, and application in extreme learning machines. *Studies in Computational Intelligence, 811,* 185–199. https://doi.org/10.1007/978-3-030-12127-3_11.
16. Wu, J., Nan, R., & Chen, L. (2019). Improved salp swarm algorithm based on weight factor and adaptive mutation. *Journal of Experimental and Theoretical Artificial Intelligence, 00*(00), 1–23. https://doi.org/10.1080/0952813X.2019.1572659.
17. Mirjalili, S. (2016). SCA: A Sine Cosine Algorithm for solving optimization problems. *Knowledge-Based System, 96,* 120–133. https://doi.org/10.1016/j.knosys.2015.12.022.
18. Nenavath, H., Kumar, R., & Das, S. (2018). A synergy of the sine-cosine algorithm and particle swarm optimizer for improved global optimization and object tracking. *Swarm and Evolutionary Computation, 43,* 1–30. https://doi.org/10.1016/j.swevo.2018.02.011.
19. Gupta, S., Deep, K., Mirjalili, S., & Hoon, J. (2020). A modified sine cosine algorithm with novel transition parameter and mutation operator for global optimization. *Expert Systems with Applications, 2020,* 113395.
20. Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawks optimization: Algorithm and applications. *Future Generation Computer Systems*. https://doi.org/10.1016/j.future.2019.02.028.
21. Kamboj, V. K., Nandi, A., Bhadoria, A., & Sehgal, S. (2020). An intensify Harris Hawks optimizer for numerical and engineering optimization problems. *Applied Soft Computing, 89,* 106018. https://doi.org/10.1016/j.asoc.2019.106018.
22. Mirjalili, S., Mohammad, S., & Lewis, A. (2014). Grey Wolf Optimizer. *Advances in Engineering Software, 69,* 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007.
23. Miao, Z., Yuan, X., Zhou, F., Qiu, X., Song, Y., & Chen, K. (2020). Grey wolf optimizer with an enhanced hierarchy and its application to the wireless sensor network coverage optimization problem. *Applied Soft Computing Journal, 96*(2020), 106602. https://doi.org/10.1016/j.asoc.2020.106602.
24. Al-betar, M. A., Awadallah, M. A., Faris, H., Aljarah, I., & Hammouri, A. I. (2018). Natural selection methods for Grey Wolf Optimizer. *Expert Systems with Applications, 113,* 481–498. https://doi.org/10.1016/j.eswa.2018.07.022.
25. Golzari, S., Zardehsavar, M. N., Mousavi, A., Saybani, M. R., Khalili, A., & Shamshirband, S. (2018). KGSA: A gravitational search algorithm for multimodal optimization based on k-means niching technique and a novel elitism strategy. *Open Mathematics, 16*(1), 1582–1606. https://doi.org/10.1515/math-2018-0132.
26. Mirjalili, S., Mirjalili, S. M., & Hatamlou, A. (2016). Multi-verse optimizer: A nature-inspired algorithm for global optimization. *Neural Computing and Applications, 27*(2), 495–513. https://doi.org/10.1007/s00521-015-1870-7.
27. Mirjalili, S., & Lewis, A. (2016). The Whale optimization algorithm. *Advances in Engineering Software, 95,* 51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008.
28. Jadhav, A. N., & Gomathi, N. (2018). WGC: Hybridization of exponential grey Wolf optimizer with whale optimization for data clustering. *Alexandria Engineering Journal, 57*(3), 1569–1584. https://doi.org/10.1016/j.aej.2017.04.013.
29. Zhou, W., Zhou, P., Wang, Y., & Wang, N. (2019). Station-keeping control of an underactuated stratospheric airship. *International Journal of Fuzzy Systems, 21*(3), 715–732. https://doi.org/10.1007/s40815-018-0566-4.
30. Singh, M., & Dhillon, J. S. (2016). Multiobjective thermal power dispatch using opposition-based greedy heuristic search. *International Journal of Electrical Power and Energy Systems, 82,* 339–353. https://doi.org/10.1016/j.ijepes.2016.03.016.

31. Yassein, L. (2009). Improvement on LEACH protocol of wireless sensor network (VLEACH). *International Journal of Digital Content Technology and its Applications, 3*(2), 132–136. https://doi.org/10.4156/jdcta.vol3.issue2.yassein.
