

# Red Deer and Simulation Annealing Optimization Algorithm-based Energy Efficient Clustering Protocol for Improved lifetime expectancy in Wireless Sensor Networks (WSNs)

Rajeswarappa Govardanagiri (✉ [rajeswarappag@gmail.com](mailto:rajeswarappag@gmail.com))

JNTUA CEA: Jawaharlal Nehru Technological University Anantapur College of Engineering  
Ananthapuramu

Vasundra S

JNTUA: Jawaharlal Nehru Technological University Anantapur

---

## Research Article

**Keywords:** Red Deer Algorithm (RDA), Simulated Annealing (SA), Network Lifetime Expectancy, Cluster Head Selection, Energy Efficiency.

**Posted Date:** July 23rd, 2021

**DOI:** <https://doi.org/10.21203/rs.3.rs-412476/v1>

**License:** © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

---

**Version of Record:** A version of this preprint was published at Wireless Personal Communications on August 16th, 2021. See the published version at <https://doi.org/10.1007/s11277-021-08808-2>.

## **Title Page**

**Title of the manuscript:** Red Deer and Simulation Annealing Optimization Algorithm-based Energy Efficient Clustering Protocol for improved lifetime expectancy in Wireless Sensor Networks (WSNs)

**First and Corresponding Author:** G. Rajeswarappa  
Research Scholar, Dept of CSE, JNTUA, Ananthapuramu,  
India.  
E-mail: [rajeswarappag@gmail.com](mailto:rajeswarappag@gmail.com)

**Second Author :** S. Vasundra  
Professor, Dept of CSE, JNTUACEA, Ananthapuramu, India,  
E-mail: [vasundras.cse@jntua.ac.in](mailto:vasundras.cse@jntua.ac.in).

# Red Deer and Simulation Annealing Optimization Algorithm-based Energy Efficient Clustering Protocol for improved lifetime expectancy in Wireless Sensor Networks (WSNs)

G. Rajeswarappa<sup>1,\*</sup>, S. Vasundra<sup>2</sup>

<sup>1,\*</sup> Research Scholar, Dept of CSE, JNTUA, Ananthapuramu, India,  
[rajeswarappag@gmail.com](mailto:rajeswarappag@gmail.com).

<sup>2</sup> Dept of CSE, JNTUACEA, Ananthapuramu, India,  
[vasundras.cse@jntua.ac.in](mailto:vasundras.cse@jntua.ac.in).

**Abstract:** Energy stability is considered as the core challenge of Wireless Sensor Networks (WSNs) as it is the main factor that contributes towards improved network lifetime expectancy. Clustering with maximized energy efficiency is determined to be the well documented NP hard optimization solution that has the possibility of resulting in prolonged network lifetime. Diversified number of computational approaches that includes nature inspired metaheuristic algorithms, reinforcement schemes and evolutionary algorithms were used for attaining efficient clustering process in WSNs. In this paper, Red Deer and Simulation Annealing Optimization Algorithm-based Energy Efficient Clustering Protocol (RDSAOA-EECP) is proposed for improving lifetime expectancy and energy stability in WSNs. This clustering protocol is proposed with the global optimization capability of the red deer optimization algorithm and local optimization potential of simulation annealing. In specific, RDA is used for diversification and Simulated Annealing (SA) for intensification in order to establish balance between them during the clustering process in order to sustain energy stability. This RDSAOA-EECP protocol is proposed with SA as the search tool that prevents the phases of fighting and roaring phases of RDA involved in the intensification process. Moreover, this clustering protocol inherits the potentialities of the proposed metaheuristic properties in order to achieve optimal cluster heads together with the optimal base station location for the objective of enhancing energy efficiency. The simulation results of the proposed RDSAOA-EECP protocol guaranteed its potentiality over the baseline clustering approaches in improving network lifetime, throughput, packet deliver rate with minimized network latency and energy consumptions independent to the homogeneous and heterogeneous sensor network setup considered for implementation.

**Keywords:** Red Deer Algorithm (RDA), Simulated Annealing (SA), Network Lifetime Expectancy, Cluster Head Selection, Energy Efficiency.

## 1. Introduction

Wireless sensor networks (WSNs) comprise of low cost and low-power sensor nodes [1]. The sensor nodes are positioned in a particular region for the purpose of organizing themselves for implementing a WSN through which decision making for a specific application can be achieved [2]. WSNs offers a variety of applications in a diversified number of domains such as health care monitoring, industrial observation units, battlefield surveillance, etc., that necessitates potential communication [3]. The overall control of WSNs is attained based on the energy possessed by the

sensor nodes, their computational potentialities and bandwidth utilized by them[4]. Thus, WSNs possess systematized accurate sensing capability of sensor nodes and achieves significant communication between the numerous nodes deployed in the implementation environment [5]. WSNs includes several merits such as lower power consumption, lower cost, flexibility in deployment, flexibility in communication and maximized cooperation between the sensor nodes deployed in the network [6]. WSNs comprises of numerous sensor nodes that generally runs on a non rechargeable battery. Hence, facilitating the objective of network connectivity, increased network lifetime, load balancing and fault tolerance necessitates the phenomenon of nodes grouping [7].

In this context, sensor nodes clustering is the process of partitioning them into small groups and selecting a group leader from all the groups based on the objective parameters such as residual energy, inter-distance between the cluster head and the cluster members and the distance between the cluster head and the base station, inter-cluster communication cost and so forth [8]. The problem of cluster head selection is addressed in the literature based on a number of approaches that are classified into normal and optimal cluster head selection strategies [9]. In specific, the optimal cluster head selection scheme is still classified into heuristic and meta-heuristic methods. These strategies are developed for providing solutions to optimization problems generally based on empirical nature with simple and compact theories [10]. Different heuristic algorithms were contributed in the literature for resolving the issues of heterogeneity [11]. However, the heuristic approaches inherited the limitations of numerical inefficiency during the process of search in the problems that involves large scale and high-dimensional factor that need to be optimized [12]. On the other hand, the meta-heuristic optimization approaches are considered to be highly suitable for solving high-dimensional complex optimization problems [13]. It is considered to be potential in bringing about an improved evolution in the region of the search. Moreover, the meta-heuristic methods of cluster head selection are identified to possess the characteristics of derivation free mechanism, prevention of local optimal point, simplicity and flexibility [14]. These meta-heuristic clustering approaches are classified into two categories, viz, i) population-based and ii) single solution-based solutions. It can also be classified into physics-based, evolutionary-based and bio-inspired, depending on the nature inspired characteristics inherent with them. But, the meta-heuristic clustering approaches independent to their dissimilarities are implemented in two stages namely, exploitation and exploration [15]. The stage of exploration is responsible for extensive investigation of the vital regions of the complete search space. Then, the exploitation process completely focuses on the ability of performing potential local searching around the potential regions that are determined by the phase of exploration. However, achieving reliable tradeoff between the steps of exploration and exploitation is considered to be a crucial challenge due to the stochastic properties of meta-heuristic algorithms. A diversified number of cluster head selection schemes were proposed in the literature over the decades using the merits of meta-heuristic approaches contributed by significant researchers associated with the problem optimization domain. But, the aspects of lifetime enhancement and energy stabilization necessitates further improvement in the process of cluster head selection. Moreover, the hybridization of Red deer

Optimization Algorithm and Simulated Annealing [16] motivated the possibility of utilizing it in the process of CH selection in order to increase network lifetime and maintain energy stability. .

In this paper, Red Deer and Simulation Annealing Optimization Algorithm-based Energy Efficient Clustering Protocol (RDSAOA-EECP) is proposed for improving the lifetime expectancy and energy stability during cluster head selection. In particular, Red Deer Optimization Algorithm (RDOA) was utilized for exploration process and Simulated Annealing (SA) used for exploitation process. This integration of RDOA and SA is mainly for attaining and maintaining potential balance between the phases of exploration and exploitation that is necessitated during the optimization process of cluster head selection. In specific, the important phases of roaring and fighting included in RDOA referred as the focus phase is eliminate through the folding and reheating phase of SA in order to facilitate better balance between the degree of exploitation and exploration. This hybrid clustering scheme incorporated SA as the new focus phase and RDA as the phase of diversification.

The remaining sections of the paper are organized as follows. Section 2 presents the comprehensive review of the existing meta-heuristic algorithms-based cluster head selection approaches contributed in the literature over the recent years. Section 3 depicts the detailed view of the steps included into the proposed RDSAOA-EECP contributed to the cluster head selection with the view to maintain energy stability and improve network lifetime. Section 4 demonstrates the experimental setup and results of the comparative investigation conducted for the proposed RDSAOA-EECP and the benchmarked schemes with proper justifications. Section 5 concludes the paper with major contributions and scope of future research.

## **2. Related Work**

A Fuzzy Clustering and Particle Swarm Optimization (FCPSO)-based cluster head selection was proposed by Ni et al. [17] for controlling the topology and prolonging the network lifecycle. This FCPSO was proposed with the merits of the fuzzy clustering algorithm that initially clustered the sensor nodes based on their geographical locations. This inherited fuzzy clustering process aided in determining the cluster in which each sensor node resides and the complete number of actual clusters that are explored in the process of clustering. This fuzzy clustering depends on an estimated cluster probability that determines which sensor nodes need to be present in which of the possible constructed clusters. The fitness function included into FCPSO considered the factors of distance and energy consumptions into account for clustering process. Then, the improved PSO is used for identifying the cluster head nodes in the hierarchical topology. The results of FCPSO confirmed better reduction in sensor nodes' mortality rate and improved network lifetime independent to the number of sensor nodes and rounds considered for implementation. A Clustering scheme-based on ABC and ACO (ABCACO-CS) was proposed by Janakiraman et al. [18] for attaining better head selection process that aided in better network lifetime and energy stability. The merits of the ACO are potentially included into ABC for preventing the issue of stagnation in the process of exploitation. On the other hand, ABC is included into ACO for the

better exploitation process by preventing the issue of delayed convergence. This integration of ABC and ACO benefits played a vital role in balancing the process of exploitation and exploration. The experimental results of this ABCACO-CS was proved to be better in terms of throughput, residual energy, dead nodes and alive nodes independent to the density of sensor nodes in the network.

Further, Spider Monkey Optimization with Sampling Strategy (SMOSS) was proposed by Lee et al. [19] for extending the stability and lifetime through significant energy consumptions. This SMOSS described the operation of sampling method that aided in the process of selecting cluster heads. It was proposed for resolving the issues that arise due to the locations of sensor nodes and cluster heads that do not lie in the sampling population. It prevented the limitations of location-based cluster head selection that introduced duplicate nodes' selection, poor selection accuracy and increased computation. The results of SMOSS were determined to be better in terms of stability period and network lifetime by 13.4% and 7.1% with homogeneous setup and heterogeneous setup, compared to the baseline clustering schemes. Then, PSO-based Energy Optimized Dynamic Clustering Scheme (PSO-EODCS) was proposed by Jothi Kumar et al. [20] for routing the data to the sink with maximized network lifetime and potential residual energy sustained in the sensor nodes. This PSO-EODCS specifically utilized the Manhattan distance as the metrics for the selection of cluster members associated with the clusters heads selected by PSO. It included the fitness function that included the factors of inactive node energy, active node energy, link quality and location for helping the PSO to select optimal cluster heads. It also employed the algorithm of shortest path in order to establish communication between cluster heads, until they reach the base station. The results of this PSO-EODCS confirmed better network lifetime and energy efficiency of 15.68% and 19.21%, better than the baseline clustering approaches.

