OPTIMAL CLUSTERING AND ROUTING FOR WIRELESS SENSOR NETWORK BASED ON CUCKOO SEARCH

Soumitra Das¹, Barani S², Sanjeev Wagh³ and S.S. Sonavane⁴

¹Research Scholar, Dept. of Computer Science and Engineering, Sathyabama University, Chennai
²Dept. of Electronics and Control Engineering, Sathyabama University, India
³Dept. of Information Technology, Government College of Engineering, Karad
⁴Dr. D.Y. Patil School of Engineering, Pune

ABSTRACT

In this research work, the egg laying radius of cuckoo search algorithm is used to create a cluster and then search for the optimum node based on multiobjective genetic algorithm with pareto ranking, so that the data can be forwarded to the sink. The primary focus is on the two performance metrics parameters, one is the maximization of network lifetime and other is the minimization of delay. For maximizing the network lifetime parameter, the overlapped target sensing by many sensors is wastage of energy by two or more sensors, where the same task can be done by one sensor. To overcome this problem, the sequence set cover methodology is used. For minimization of delay parameter, the sleep-wake scheduling mechanism will be considered, but substantial delays are introduced as transmitting node needs to wait for its next-hop relay node to wake up. These delays can be taken care by developing any cast based packet forwarding schemes where individual node forwards a packet to the first neighboring node that wakes up among multiple candidate nodes. This any cast forwarding schemes minimizes the expected packet-delivery delays from the sensor nodes to the sink node. The introduced work will perform energy proficient routing with an objective to improve the network life, packet loss ratio and overall network throughput. The proposed algorithm was simulated in MATLAB and compared with LEACH algorithm. The results show that our proposed algorithm is superior for prolonging the network lifetime, minimizing the packet loss and increasing the throughput.

KEYWORDS:


1. INTRODUCTION

Wireless Sensor Networks (WSNs) consist of hundreds of thousands of micro sensor nodes, which are connected by a wireless medium. These sensor nodes are constrained in power, memory and computational capabilities. To extend the network life time, we normally use energy aware approaches such as multihop communication, in-network data processing, data fusion, and sleep& wake up methods [1]. Clustering based routing techniques are mostly used in WSN applications because of their divide and conquer strategy. As per [2-3], the key element to prolong the network lifetime can be done by balancing the dissipated energy among the available nodes. Hence efficient data clustering and routing techniques should be used to prolong the network lifetime [4].
The motive behind this research work is to increase the lifespan of the WSNs as sensor nodes are limited in energy resource and there is no option to recharge them on the go. The main objective of this article is to construct a new energy efficient routing protocol based on Cuckoo Search (CS) and multiobjective genetic algorithm with pareto-ranking techniques.

2. Preliminaries

2.1 A Brief Overview of Cuckoo Search

Cuckoo search is a meta–heuristic optimization method for solving problems to provide optimal solution. Cuckoo search was first introduced by Xin-She Yang and Suash Deb in 2009[1]. The hypothesis of cuckoo search was motivated by the reproducing conduct of types of bird called cuckoo.

The cuckoo birds lay their eggs in the homes of other host birds. At the point when the host birds find the eggs are not their own, it will either discard these outsider eggs or essentially surrender its home and construct another home somewhere else. Cuckoo search admired such reproducing conduct, and in this way can be related for different optimization issues.

The Cuckoo search works on three idealized energetic standards:

1. One egg is laid by a cuckoo at a particular time, which is placed at anarbitrarily selected nest.
2. The finest nest with extraordinary caliber of eggs gets carried over to the subsequent generation.
3. The quantity of existing host nests are fixed, and the host bird with a probability $Pa \in [0,1]$ discovers the egg laid by a cuckoo.

Taking into account the above three guidelines, the likelihood is that the host bird can either discard the egg or surrender the nest and assemble a totally new nest.