32. Mu, T., & Tang, M. (2010). LEACH-B: An improved LEACH protocol for wireless sensor network. In *2010 6th international conference wireless communication network mobile computing WiCOM 2010* (pp. 2–5). https://doi.org/10.1109/WICOM.2010.5601113.
33. Younis, O., & Fahmy, S. (2004). HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing, 3*(4), 366–379. https://doi.org/10.1109/TMC.2004.41.
34. Mirzaie, M., & Mazinani, S. M. (2017). Adaptive MCFL: An adaptive multi-clustering algorithm using fuzzy logic in wireless sensor network. *Computer Communications, 111,* 56–67. https://doi.org/10.1016/j.comcom.2017.07.005.
35. Mahajan, S., Malhotra, J., & Sharma, S. (2014). An energy balanced QoS based cluster head selection strategy for WSN. *Egyptian Informatics Journal, 15*(3), 189–199. https://doi.org/10.1016/j.eij.2014.09.001.
36. Shankar, T. (xxxx). *Whale optimization based energy-efficient cluster head selection algorithm for wireless sensor networks* (pp. 1–22).
37. Guo, L., Li, Q., & Chen, F. (2011). A novel cluster-head selection algorithm based on hybrid Genetic Optimization for wireless sensor networks. *Journal Networks, 6*(5), 815–822. https://doi.org/10.4304/jnw.6.5.815-822.
38. Kuila, P., & Jana, P. K. (2014). Energy efficient clustering and routing algorithms for wireless sensor networks: Particle swarm optimization approach. *Engineering Applications of Artificial Intelligence, 33,* 127–140. https://doi.org/10.1016/j.engappai.2014.04.009.
39. Zeng, B., & Dong, Y. (2016). An improved harmony search based energy-efficient routing algorithm for wireless sensor networks. *Applied Soft Computing Journal, 41,* 135–147. https://doi.org/10.1016/j.asoc.2015.12.028.
40. Karaboga, D., Okdem, S., & Ozturk, C. (2012). Cluster based wireless sensor network routing using artificial bee colony algorithm. *Wireless Networks, 18*(7), 847–860. https://doi.org/10.1007/s11276-012-0438-z.
41. Chen, R. C., Chang, W. L., Shieh, C. F., & Zou, C. C. (2012). Using hybrid artificial bee colony algorithm to extend wireless sensor network lifetime. In *Proceeding 3rd international conference innovation bio-inspired computing application IBICA* (pp. 156–161). https://doi.org/10.1109/IBICA.2012.27.
42. Kumar, R., & Kumar, D. (2016). Multi-objective fractional artificial bee colony algorithm to energy aware routing protocol in wireless sensor network. *Wireless Networks, 22*(5), 1461–1474. https://doi.org/10.1007/s11276-015-1039-4.
43. Vinodhini, R., & Gomathy, C. (2020). MOMHR: A dynamic multi-hop routing protocol for WSN using Heuristic based multi-objective function. *Wireless Personal Communication, 111*(2), 883–907. https://doi.org/10.1007/s11277-019-06891-0.
44. Ghugar, U., Pradhan, J., Bhoi, S. K., & Sahoo, R. R. (2019). LB-IDS: Securing wireless sensor network using protocol layer trust-based intrusion detection system. *Journal Computing Networks Communication., 5,* 71.
45. Ghugar, U., Pradhan, J., & Kumar, S. (2018). PL-IDS: physical layer trust based intrusion detection system for wireless sensor networks. *International Journal Information Technology, 10*(4), 489–494. https://doi.org/10.1007/s41870-018-0147-7.
46. Ranjan, R., Sudhabindu, S., Souvik, R., Sourav, S., & Bhoi, K. (2018). Guard against trust management vulnerabilities in Wireless Sensor Network. *Arabian Journal for Science and Engineering, 43*(12), 7229–7251. https://doi.org/10.1007/s13369-017-3052-7.
47. Bhoi, S. K., Panda, S. K., & Khilar, P. M. (2013). A density-based clustering paradigm to detect faults in wireless sensor networks. In *International conference on advances in computing* (pp. 865–871).
48. Bhoi, S. K., Obaidat, M. S., Puthal, D., Singh, M., & Hsiao, K.-F. (2018). Software defined network based fault detection in industrial wireless sensor networks. In *IEEE global communication conference (GLOBECOM)* (pp. 1–6).