Furthermore, a cluster head selection scheme-based on Monarchy Butterfly Optimization (MBO) and Artificial Bee Colony (ABC) was proposed by Rambabu et al. [21] for effective cluster head selection to attain the objective of energy balance and network lifetime. This clustering approach inherited the benefits of mutated butterfly adjusting operator for improving ABC in the dimension of enhancing it to prevent from early convergence into an optimal point of locality. This integration of MBO and ABC also aided in resolving the delay convergence issue that aided in better balancing between the exploitation and exploration process. It was proposed for enhancing the diversity of the population in order to improve its potential towards global search optimization. It also prevented the cluster heads being heavily loaded with maximum number of sensor nodes for the propose of preventing their rapid death. The results confirmed better performance in terms of alive nodes and dead nodes by 18.92% and 16.24%, compared to the baseline clustering approaches. A hybrid GA and Moth Flame Optimization (GAMFO)-based clustering scheme was proposed by Sharma et al. [22] for achieving energy and secure aware routing. This GAMFO included the benefits of moth flame optimization for identifying and selecting the sensor nodes that deserves the role of cluster heads. This cluster head selection process of GAMFO was achieved

using fitness function computation that derived the factors of mean transmission delay, mean cluster distance, connected node density, successive packet forwarding potential and residual energy into account. The simulation results of GAMFO exhibited an improvement in network period enhancement and energy conservation by 34.21% and 23.28%, compared to the baseline schemes considered for investigation.

In addition, Elephant Herd Optimization and Camel Series (EHOCS)-based clustering scheme was proposed by Lavanya and Shankar [23] for attaining better coverage, network lifetime and reliability during the data aggregation process. This EHOCS achieved the balance node establishment that completely concentrated on enhancing the overall network lifetime based the sensed information aggregation process attained at the cluster head. EHO handled the issue of random occurrence inherent with the Camel Algorithm for improving the possibility of optimal cluster head selection. It adopted the factor of visibility monitoring inherent with the Camel algorithm for enhancing exploitation efficiency. On the other hand, the clan and separator operator of EHO played a vital role in compensating the exploration inefficiency of the Camel algorithm to the desired level. The results of this EHOCS also confirmed better network lifetime and energy efficiency of 13.39% and 17.62%, better than the baseline clustering approaches. A hybrid GA and PSO-based clustering scheme (GAPSO-CS) was proposed by Sahoo et al. [24] for attaining better mobility-based data transmission and cluster head selection. This GAPSO-CS specifically included GA for attaining the objective of optimizing the process of cluster head selection. Moreover, PSO was used for determining the optimized route that aided in better sink mobility management to the significant level. It was also proposed for maintaining the balance between exploration and exploitation in order to facilitate rapid and reliable cluster head selection process. The experimental results of GAPSO-CS also confirmed better throughput of 23.28%, improved residual energy of 18.12%, decreased dead nodes by 15.48% and sustained alive nodes by 15.48%, compared to the baseline cluster head selection schemes.

### **3. Proposed Red Deer and Simulation Annealing Optimization Algorithm-based Energy Efficient Clustering Protocol (RDSAOA-EECP)**

The proposed Red Deer and Simulation Annealing Optimization Algorithm-based Energy Efficient Clustering Protocol (RDSAOA-EECP) focuses on the purpose of sustaining network lifetime and energy stability in WSNs.

#### **3.1 Primitive Red Deer Optimization Algorithm**

The Red Deer Optimization Algorithm (RDOA) initially starts with an initial random population of search agents that are deployed randomly over the entire population [25]. In the starting, the sensor nodes that possess high potentiality in terms of residual energy, distance between the cluster heads and the distance between cluster and the base station are selected as the male red deers and the remaining nodes are termed as hinds. The male red deers are the sensor nodes that initially has the probability of being selected as CH. The hinds are the sensor nodes that

acts as cluster members and they are completely controlled by the male red deer nodes. The male red deers (CH nodes) are classified into two categories, such as i) commander CH nodes and ii) stags CH nodes depending on their capability of acting as CH depending on the satisfaction of the objective function. During the process of CH selection, the commander nodes and stags nodes compete with one another and the energy optimal node act as CH of the cluster (harem) during that round. In the clustering process, the number of hinds (cluster members) of a cluster is determined completely depends on the capability of the commander nodes and its controlling potential. The commander sensor node (Primary CH) always complete with the number of hinds in the cluster (harem). Moreover, the stag sensor nodes (secondary CH) competes with the neighborhood hind nodes of the clusters during every rounds of implementation. Further, the aforementioned RDOA steps are developed in such as way that the phases of exploration and exploitation are satisfactorily achieved. Furthermore, the parameters considered during the implementation of the proposed RDOA have the feasibility of fine tuning its inherited phases. The competitive phase of male RDs is considered as the counterpart of exploitation in order to improved the properties of local search in the solution space. Further, the competitive phase of the commander and stag sensor nodes is also determined as the local search, even though it accepts only the optimal solutions observed during the process. The exploration phase is applied, once the clusters are constructed and the commander sensor nodes (primary CH) are assigned to them depending on its potential. In addition, the commander sensor nodes of a cluster compete with percentage of sensor members within cluster and the percentage of sensor nodes associated with the other clusters in order to improve the properties of exploration. Moreover, the stag sensor nodes (secondary CH) competes with the neighborhood hind nodes that exists in their local proximity. This phase of competition between the stag sensor nodes and the hind nodes attribute towards both the phases of exploration and exploitations, concurrently. This RDOA includes the mating process that results in the generation of potential offspring (candidate solutions) for contributing towards better new solutions in the solution space. It also facilitates superior chance for the worst solutions in order to prevent worst sensor nodes being selected as CH.

### 3.1.1 Initial red deer population generation

The primary of optimization targets in determining the global solution or near-optimal solution with respect to the variables utilized in the problem. In this optimization problem of CH selection, an array of variables are considered for optimization. In this RDOA, red deer represents the complete set of feasible solution ( $F_S$ ) that lies inside the solution space. In this context, the dimensionality of the solution in the solutions depends on the number of factors ( $d$ ) considered during the selection of CH. This RDOA algorithm of CH selection is considered as a  $d$ -dimensional optimization problem, then the red deer represents the number of  $d$ -dimensional array that ranges between 1 to  $d$ . This red deer array is defined using Equation (1).

$$R_{D(Array)} = [F_{S(1)}, F_{S(2)}, F_{S(3)}, \dots, F_{S(d)}] \quad (1)$$



Where,  $R_{D(Array)}$  represents the individual dimensions through which the feasible solution ( $F_S$ ) can be investigated during the selection of CH. Then, the value fitness function (formulated based on residual energy, distance between the cluster members and the CH and distance between CH and base station) can be evaluated for each red deer as defined in Equation (2).

$$Fit_{value}(R_{D(Array)}) = Fit_{Fn}[F_{S(1)}, F_{S(2)}, F_{S(3)}, \dots, F_{S(d)}] \quad (2)$$

Initially, the initial population size ( $P_{Size}$ ) is generated to start the implementation of the algorithm. Then, the set of best red deer (search solution) is selected as best male RDs ( $RD_{Male}$ ) and the remaining search solutions are designated as hind deers ( $RD_{Hinds} = P_{Size} - RD_{Male}$ ). It is notable that the number of male deer solution represents the elitist criteria of the algorithm. Moreover, the number of  $RD_{Male}$  and  $RD_{Hinds}$  aids in maintaining the intensification and diversification of the algorithm.

### 3.1.2 Male red deer roaring phase

The male red deer solutions are the superior solutions in the algorithmic search space. In this roaring phase, the objective function of the male red deer solutions and its identified neighboring solutions are compared in order to determine the best among them in order to facilitate better CH selection. If the fitness function of the neighbor red deer solutions are better than the male red deer solutions, then the replacement is attained with the prior ones. In specific, the male red deer solutions always change their position and its updating is attained based on Equation (3).

$$RD_{Male} = \begin{cases} RD_{Male-Old} + UD_{UD_{R(2)}} \times (((U_{TH} - L_{TH}) * UD_{R(2)}) + L_{TH}) & \text{If } UD_{R(3)} \geq 0.5 \\ RD_{Male-Old} - UD_{UD_{R(1)}} \times (((U_{TH} - L_{TH}) * UD_{R(2)}) + L_{TH}) & \text{If } UD_{R(3)} < 0.5 \end{cases} \quad (3)$$

Where  $RD_{Male-Old}$  represents the current position of male red deer solution with  $U_{TH}$  and  $L_{TH}$  depicting the upper and lower search space limiting threshold in order to generate the possible neighboring solution of the male red deer. In addition, the random variables such as  $UD_{R(1)}$ ,  $UD_{R(2)}$  and  $UD_{R(3)}$  are generated using uniform distribution (for including randomization in the roaring process) that lies between 0 and 1.

### 3.1.3 Male commanders selection from ' $\beta$ ' percent of male red deer solutions

In this phase, the male red deer solutions are categorized into commander and stage red deer solutions (in practical, the difference between red deers have huge difference in terms of power, attractiveness and success in expanding its territory). In the same way, the commander and stage red deer solutions are determined based on their potential in acting as CH with respect to residual energy, inter-cluster distance, intra-cluster distance and its significance in controlling its cluster members. At this juncture, the number of male commander solutions are determined based on Equation (4)

$$RD_{Comd-Count} = Round(\beta, RD_{Male}) \quad (4)$$

Where ' $RD_{Cmd-Count}$ ' represents the number of commander male deer solutions in the clusters (harems). It is also noted that ' $\beta$ ' is the initial value associated with the proportion of solutions considered for exploitation. This value of ' $\beta$ ' lies between 0 and 1. At this juncture, the stag red deer solutions ( $RD_{Stag-Count}$ ) are determined based on Equation (5)

$$RD_{Stag-Count} = RD_{Male} - RD_{Cmd-Count} \quad (5)$$

### 3.1.4 Stag and Male commanders fighting phase

In this phase of fighting, each commander red deer solutions are compared with stag solutions randomly by letting them to approach towards each other. Thereby, two new solutions are determined and made to replace the commander red-deer solutions by the superior optimal solutions. In this context, one superior most solution is determined compared to the commander, stag and two new solutions ( $New_{Sol(1)}$  and  $New_{Sol(2)}$ ) attained after the process of approaching). This process of fighting is represented mathematically based on Equation (6) and (7).

$$New_{Sol(1)} = \frac{(Sol_{Cmd} + Sol_{Stag})}{2} + UR_{D(1)} \times ((U_{TH} - L_{TH}) * UR_{D(2)}) + L_{TH} \quad (6)$$

$$New_{Sol(2)} = \frac{(Sol_{Cmd} + Sol_{Stag})}{2} - UR_{D(1)} \times ((U_{TH} - L_{TH}) * UR_{D(2)}) + L_{TH} \quad (7)$$

Where,  $U_{TH}$  and  $L_{TH}$  represents the upper and lower search space limiting threshold in order to generate the new solution in the fighting phase. Moreover,  $UR_{D(1)}$  and  $UR_{D(2)}$  are also random variables generated using uniform distribution (for including randomization in the fighting phase). This value of  $UR_{D(1)}$  and  $UR_{D(2)}$  also lies between 0 and 1. In this fighting phase, the best solution is selected among the identified commander, stag and two new solutions ( $New_{Sol(1)}$  and  $New_{Sol(2)}$ ). In this fighting phase, the first solution ( $New_{Sol(1)}$ ) selects the winner solution and the second solution ( $New_{Sol(2)}$ ) helps in identifying losing solution.

### 3.1.5 Construction of Clusters (Harems)

In the clusters construction phase, male commander solution becomes the cluster head and the group of hinds (cluster members) are controlled by CH. In order to construct clusters, the number of hind solutions are proportionally shared among the commander solutions (CH) based on Equation (8)

$$N_{Val} = Cmd_{Power(n)} - Max(Cmd_{Power(i)}) \quad (8)$$

Where,  $N_{Val}$  and  $Cmd_{Power(n)}$  refers to the normalized value of the commander and power of each individual commander at each round of implementation. Then, the commanders' normalized power is calculated based on Equation (9)

$$RD_{Cmd(Power)} = \left| \frac{Cmd_{Power(n)}}{\sum_{i=1}^{RD_{Cmd-count}} Cmd_{Power(i)}} \right| \quad (9)$$

Where,  $RD_{Cmd(Power)}$  represents the power possessed by each commander red deer during the process of clustering and CH selection. This determines the number of hind solutions that that could be controlled by the commander solutions during the clustering process. Thus the number of hinds that could possibly exists in a cluster (harem) is determined based on Equation (10)

$$RD_{Hind-Harem} = Round(\alpha, RD_{Hind-Cnt}) \quad (10)$$

Where,  $RD_{Hind-Harem}$  represents the number of hinds that could exist in a cluster during the process of CH selection. As a whole, the commander red deer solution that possess better fitness function is determined to be better than the hind solutions. The commander and hinds represent the CH and cluster members in the process of clustering in WSNs.

### 3.1.6 Commander red deer mating with a percent of hinds of a cluster (harem)

The process of mating is achieved by a commander red deer solution with a percent of hinds that exists in the cluster (harem) where its resides. This mating process refers to the strategy of integrating commander and some specific hind solutions of the search space in order to generate offspring solutions as defined in Equation (11).

$$Hind_{Count(Mate)} = (\gamma, Harem_{nth}) \quad (11)$$

Where,  $Hind_{Count(Mate)}$  refers to the number of hind solutions considered from the  $n^{th}$  cluster to mate with the associated commander red deer solutions in the solution space. This  $\gamma$  percent of hind solutions are randomly selected from the total number of constructed clusters. It is noted that ‘ $\gamma$ ’ is the initial parameter value that ranges between 0 and 1.

This process of mating that concentrates on the generation of potential offspring solutions are determined based on Equation (12).

$$Offspring_{Sol} = \frac{(Sol_{Comd} + Sol_{Hind})}{2} + (U_{TH} - L_{TH}) \times U_{RV} \quad (12)$$

Where,  $U_{RV}$  is also a uniformly distributed random function whose value also lies between 0 and 1.

### 3.1.7 Commander red deer mating with a percent of hinds of another clusters (harems)

In this mating process, a new offspring solution is determined by integrating the male commander solution with some percent of hinds that are present in another clusters (say k). This selection of  $k^{th}$  cluster for mating is completely random.

$$Hind_{Count(Mate-k)} = (\delta, Harem_{kth}) \quad (13)$$

In this case,  $\delta$  percent of hind solutions are selected from  $k^{th}$  constructed cluster selected randomly. In this process of mating, the generation of offspring solution is determined based on Equation (14)

$$Offspring_{Sol(Comd-hind(kth))} = \frac{(Sol_{Comd} + Sol_{Hind(k)})}{2} + (U_{TH} - L_{TH}) \times U_{RV} \quad (14)$$

### 3.1.8 Stag mating with neighboring hinds

In this step, each stag solutions are made to get mate with the neighborhood hind (the solutions that is considered to be more optimal on par with all the hinds in various clusters of the network) for superior generation of offspring solutions. In this case, the stag solutions determines the nearest hind solution based on the distance between each stag solution with all hinds in the d-dimensional space presented in Equation (15)

$$Dist_{Stag-Hind} = \sqrt{\sum_{d \in D}^{ND} (Sol_{stag} - Sol_{Hind(j)}^{(i)})^2} \quad (15)$$

Where,  $Dist_{Stag-Hind}$  refers to the distance between a stag and the hind solutions. In addition, the process of mating is completely taken care by the mathematical formula specified in Equation (12).

### 3.1.9 Next generation selection process

In this proposed RDOA-based CH selection scheme, two different methodologies are adopted for selecting the subsequent generations. During the first strategy, the male red deer, all stags and commander (only a percent of the best optimal solutions from the complete set of solutions in the search space is considered). In the second strategy, the remaining population is considered in the next generation during the exploration process. In particular, the roulette wheel or fitness tournament mechanism is adopted for selecting optimal hind solutions from the complete set of hinds and offspring solution achieved during different mating process

### 3.1.10 Termination Criterion

The termination criterion for stopping the execution of the algorithm is attained whenever the quality of best solution is unchanged for a time interval or the completion of maximum number of rounds considered for implementation.

## 3.2 Improved Simulated Annealing

The classical Simulated Annealing (SA) algorithm was proposed as a single solution-based metaheuristic optimization approach developed based on the metallurgy annealing phenomenon. In SA, the defects are reduced by controlling the metals cooling process during the construction of

stable crystal structures. The energy of the thermodynamic system is represented by the value of the objective function utilized in SA. It adopted an iterative approach through which the search agents updates their position based on a random walk procedure during each and every iteration. This procedure of random walk is included by considering deviation that are estimated by magnitude highlighted through temperature. The algorithm permits large movements at high temperatures in the search space and accepts only the solutions whose probability does not get improved compared to the optimal value determined in the previous iterations. Hence, SA prevents getting stuck in a local solution. However, the probability and perturbation for accepting worst solution is dramatically reduced at low temperatures. In general, SA starts with an initial solution  $S_{Initial}$  and a mechanism of cooling  $C_M$  with each element temperature  $t_{(i)}$  with respect to individual iteration ‘ $i$ ’ is specified in Equation (16)

$$C_{M(i)} = \{t_{(1)}, t_{(2)}, \dots, t_{(n)}\} \quad (16)$$

Then, a process of iteration is initiated during which a neighboring solution ( $N_{Sol(i)}$ ) close to the current solution ( $C_{Sol(i)}$ ) is generated based on the utilization of random perturbation in each iteration ‘ $i$ ’. In this context, temperature  $t_{(i)}$  influences the perturbation strength. When the temperature is low, small perturbations are visualized. In contraction, bigger perturbation is identified with high temperature. Further, the solutions ( $N_{Sol(i)}$ ) and ( $C_{Sol(i)}$ ) are compared for identifying the best solution. The preservation of the current best solution ( $C_{Sol(i)}$ ) or replacing the current solution with newly estimated neighboring solution ( $N_{Sol(i)}$ ) completely depends on the probability  $P_{accept}$ . This probability of acceptance (formulated based on minimization function) considered for accepting a solution is determined based on Equation (17)

$$P_{accept} = \begin{cases} e^{-\frac{f(N_{Sol(i)}) - f(C_{Sol(i)})}{t_{(i)}}} & \text{If } f(N_{Sol(i)}) - f(C_{Sol(i)}) > 0 \\ 1 & \text{If } f(N_{Sol(i)}) - f(C_{Sol(i)}) \leq 0 \end{cases} \quad (17)$$

In this case, the temperature is decreased based on the employed cooling mechanism without considering whether the new solution was accepted or rejected. This iterative process is continued unless the temperature reaches the defined final value. However, the exploration and exploitation capabilities still possesses a room for improvement. Hence, an improved SA based on the merits of OrikaeshiTanrenJapanese technique was proposed with steel folding and reheating incorporated into the traditional SA for attaining superior exploration and exploitation capabilities [26].

This Improved OrikaeshiTanren SA (IOTSA) considered a population of ‘ $u$ ’ agents. Every individual search agent is responsible for achieving search strategy based on the improvement included over the classical SA scheme. This IOTSA algorithm included two new significant operators such as folding and reheating. In specific, folding process is perceived as the search space compression and reheating mechanism referring to the cooling process re-initialization included into the classical SA approach. Therefore, folding process concentrates on the

determination of best-current solution, while reheating phase is always applied after folding process in order to compress the search space. Thus, the reheating process plays a vital role in exploring the search agent capability that attributes towards reduced area exploration. Moreover, this process of folding and reheating potentially improves the capabilities of exploitation and exploration inherent with the primitive SA. This SA algorithm starts with a population of ‘ $u$ ’ agents from the first iteration to the maximum number of iterations. In the first iteration, the search agents are initialized randomly. Then, SA scheme is adopted as the search strategy by the search agents until the number of iterations reaches  $Iter - Fold$ . After the fold iterations, an operation of folding is incorporated for determining the best search agent from the complete set of ‘ $u$ ’ agents existing in the search population. This folding process enables the search space compression by considering the best search agent as the positional center. This process calculates the position of all the search agents within the newly determined search space based on the recently compressed search space. Finally, the operation of reheating is included for re-initializing the initial temperature of the search agents. This SA scheme also used stop criterion as another dimension of implementation.

### 3.2.1 Population Initialization

A population of ‘ $u$ ’ agents  $[a_1, a_2, \dots, a_u]$  is generated randomly based on Equation (18)

$$a_{(i)} = rand(B^{Up} - B^{Low}) + B^{Low} \quad (18)$$

Where,  $B^{Low}$  and  $B^{Up}$  are the lower and upper boundaries representing the d-dimensional search space with ‘ $rand$ ’ considered as a uniformly distributed random number that ranges between 0 and 1. Further, the value of the initial and final temperature are determined based on Equation (19) and (20).

$$Temp_{(Init)} = (B^{Up} - B^{Low}) * \frac{Length_{SP}}{100} \quad (19)$$

$$Temp_{(Final)} = (Temp_{(Init)} - R_{Cooling})^{(Iter-Fold)} \quad (20)$$

Where,  $Length_{SP}$  and  $Iter - Fold$  refers to the search space full length and number of iterations before the folding process. Moreover,  $R_{Cooling}$  represents the rate of cooling in the SA phenomenon. The aforementioned Equations clearly guaranteed that the initial and final temperature depends on the search space and initial number of iterations in the process of folding, respectively. Further, normalization in the initial temperature is essential as the search agents movement completely depends on the temperature. Otherwise, the algorithm without normalized initial temperature is impacted by the deviations of the search space size. This IOTSA algorithm possess different configuration parameters as similar to other metaheuristic algorithms. The configuration of population size (100), maximum number of folds (50), rate of cooling (0.3), number of folding iterations (20) are considered during the implementation process.