49. Singh, M., Bhoi, S. K., & Khilar, P. M. (2017). Geometric constraint-based range-free localization scheme for wireless sensor networks. *IEEE Sensors Journal, 17*(16), 5350–5366.
50. Swain, R. R., Khilar, P. M., & Bhoi, S. K. (2018). Heterogeneous fault diagnosis for wireless sensor networks. *Ad Hoc Networks, 69,* 15–37. https://doi.org/10.1016/j.adhoc.2017.10.012.
51. Chauhan, S., Singh, M., & Aggarwal, A. K. (2020). Diversity driven multi-parent evolutionary algorithm with adaptive non-uniform mutation. *Journal of Experimental and Theoretical Artificial Intelligence, 2020,* 1–32.

52. Ali, M. Z., Awad, N. H., Suganthan, P. N., Shatnawi, A. M., & Reynolds, R. G. (2018). An improved class of real-coded Genetic Algorithms for numerical optimization✰. *Neurocomputing, 275,* 155–166. https://doi.org/10.1016/j.neucom.2017.05.054.
53. Wang, H., Wang, W., & Wu, Z. (2013). Particle Swarm optimization with adaptive mutation for multimodal optimization. *Applied Mathematics and Computation, 221,* 296–305. https://doi.org/10.1016/j.amc.2013.06.074.
54. Jun, T., & Xiaojuan, Z. (2009). Particle swarm optimization with adaptive mutation. In *2009 WASE international conference information engineering ICIE 2009* (Vol. 2, No. 1, pp. 234–237). https://doi.org/10.1109/ICIE.2009.59.
55. Verma, S., Sood, N., & Sharma, A. K. (2019). Genetic algorithm-based Optimized Cluster Head selection for single and multiple data sinks in Heterogeneous Wireless Sensor Network. *Applied Soft Computing Journal, 85,* 105788. https://doi.org/10.1016/j.asoc.2019.105788.
56. Dhillon, J. S., Parti, S. C., & Kothari, D. P. (2001). Fuzzy decision making in multiobjective long-term scheduling of hydrothermal system.
57. Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. In *Proceeding of sixth international symposium micro machine human science IEEE* (pp. 39–43). https://doi.org/10.1109/mhs.1995.494215.
58. Rao, R. V., Savsani, V. J., & Vakharia, D. P. (2011). Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems. *Computing Design, 43*(3), 303–315. https://doi.org/10.1016/j.cad.2010.12.015.
59. Mirjalili, S. (2015). Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-Based System, 89,* 228–249. https://doi.org/10.1016/j.knosys.2015.07.006.
60. Storn, R. (1997) Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces. In *Technical report, international computing science institution* (Vol. 11).
61. Askarzadeh, A. (2016). A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm. *Computers and Structures, 169,* 1–12. https://doi.org/10.1016/j.compstruc.2016.03.001.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Sumika Chauhan received her B.tech. (electronics and communication) from Guru Nanak Education trust, Group of Institutions, Roorkee; her M.tech (instrumentation and control) from from Sant Longowal Institute of Engineering and Technology, Longowal. She is currently pursuing her Ph.D. from Sant Longowal Institute of Engineering and Technology, Longowal.

Manmohan Singh received his B.E. (electrical) from Guru Nanak Dev Engineering College, Ludhiana (GNDEC); his M.E. (power apparatus and electrical drives) from University of Roorkee, Roorkee; and his Ph.D. from Sant Longowal Institute of Engineering and Technology, Longowal. He has been an assistant professor since May 1999 at the Electrical and Instrumentation Engineering Department, Sant Longowal Institute of Engineering and Technology, Longowal. Presently, he is an associate professor in the Electrical and Instrumentation Engineering Department, Sant Longowal Institute of Engineering and Technology, Longowal. His research activities include microprocessor applications, multi-objective thermal dispatch, hydrothermal scheduling, optimization, neural networks, fuzzy theory, and soft computing applications in power systems.

Ashwani Kumar Aggarwal Aggarwal received his B.E. (electronics and instrumentation) from College of Engineering and Technology, Bathinda; his M.Tech (instrument technology) from Indian Institute of Science, Bangalore (IISC); and his Ph.D. from The University of Tokyo, Japan. He has been an Associate professor since December 1996 at the Electrical and Instrumentation Engineering Department, Sant Longowal Institute of Engineering and Technology, Longowal. His research activities include image processing, signal processing and machine vision.