### 3.2.2 Process of classical SA

In this phase, IOTSA algorithm adopts the merits of classical SA as the search mechanism for the set of search agents in order to achieve a movement and cooling process. In particular, IOTSA algorithm a geometric approach as the cooling strategy due to its good results and simplicity. This geometric method of cooling used in this phase is presented in Equation (21).

$$Cooling_{(strategy)} = (Temp_{(Init)} - R_{Cooling})^{(Iter)} \quad (21)$$

At this juncture, the search agent movement is attained based on random perturbation with respect to their positions updated by temperature and standard deviation defined in Equation (22)

$$a_{(i)}^{(k+1)} = a_{(i)}^{(k)} + rand(n) * \sigma * T \quad (22)$$

In addition, the search agents performs large and small perturbations depending on the high or low temperature.

### 3.2.3 Process of Folding

This process of folding minimizes the width or length of the search space, at the same time increases the search agents concentration in the minimized search space. Thus, a new search space comparatively less than the search space provisioned by the classical SA is defined at the end of the folding process. Further, the search space reduction is attained through linear reduction for the objective of minimizing the algorithmic computational complexity. With this operation, folding factor ( $Factor_{Fold}$ ) is used for calculating new boundaries (lower- $SP_{Reduced(Low)}$  and upper- $SP_{Reduced(High)}$  boundaries) for minimizing the original search space size as presented in Equation (23) and (24).

$$SP_{Reduced(Low)} = \frac{B^{Low}}{Factor_{Fold}} \quad (23)$$

$$SP_{Reduced(High)} = \frac{B^{Up}}{Factor_{Fold}} \quad (24)$$

Where,  $B^{Low}$  and  $B^{Up}$  are the lower and upper boundaries representing the d-dimensional original search space.

Once different iterations with respect to  $Iter - Fold$  operations are conducted by the classical SA method, then the process of folding is initiated. In this folding process, the solution with best fitness value ( $B_{Best}$ ) is selected and considered as the center in order to reduce the search space. The complete range of new search space is accomplished by the inclusion of limits ( $LL_{Fold}$ ) and  $UL_{Fold}$  associated with them based on Equation (25) and (26)

$$LL_{Fold} = S_{Best} - \left( \frac{SP_{Reduced(High)} - SP_{Reduced(Low)}}{2} \right) \quad (25)$$

$$UL_{Fold} = S_{Best} + \left( \frac{SP_{Reduced(High)} - SP_{Reduced(Low)}}{2} \right) \quad (26)$$

Moreover, the folding factor ( $Factor_{Fold}$ ) is adjusted for preventing premature convergence during the process of optimization. This  $Factor_{Fold}$  starts with the small value and it is considered to increase based on Equation (27).

$$Factor_{Fold} = \begin{cases} LL_{Fold} + k * \left( \frac{UL_{Fold} - LL_{Fold}}{\frac{Fold\_Max}{2}} \right) & \text{If } k < \frac{Fold\_Max}{2} \\ UL_{Fold} & \text{Otherwise} \end{cases} \quad (27)$$

The aforementioned equation is defined in such a way that the folding factor  $Factor_{Fold}$  begins with lower limit  $LL_{Fold}$  and reaches the upper limit  $UL_{Fold}$ , when the iteration 'k' of the current fold reaches  $\frac{Fold\_Max}{2}$ .

### 3.2.4 Process of Reheating

Reheating process gets initiated after the completion of folding process. This reheating process is responsible for re-initialization of current temperature from the classical SA. It is also responsible for restarting the search potential of the SA algorithm. This process of temperature re-initialization is attained based on Equation (28).

$$t_{(i)} = Temp_{Init} \quad (28)$$

The algorithm permits large perturbations in the search space at high temperature. The probability involved in solutions acceptance does not improve, when the previously estimated value is higher. Therefore, this IOTSA algorithm prevents getting stuck in a local solution with greatly reduced probability and perturbation incurred in accepting worst solution at low temperatures.

### 3.3 Red Deer and Simulation Annealing Optimization Algorithm-based CH Selection

In this proposed RDSAOA-EECP scheme, RDA and SA simulation are integrated and applied as a potential hybrid algorithm by adjusting the parameters with one or more of the necessitated dimensions. In the hybrid algorithm, SA and RDA serves as a core tool for facilitating significant local search process. In specific, the important phases of roaring and fighting included in RDOA referred as the focus phase is eliminate through the folding and reheating phase of SA in order to facilitate better balance between the degree of exploitation and exploration. This hybrid clustering scheme incorporated SA as the new focus phase and RDA as the phase of diversification. In each and every individual generation, individual brocket acts as the input to SA such that



maximized optimization during CH selection can be attained to the expected level. The integrated RDOA is a population-based algorithm that incorporated multiple stages by minimizing computation time and excluding specific stages by submitting the rules of IOTSA scheme. In addition, the detailed process incorporated in the proposed RDSAOA-EECP scheme is presented in Algorithm 1.

**Algorithm 1: RDSAOA-EECP protocol-based CH selection process**

- 1.1 Initialize the population of RDA (Search agents that are randomly deployed over sensor nodes) for exploring the potential of sensor nodes based on the factors of residual energy, inter-cluster distance and intra-cluster distance.
- 1.2 Compute the fitness value of randomly deployed sensor nodes and sort them into best and worst potential sensor nodes
- 1.3 Designate the best sensor nodes into male RDs and the remaining worst sensor nodes into hinds.
- 1.4 Initialize the fronts of Pareto Optimality
- 1.5 Until ( $t < Max_{Iteration}$ ) //  $Max_{Iteration}$  -Maximum number of iterations )
- 1.6 For every individual male RD
- 1.7 Iteration=Iteration+1
- 1.8 While
- 1.9 If the generated new solution is better than the old solution
- 1.10 New solution replaces the old solution
- 1.11 Else
- 1.12 Calculate ‘  $\delta$  ’ based on  $\delta = |Fit_{Old} - Fit_{New}|$  (Apply folding and reheating phase of SA explained in 3.2.3 and 3.2.4)
- 1.13 If  $rand < \exp(-\frac{\delta}{T_{Max}})$  (Re-initialize temperature used in IOTSA based on Equation (28) )
- 1.14 Replace the Old solution
- 1.15 End If
- 1.16 End If
- 1.17  $Iter_{Sub} = Iter_{Sub} + 1$
- 1.18 End While
- 1.19 End For
- 1.20 Update  $T_{Max}$ .
- 1.21. Arrange the males (best search agents) and arrange the solutions attained from them into solutions designated as stags and the commanders.
- 1.22 Construct Cluster based on Equation (8), (9) and (10)
- 1.23 For every individual male commander solution

- 1.24 Randomly mate the male commander solutions with the selected hind solution for determining offspring solutions
- 1.25 Determine offspring solutions based on Equation (12).
- 1.26 Randomly select a cluster 'k' (Harem).
- 1.27 Generate new offspring solutions based on the male commander and randomly selected hind solutions based on Equation (14).
- 1.28 End For.
- 1.29 For every stag solution in the population space
- 1.30 Compute the distance between all the hind solutions and the stag solutions and select the optimal hind solution for CH selection based on Equation (15)
- 1.31 Generate new solution based on selected hind and male stag solution in the search space.
- 1.32 End For
- 1.33 Use Roulette wheel selection by selecting the next generation
- 1.34 Enforce Pareto optimal solutions updating.
- 1.35  $Iter_{Sub} = Iter_{Sub} + 1$
- 1.36 End while.
- 1.37 Utilize the best solution and determined the solutions based on objective function formulated for CH selection.

#### 4. Simulation Results and Discussions

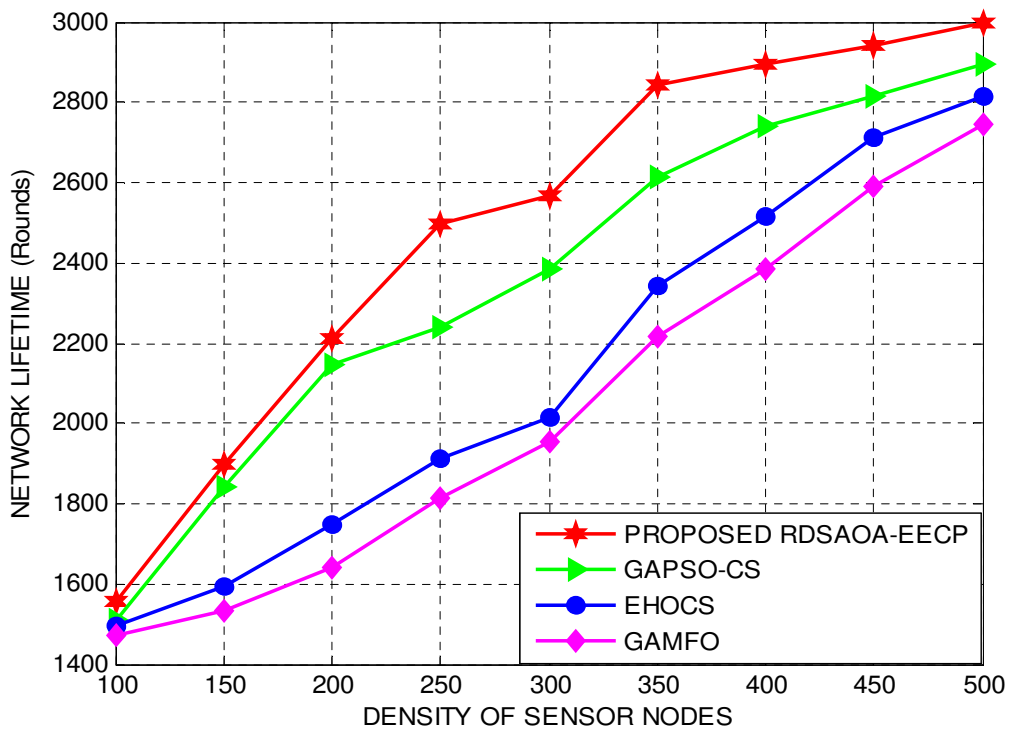
The proposed RDSAOA-EECP protocol and the benchmarked GAPSO-CS, EHOCS and GAMFO approaches are evaluated using MATLAB 2014b version installed over Intel Core i5 processor with 8GB RAM and 2.30 GHz CPU [27-31]. The simulation environment comprises of 500 sensor nodes deployed randomly in a two-dimensional network area of 1500x1500 square meters with one sink node placed at the position of (500m,500m) in the network. The number of rounds used for implementation is 3500 rounds. In addition, the simulation setup [32-34] considered for implementing the proposed RDSAOA-EECP protocol is presented in Table 1.

**Table 1: Simulation setup of the proposed RDSAOA-EECP protocol**

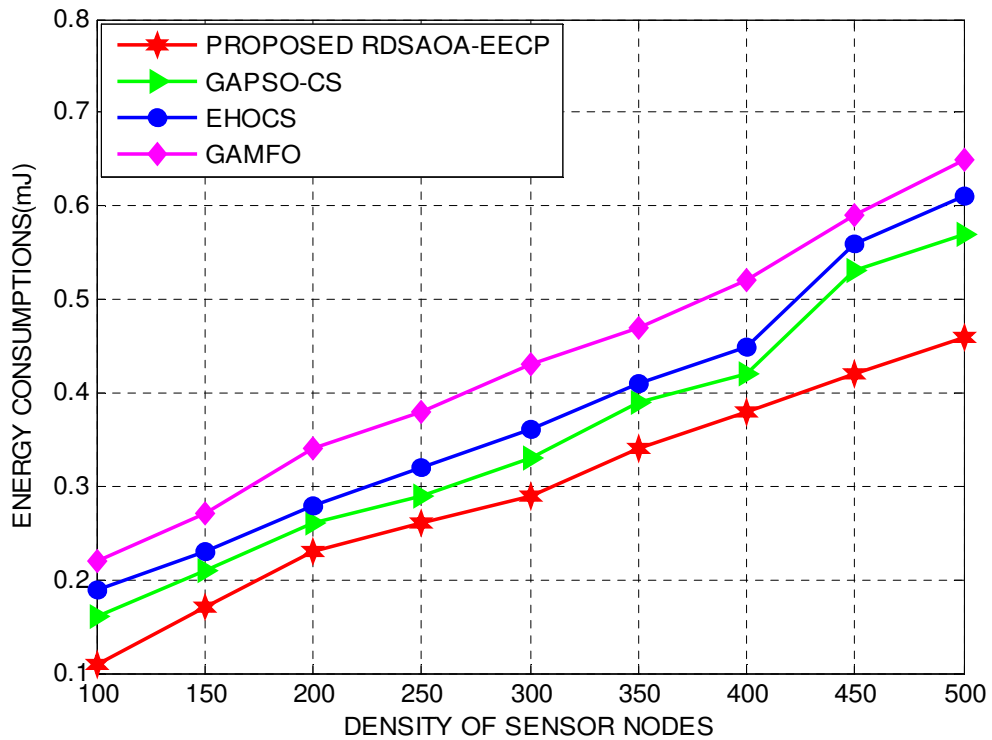
Simulation parameters	Values used
Network area	1500x1500 square meters
Number of sensor nodes	500
Number of rounds used for implementation	3000
Location of the sink node	(500m,500m)
Size of the packets	10000 bits
Initial energy of the node	2 Joules
Range of nodes	20-30 meters
$E_{amp}$	100 pJ/bit/square meters
$E_{elec}$	50 nJ/bit
$E_{DA}$	5 nJ/bit/signal

Maximum network throughput	1 Mbps
----------------------------	--------

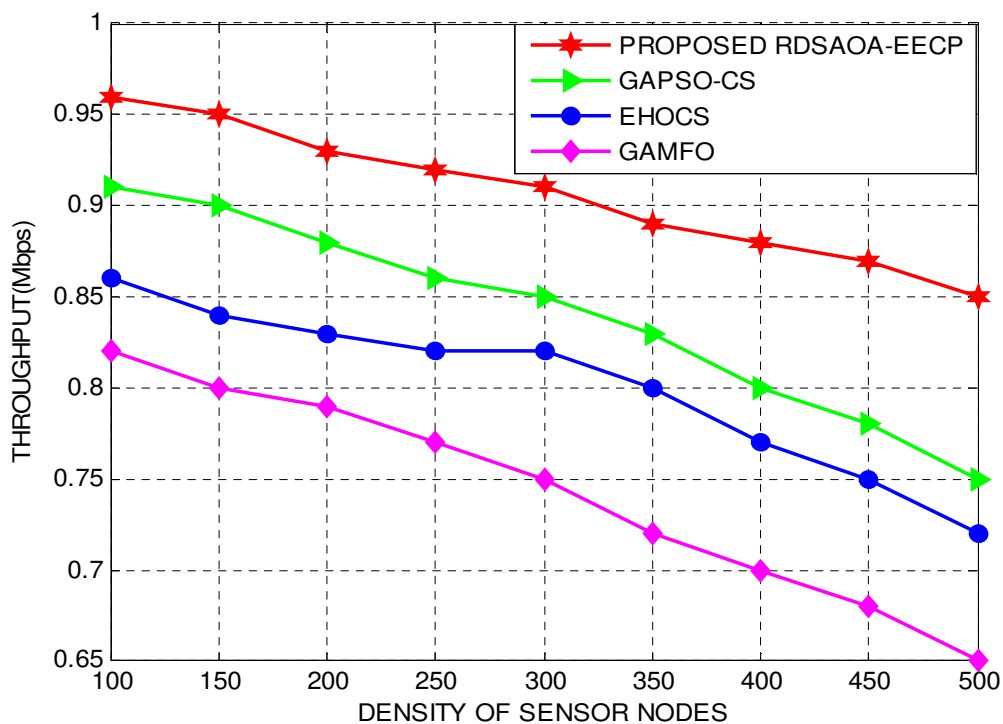
Initially, the proposed RDSAOA-EECP protocol and the benchmarked GAPSO-CS, EHOCS and GAMFO approaches are evaluated based on Network Lifetime (rounds), Energy Consumptions, Throughput and Packet Delay with density of sensor nodes. Figure 2 and 3 depicts the network lifetime and energy consumptions of the proposed RDSAOA-EECP protocol and the benchmarked schemes with respect to densities of sensor nodes (varied from 100 to 500 nodes). The network lifetime of the proposed RDSAOA-EECP protocol and the benchmarked schemes confirm better lifetime with systematic increase in the density of sensor nodes, since it integrated the benefits of IOTSA into the RDOA for selecting the superior nodes of the network as cluster head. This predominance of the proposed RDSAOA-EECP protocol is attained during the substitution roaring and fighting phase of RDOA with folding and reheating phase of IOTSA for maintaining the tradeoff between exploitation and exploration during the search space. The energy consumptions of the proposed RDSAOA-EECP protocol and the benchmarked schemes is determined to systematic increase with corresponding increase in sensor node densities. However, the proposed RDSAOA-EECP protocol is capable enough in sustaining energy consumptions, since it included different strategies that integrated different candidate solutions into potential offspring solutions . This determination of potential offsprings targeted on the significant removal of worst solutions that results in preventing the least important sensor nodes from being selected as cluster heads. With different densities of sensor nodes, the proposed RDSAOA-EECP protocol increases network lifetime by 8.21%, 9.94% and 11.16%, predominant to the baseline GAPSO-CS, EHOCS and GAMFO approaches. The energy consumptions of the proposed RDSAOA-EECP protocol with different densities of sensor nodes also get minimized by 8.92%, 9.74% and 12.92%, better than the benchmarked schemes.



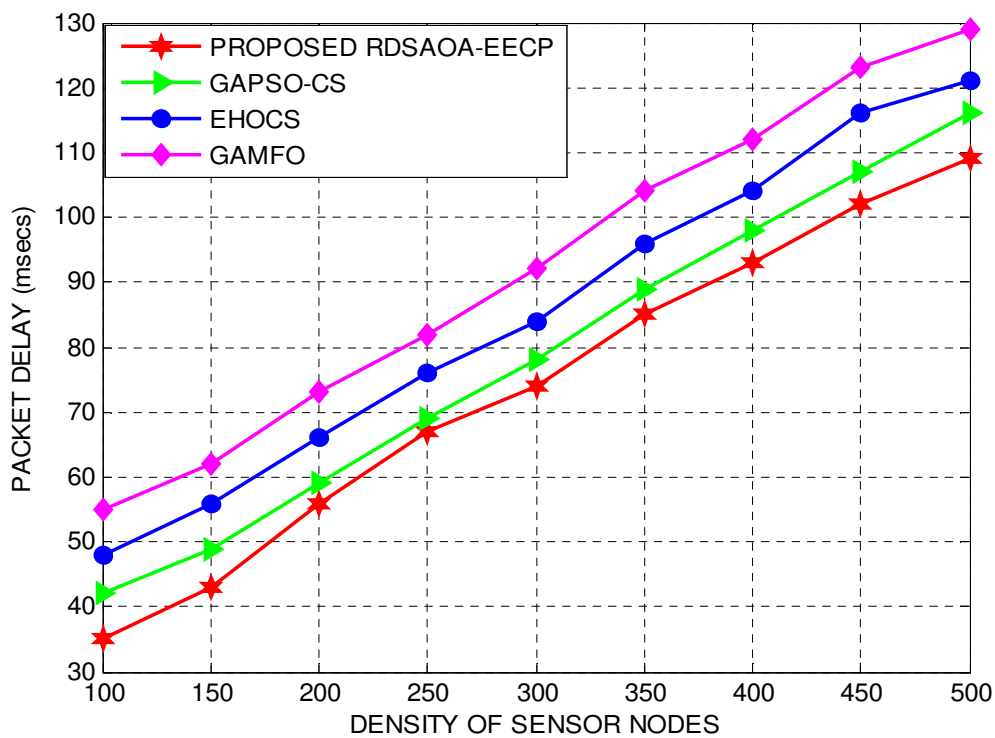
**Figure 1: RDSA OA-EECP: Network Lifetime (Rounds) with density of sensor nodes**



**Figure 2: RDSA OA-EECP: Energy Consumptions with density of sensor nodes**



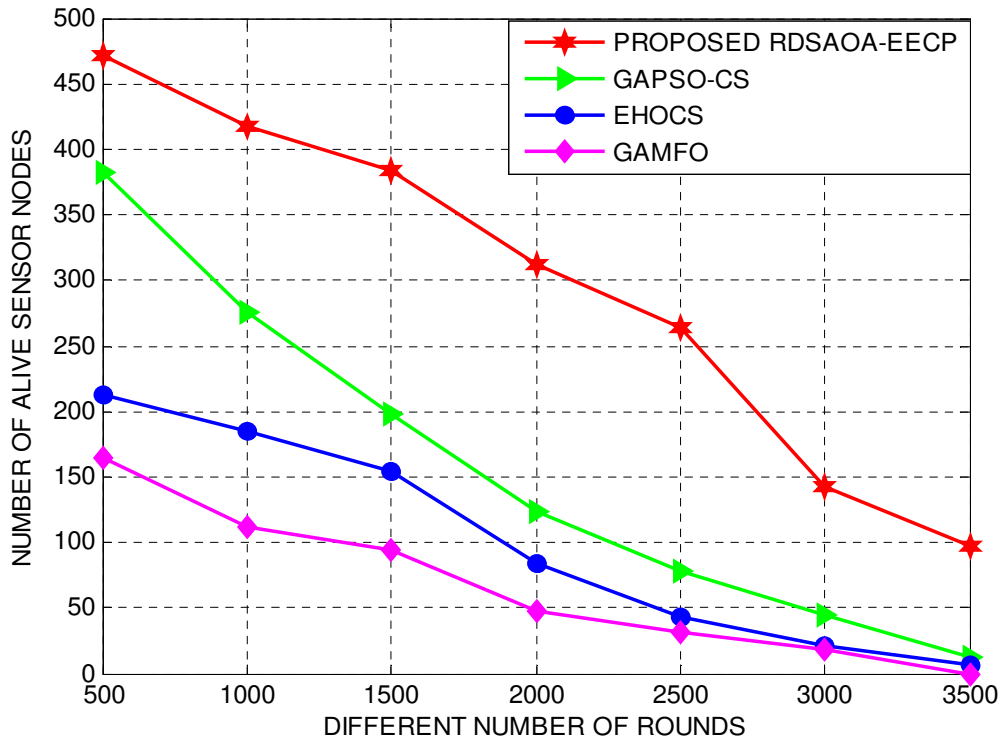
**Figure 3: RDSA OA-EECP: Throughput with density of sensor nodes**



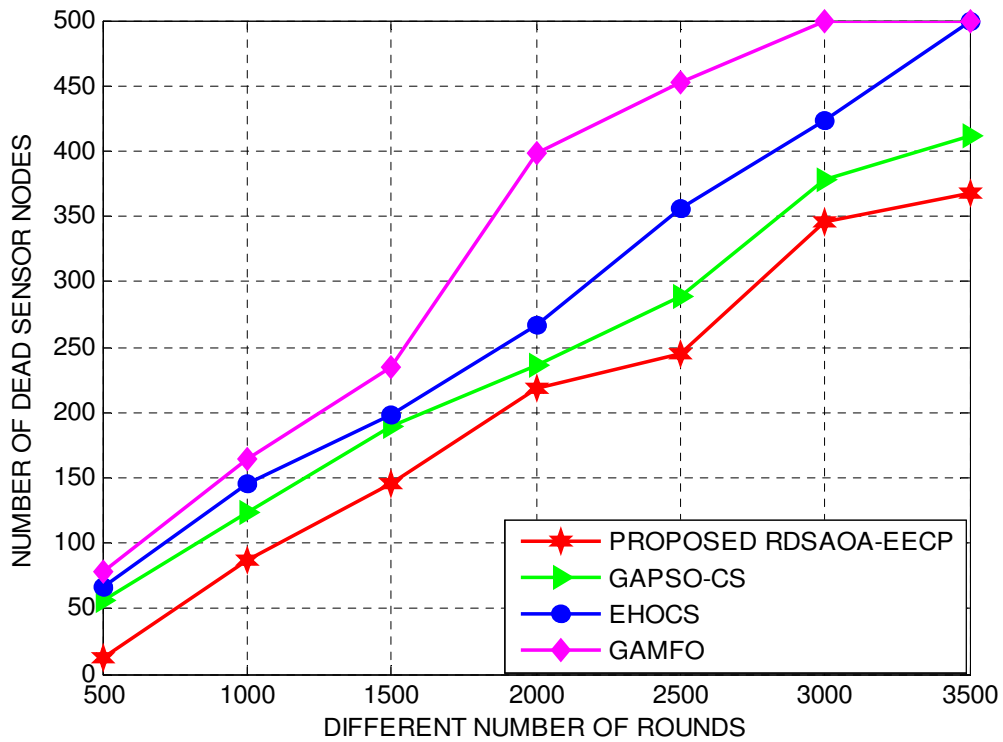
**Figure 4: RDSA OA-EECP: Packet latency with density of sensor nodes**

Figure 3 and 4 highlights the throughput and packet latency of the proposed RDSAOA-EECP protocol and the benchmarked schemes with respect to densities of sensor nodes (varied from 100 to 500 nodes). The throughput of the proposed RDSAOA-EECP protocol with scalable increase in the density of sensor nodes is confirmed to be better than the baseline GAPSO-CS, EHOCS and GAMFO approaches. This capability of the proposed protocol gets proportionally increased due to the merits of reheating that aided in better selection of CH that prevented packet dropping to an expected level. On the other hand, the packet latency in the network attained by the proposed RDSAOA-EECP protocol is considerably minimized as it incorporated the process of exploring the solutions by categorizing them into male commander, stag and hind candidate solutions. This process of classifying solutions and generating significant offspring solutions always guaranteed the selection of potential CH in the network. Therefore, the proposed RDSAOA-EECP protocol with different densities of sensor nodes improves the throughput by 9.56%, 10.82% and 11.74%, excellent to the baseline GAPSO-CS, EHOCS and GAMFO approaches. The packet latency of the proposed RDSAOA-EECP protocol with different densities of sensor nodes also get minimized by 7.69%, 8.42% and 10.66%, better than the benchmarked schemes.

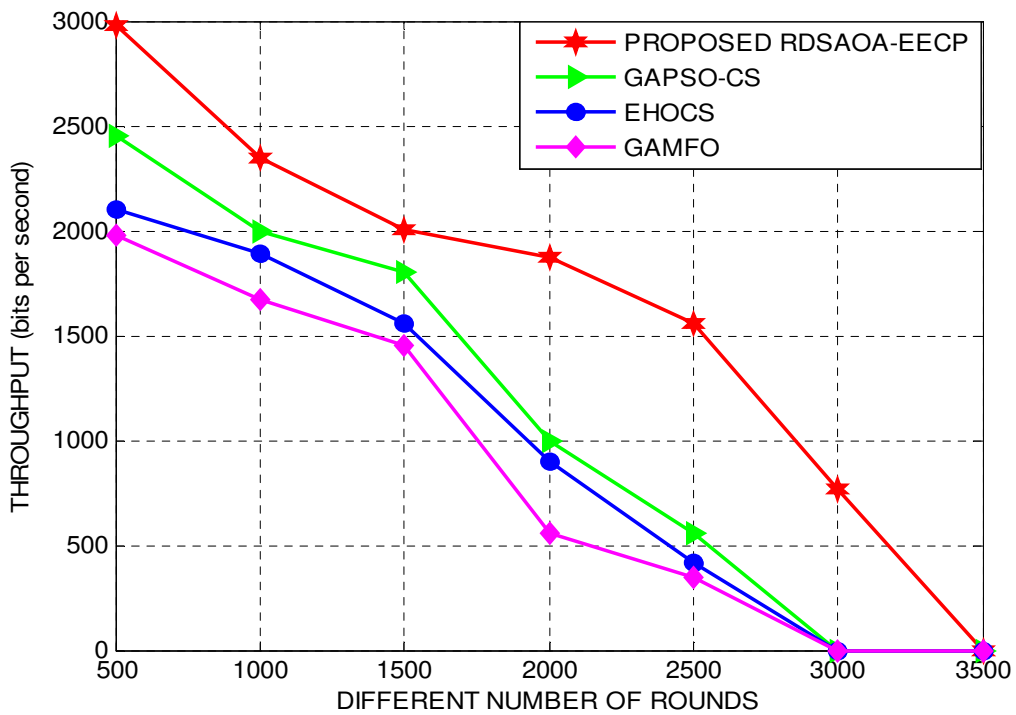
Further, the proposed RDSAOA-EECP protocol and the benchmarked GAPSO-CS, EHOCS and GAMFO approaches are evaluated based on number of alive sensor nodes, number of dead sensor nodes, throughput and mean residual energy with different number of rounds. Figure 5 and 6 demonstrates the number of alive sensor nodes and number of dead sensor nodes present in the network during the implementation of the proposed RDSAOA-EECP protocol and the benchmarked GAPSO-CS, EHOCS and GAMFO approaches. The number of alive nodes sustained by the proposed RDSAOA-EECP scheme is considered to be predominant over the baseline schemes, since the energy of the network is conserved by selecting only the potential sensor nodes as CH during the clustering process. This selection process often prevents unnecessary clustering as it prevents worst sensor nodes from being chosen CH node. It is responsible for preventing unnecessary draining of energy in the sensor nodes, thereby maintaining the energy of the network to an expected level. Hence, the proposed RDSAOA-EECP protocol with different number of rounds sustains number of alive nodes by 10.86%, 11.53% and 13.82%, excellent to the baseline GAPSO-CS, EHOCS and GAMFO approaches. The number of dead nodes minimized by the proposed RDSAOA-EECP protocol with different number of rounds also gets reduced by 9.64%, 11.72% and 12.76%, excellent to the benchmarked schemes.



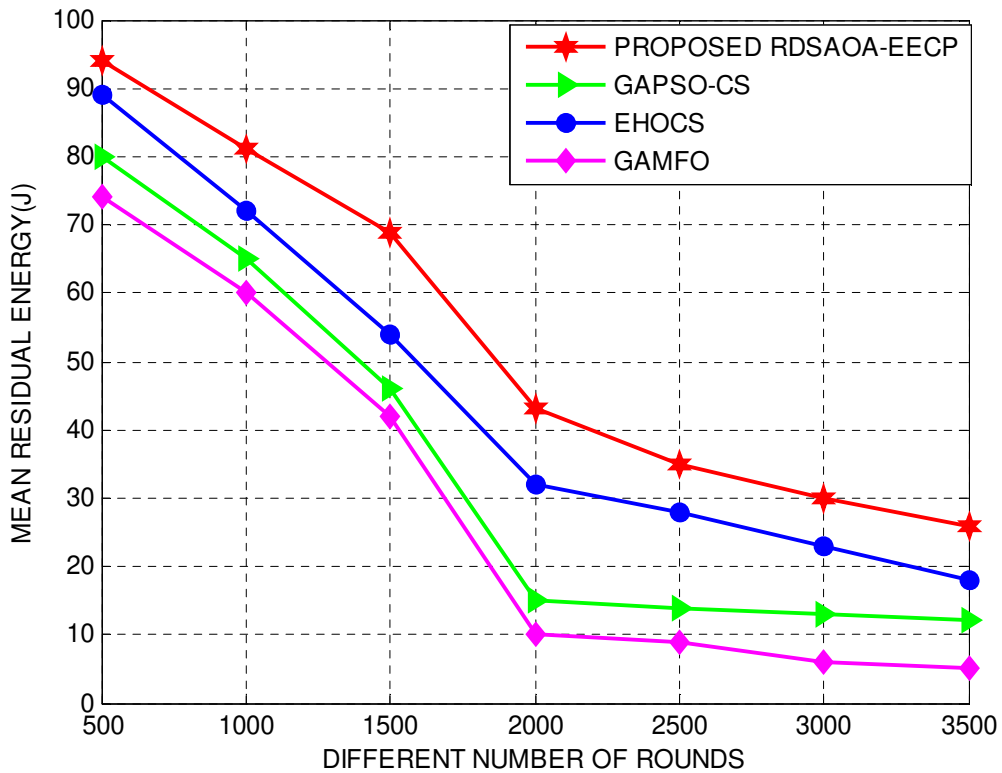
**Figure 5: RDSA OA-EECP: Number of alive sensor nodes with increase in rounds**



**Figure 6: RDSA OA-EECP: Number of dead sensor nodes with increase in rounds**



**Figure 7: RDSA OA-EECP: Throughput (bits per seconds) with increase in rounds**

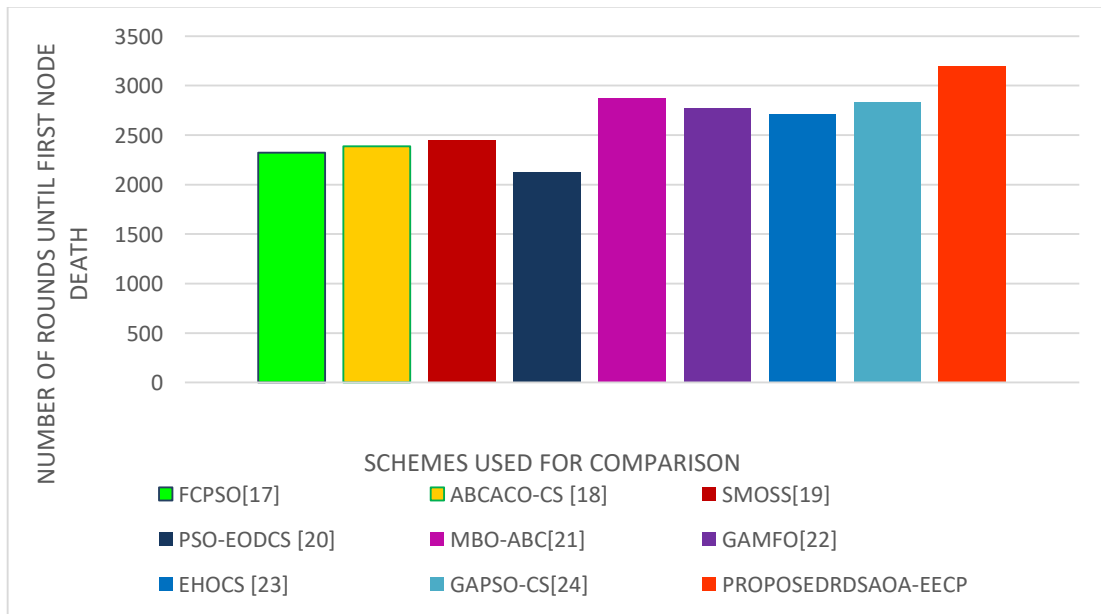


**Figure 8: RDSA OA-EECP: Residual Energy with increase in rounds**

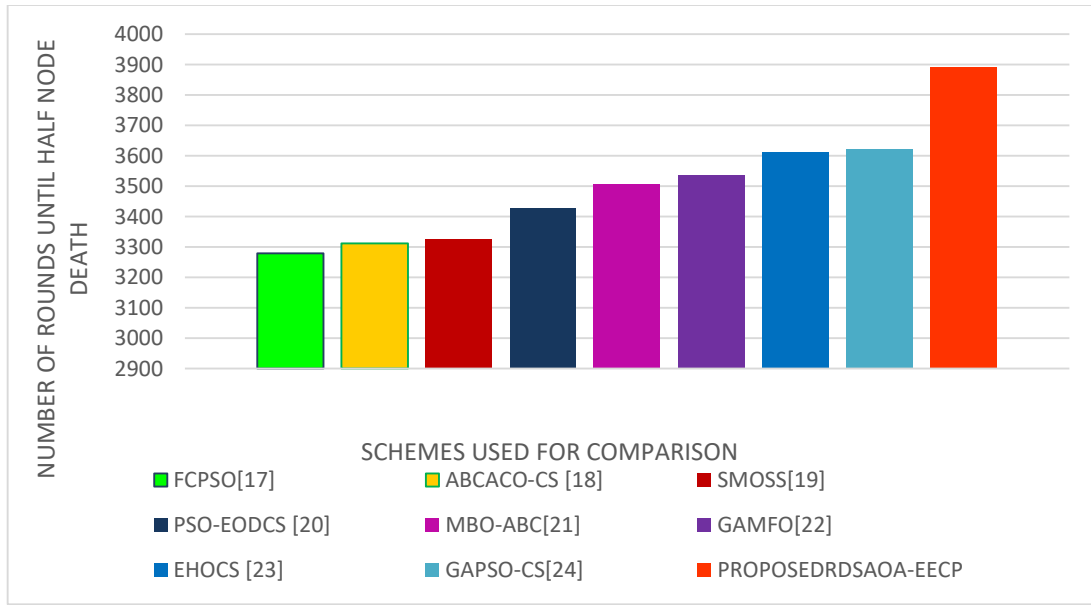


Figure 7 and 8 presents the throughput and mean residual energy of of the proposed RDSAOA-EECP protocol and the benchmarked GAPSO-CS, EHOCS and GAMFO approaches with different number of rounds. The throughput achieved by the proposed RDSAOA-EECP scheme is predominant over the compared schemes as the local search capability introduced by IOTSA into RDOA facilitates better CH selection in the network. This local search potential of IOTSA also plays a vita role in preventing local point of optimality and stagnation to the maximum level independent to the number of rounds used for implementation. Hence, the proposed RDSAOA-EECP protocol with different number of rounds improves throughput by 8.52%, 10.98% and 12.69%, excellent to the baseline GAPSO-CS, EHOCS and GAMFO approaches. The mean residual energy of the proposed RDSAOA-EECP protocol with different number of rounds also gets improved by 8.64%, 10.18% and 11.72%, excellent to the benchmarked schemes.

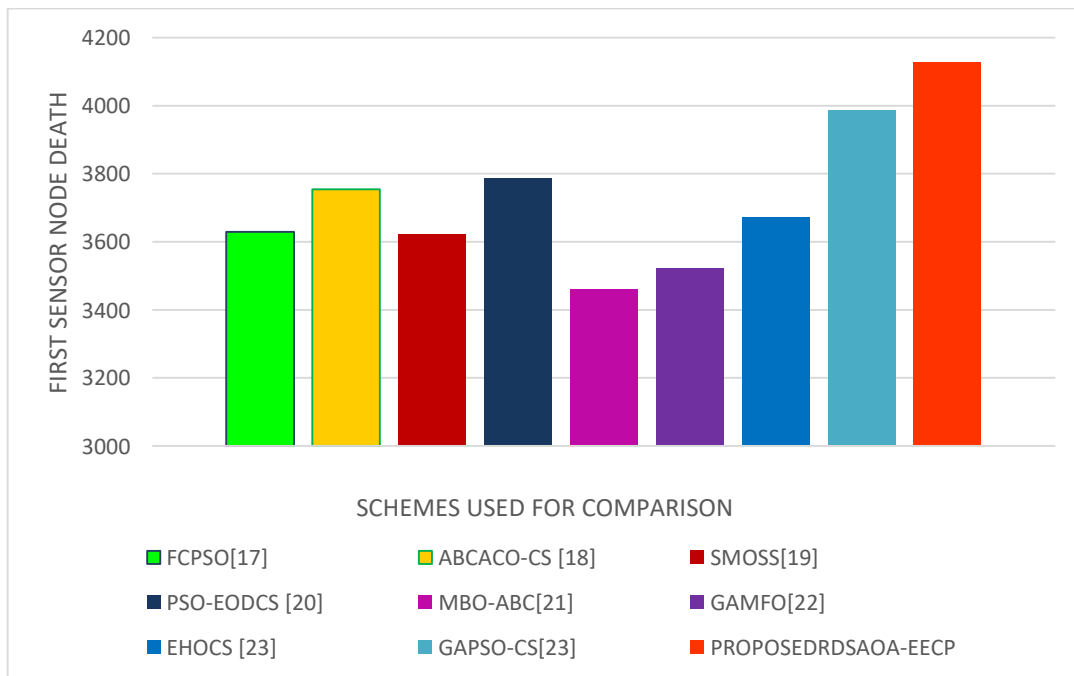
Furthermore, the proposed RDSAOA-EECP protocol and the reviewed existing works of the literature [16-23] are investigated based on number of rounds until first node death, half of the nodes death and last node death in the network. From Figure 9, it is proved that the proposed RDSAOA-EECP protocol face the first node death at the round of 3200, while first sensor node death of the compared approaches ranges between 2321-2829 rounds. This significant performance of the proposed RDSAOA-EECP protocol is mainly due to the adaptive strategy include by SA into the primitive RDOA scheme during exploitation process.



**Figure 9: RDSAOA-EECP: Number of rounds until first node death**



**Figure 10: RDSAOA-EECP: Number of rounds until the half nodes death**



**Figure 11: RDSAOA-EECP: Number of Rounds until the last node death**

Figure 10 highlights that the proposed RDSAOA-EECP protocol is determined to face to the half of the nodes death at the round 3892, while half of the nodes death of the compared approaches ranges between 3279-3621 rounds. This vital performance of the proposed RDSAOA-EECP protocol is attained with the inclusion of flexible exploitation process that varies depending on the potentiality of solutions determined during mating process. Moreover, Figure 11 explained

that the last node death of the proposed RDSAOA-EECP protocol is identified to happen at 4126 rounds, while in the compared scheme they happen at the range of 3629-3986 rounds. This remarkable capability of the proposed RDSAOA-EECP protocol is visualized mainly due to the policy adopted for maintaining the degree of exploration and exploitation to an expected level independent to the increasing number of rounds in implementation.

In addition, Table 2 is presented for portraying the mean, median and standard deviation (SD) in the number of alive nodes and normalized energy in the network maintained by the proposed RDSAOA-EECP protocol. The results depict that mean and median with respect to the number of alive nodes and normalized energy in the network achieved by the proposed RDSAOA-EECP protocol is better than the baseline approaches. This significant performance guaranteed by the proposed RDSAOA-EECP is mainly due to the multiple selection of random sample solution derived not only from the same cluster in which commander solution resides, but also based on the sample solution taken from the remaining clusters of the network. The proposed RDSAOA-EECP protocol improved the mean number of alive nodes by 5.28%, 5.82% and 6.94% with a mean normalized energy maintained by 6.32% and 7.86%, superior to the baseline schemes. Further, the proposed RDSAOA-EECP protocol improved the median number of alive nodes by 6.38% and 7.56% with a median normalized energy maintained by 4.99%, 5.93% and 6.86%, superior to the baseline schemes. In contrast, the proposed RDSAOA-EECP protocol minimized the SD in the number of alive nodes and SD in normalized energy to a maximum level of 7.62% and 8.64%, excellent to the baseline schemes.

**Table 2: RDSAOA-EECP:Mean, Median and SD in the number of alive nodes and maintained normalized energy**

<b>Compared CH selection schemes</b>	Mean number of alive nodes	Mean normalized energy	Median number of alive nodes	Median normalized energy	SD in the number of alive nodes	SD in normalized energy
Proposed RDSAOA-EECP	72.54	0.24621	64	0.16724	31.984	0.1953
GAPSO-CS,	64.92	0.21456	59	0.13282	34.582	0.2185
EHOCS	61.36	0.19828	57	0.129641	38.921	0.2384
GAMFO	58.42	0.18742	53	0.11764	39.568	0.2451

**Table 3: RDSAOA-EECP: Best, Worst, Mean, Median and SD in time complexity**

Compared CH selection schemes	Best	Worst	Mean	Median	Standard Deviation
Proposed RDSA OA-EECP	1.9874	3.4123	2.1116	2.1016	0.1912
GAPSO-CS,	2.2741	5.6487	2.2341	2.2116	0.2043
EHOCS	2.3942	12.1642	2.6164	2.6142	0.2142
GAMFO	2.5124	13.2463	3.1212	3.0241	0.3044

Table 3 demonstrates the best, worst, mean, median and Standard deviation in the time complexity of the proposed RDSA OA-EECP protocol compared to the benchmarked schemes used for investigation. The results transparently proved that the proposed RDSA OA-EECP evaluated in terms of best, worst, median and SD in time complexity is better as it included the merits of IOTSA into the roaring and fighting phases that in turn reduced the complexity incurred in CH selection. The best time complexity of the proposed RDSA OA-EECP is minimized by 6.24%, 7.83% and 8.19%, better than the compared schemes. The worst time complexity of the proposed of the proposed RDSA OA-EECP is lower by 4.86%, 5.82% and 7.49%, better than the baseline schemes. The mean time complexity of the proposed of the proposed RDSA OA-EECP is also minimized by 5.84%, 6.72% and 7.06%, better than the baseline schemes. On other hand. The median time complexity of the proposed RDSA OA-EECP is also minimized by 4.48%, 5.41% and 6.84%, better than the baseline schemes. Moreover, SD in time complexity of the proposed of the proposed RDSA OA-EECP is lowered by 4.92%, 6.38% and 8,29%, better than the baseline schemes.

## 5. Conclusions

The RDSA OA-EECP scheme was proposed for maintaining lifetime expectancy and energy stability in WSNs. It integrated the global optimization capability of the red deer optimization algorithm and with the local optimization potential of simulation annealing by replacing the roaring and fighting phase with folding and reheating phase. This inclusion of SA into RDA is helpful for establishing the balance between exploration and exploitation under CH selection for sustaining energy stability. This clustering protocol inherited the metaheuristic properties of SA and RDOA for attaining indispensable cluster heads together with the optimal base station location for the objective of enhancing energy efficiency. The simulation results of the proposed RDSA OA-EECP protocol increased network lifetime by 8.21%, 9.94% and 11.16%, predominant to the baseline GAPSO-CS, EHOCS and GAMFO approaches. Further, the energy consumptions of the proposed RDSA OA-EECP protocol with different densities of sensor nodes also get minimized by 8.92%, 9.74% and 12.92%, better than the benchmarked schemes. The results also confirmed that the proposed RDSA OA-EECP protocol improved the mean number of alive nodes by 5.28%, 5.82% and 6.94% with a mean normalized energy maintained by 6.32% and

7.86%, superior to the baseline schemes. Moreover, It was determined to improve the median number of alive nodes by 6.38% and 7.56% with a median normalized energy maintained by 4.99%, 5.93% and 6.86%, superior to the baseline schemes. As the future scope of plan, it is decided to formulate a Salp-Swarm and Moth-Flame Optimization-based CH selection scheme and compare it with the proposed RDSAOA-EECP scheme.

### **Data Availability Statement**

Data sharing not applicable – no new data generated, Data sharing is not applicable to this article as no new data were created or analyzed in this study.

### **Funding Information**

There is no funding received for this research work

### **Conflict of Interest Statement**

The authors declare that there is no competing interest

### **References**

- 1) S. Biswas, J. Saha, T. Nag, C. Chowdhury and S. Neogy, "A Novel Cluster Head Selection Algorithm for Energy-Efficient Routing in Wireless Sensor Network," 2016 IEEE 6th International Conference on Advanced Computing (IACC), Bhimavaram, 2016, pp. 588-593.
- 2) Batra, P.K., Kant, K. LEACH-MAC: a new cluster head selection algorithm for Wireless Sensor Networks. *Wireless Netw* 22, 49–60 (2016).
- 3) P. T. Karthick & C. Palanisamy (2019) Optimized cluster head selection using krill herd algorithm for wireless sensor network, *Automatika*, 60:3, 340-348.
- 4) Zhao, L., Qu, S. & Yi, Y. A modified cluster-head selection algorithm in wireless sensor networks based on LEACH. *J Wireless Com Network* 2018, 287 (2018).
- 5) A. E. Fawzy, A. Amer, M. Shokair and W. Saad, "Proposed intermittent Cluster Head selection scheme for efficient energy consumption in WSNs," 2017 34th National Radio Science Conference (NRSC), Alexandria, 2017, pp. 275-283,
- 6) A. John, A. Rajput and K. V. Babu, "Energy saving cluster head selection in wireless sensor networks for internet of things applications," 2017 International Conference on Communication and Signal Processing (ICCSP), Chennai, 2017, pp. 0034-0038.
- 7) T. Jagannadha Swamy, G. Ramamurthy and P. Nayak, "Optimal, Secure Cluster Head Placement Through Source Coding Techniques in Wireless Sensor Networks," in *IEEE Communications Letters*, vol. 24, no. 2, pp. 443-446, Feb. 2020.
- 8) D. Izadi, J. Abawajy and S. Ghanavati, "An Alternative Clustering Scheme in WSN," in *IEEE Sensors Journal*, vol. 15, no. 7, pp. 4148-4155, July 2015.

- 9) N. Shivappa and S. S. Manvi, "Fuzzy-based cluster head selection and cluster formation in wireless sensor networks," in *IET Networks*, vol. 8, no. 6, pp. 390-397, 11 2019.
- 10) S. Umbreen, D. Shehzad, N. Shafi, B. Khan and U. Habib, "An Energy-Efficient Mobility-Based Cluster Head Selection for Lifetime Enhancement of Wireless Sensor Networks," in *IEEE Access*, vol. 8, pp. 207779-207793, 2020.
- 11) Sarkar, A., Senthil Murugan, T. Cluster head selection for energy efficient and delay-less routing in wireless sensor network. *Wireless Netw* 25, 303–320 (2019).
- 12) W. Dargie and J. Wen, "A Simple Clustering Strategy for Wireless Sensor Networks," in *IEEE Sensors Letters*, vol. 4, no. 6, pp. 1-4, June 2020.
- 13) C. Tselikis, S. Mitropoulos, N. Komninos and C. Douligeris, "Degree-Based Clustering Algorithms for Wireless Ad Hoc Networks Under Attack," in *IEEE Communications Letters*, vol. 16, no. 5, pp. 619-621, May 2012.
- 14) S. Lata, S. Mehfuz, S. Urooj and F. Alrowais, "Fuzzy Clustering Algorithm for Enhancing Reliability and Network Lifetime of Wireless Sensor Networks," in *IEEE Access*, vol. 8, pp. 66013-66024, 2020.
- 15) S. H. Kang and T. Nguyen, "Distance Based Thresholds for Cluster Head Selection in Wireless Sensor Networks," in *IEEE Communications Letters*, vol. 16, no. 9, pp. 1396-1399, September 2012.
- 16) Dogani, A., Dourandish, A., Ghorbani, M., & Shahbazbegian, M. R. (2020). A hybrid meta-heuristic for a Bi-objective stochastic optimization of urban water supply system. *IEEE Access*, 8(3), 135829-135843.
- 17) Q. Ni, Q. Pan, H. Du, C. Cao and Y. Zhai, "A Novel Cluster Head Selection Algorithm Based on Fuzzy Clustering and Particle Swarm Optimization," in *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 14, no. 1, pp. 76-84, 1 Jan.-Feb. 2017.
- 18) Sengathir Janakiraman, A Hybrid Ant Colony and Artificial Bee Colony Optimization Algorithm-based Cluster Head Selection for IoT, *Procedia Computer Science*, Volume 143, 2018, Pages 360-366.
- 19) Lee, J.-G.; Chim, S.; Park, H.-H. Energy-Efficient Cluster-Head Selection for Wireless Sensor Networks Using Sampling-Based Spider Monkey Optimization. *Sensors* 2019, 19, 5281.
- 20) C, Jothi kumar; Venkataraman, Revathi. EODC: An Energy Optimized Dynamic Clustering Protocol for Wireless Sensor Network using PSO approach. *International Journal Of Computers Communications & Control*, [S.L.], v. 14, n. 2, p. 183-198, apr. 2019.
- 21) Bandi Rambabu, A. Venugopal Reddy, Sengathir Janakiraman, Hybrid Artificial Bee Colony and Monarchy Butterfly Optimization Algorithm (HABC-MBOA)-based cluster head selection for WSNs, *Journal of King Saud University - Computer and Information Sciences*, 2019.
- 22) Sharma, R., Vashisht, V. & Singh, U. eeTMFO/GA: a secure and energy efficient cluster head selection in wireless sensor networks. *Telecommun Syst* 74, 253–268 (2020).

- 23) N. Lavanya and T. Shankar, "Hybrid based Energy Efficient Cluster Head Selection using Camel Series Elephant Herding Optimization Algorithm in WSN" *International Journal of Advanced Computer Science and Applications(IJACSA)*, 11(5), 2020.
- 24) Sahoo, B. M., Pandey, H. M., & Amgoth, T. (2021). GAPSO-H: A hybrid approach towards optimizing the cluster based routing in wireless sensor network. *Swarm and Evolutionary Computation*, 60(3), 100772.
- 25) Bernardo Morales-Castañeda, Daniel Zaldívar, Erik Cuevas, Oscar Maciel-Castillo, Itzel Aranguren, Fernando Fausto, An improved Simulated Annealing algorithm based on ancient metallurgy techniques, *Applied Soft Computing*, Volume 84, 2019, 105761.
- 26) Fathollahi-Fard, A.M., Hajiaghaei-Keshteli, M. & Tavakkoli-Moghaddam, R. Red deer algorithm (RDA): a new nature-inspired meta-heuristic. *Soft Comput* **24**, 14637–14665
- 27) Amir Abbas Baradaran, Keivan Navi, HQCA-WSN: High-quality clustering algorithm and optimal cluster head selection using fuzzy logic in wireless sensor networks, *Fuzzy Sets and Systems*, Volume 389, 2020, Pages 114-144.
- 28) R. K. Poluru and L. K. Ramasamy, "Optimal cluster head selection using modified rider assisted clustering for IoT," in *IET Communications*, vol. 14, no. 13, pp. 2189-2201, 11 8 2020.
- 29) Kardi, A., Zagrouba, R. RaCH: A New Radial Cluster Head Selection Algorithm for Wireless Sensor Networks. *Wireless Pers Commun* 113, 2127–2140 (2020).
- 30) Amutha S Kannan B Kanagara J M, Energy-efficient cluster manager-based cluster head selection technique for communication networks. *Int J Commun Syst*. 2020; 33:e4427.
- 31) Sharad, Er. Navpreet Kaur & Inderdeep Kaur Aulakh (2020) Evaluation and implementation of cluster head selection in WSN using Contiki/Cooja simulator, *Journal of Statistics and Management Systems*, 23:2, 407-418.
- 32) Amutha, S., Kannan, B., & Kanagaraj, M. (2021). Energy-efficient cluster manager-based cluster head selection technique for communication networks. *International Journal of Communication Systems*, 34(5).
- 33) S. Vasundra, D Venkatesh "Performance Evaluation of Routing Protocols For Voice and Video Traffics" *AJCST- Asian Journal of Computer Science & Technology*, ISSN: 2249-0701, Vol-7, N0-3, Dec-2018,
- 34) D.Venkatesh, A Subramanyam, "Lifetime Enhancement of a Node Using I—Leach Protocol in WSN", *Advances in Intelligent Systems and Computing* , pp 113-120, march 2018.