2.2 A Brief Overview of Multi-objective Genetic Algorithm with Pareto Ranking

The multi-objective optimization is an area which encompasses additional objective function to be optimized concurrently. Multi-objective optimization method has been applied in various areas wherein optimal decisions are required in the presence of trade-offs between two or more differing objectives [13-14]. A few ideas of multi-objective improvement issues considered are as below.

2.2.1 A Multi-Objective Decision Problem:

Let $s = \{s_1, \ldots, s_x\}$ be the decision variable vector of $x$-dimension, in the solution space $S$. A vector $s^*$ is prerequisite to decrease a known set of $M$ objective functions $z(s^*) = \{z_1(s^*), \ldots, z_k(s^*)\}$. The solution space $S$ is restricted by a sequence of constraints, as $f_i(s^*) = c_i$ for $i = 1, \ldots, n$, and is bound by the decision variables.
2.2.2 DOMINANCE:

Let’s say vector \( a = \{a_1, a_2, a_3, \ldots, a_m\} \) dominates vector \( b = \{b_1, b_2, b_3, \ldots, b_m\} \) if and only if \( a \) is partially less than \( b \).

2.2.3 PARETO OPTIMALITY:

A solution \( Z_u \in Z \) is assumed to be pareto optimal if and only if there is not at all \( Z_v \in Z \) for which \( v = f(Z_v) = \{v_1 \text{ to } v_m\} \) dominates \( u = f(Z_u) = \{u_1 \text{ to } u_m\} \).

2.2.4 PARETO OPTIMAL SET AND FRONT:

Let the non-dominated set \( D \) where \( D \subseteq Z \), is defined as \( Z_p = \{a \in D | a \text{ is non-dominated regarding } Z\} \).

The objective function values in the objective space are \( S_p = F(Z_p) = \{f(a) | a \in Z_p\} \).

Where \( Z_p \) = Pareto optimal set

\( S_p \) = cohere pareto optimal front

The final objective of a multi-objective optimization algorithm is to categorize solutions in the Pareto optimal set. So, a useful method for multi-objective optimization is to discover a set of solutions that signify the Pareto optimal set as much as possible.

3. RELATED WORK

Historically nature inspired algorithms have been extensively used in computational intelligence. Many researchers are working in this area and have developed different algorithms serving different purposes. One of the most interesting areas where these algorithms can be used widely is clustering in wireless sensor networks [5]. Still limited energy of node is one the main obstacles; therefore research is going towards the nature inspired algorithms which tend to give better solutions as compared to classical algorithms. This research focuses on cuckoo search for the development of clustering algorithms and multiobjective genetic algorithm for optimum node selection. Cuckoo search was first presented by Xin-She Yang et al. in 2009[6]. The theory of cuckoo search was inspired by the species of bird called cuckoo [7]. The concept of laying eggs and breeding of cuckoos has triggered the concept of basic algorithms using cuckoo search. Dhivya et al [8] has suggested a clustering algorithm based on cuckoo search. The authors of this article claims that their implemented technique increases the lifetime of the network by aggregating active nodes to about 15 percent.

Bhatti et al.[9], has proposed cuckoo based energy effective routing in WSN to improve the network throughput and network lifetime without increasing the congestion over the network. The authors used fuzzy systems to modify PEGASIS and named it as fuzzybased PEGASIS whereincuckoo search algorithm is then used for optimization process. The simulations were carried out in matlab simulation software. The parameters considered for simulations are 100 sensor nodes deployed in the 100 by 100 meter square field. The sensor nodes were equipped with 45 meter communication range. The results were measured with respect to number of nodes
The author claims that earlier system’s lifetime was 76% and the proposed system has a lifetime of 92% with an achievement of 16% of system lifetime. Dhivya et al [10], the cuckoo based particle approach (CBPA) was proposed to increase energy efficiency in WSN and multimodal functioning. Using Cuckoo search, cluster head is selected and clusters are formed among the sensor nodes. They measured performances such as energy minimization, energy conservation of WSN and lifetime maximization. On comparison of this proposed CBPA with the standard LEACH and HEED protocol it was proved that exhibited simulation results were comparable largely because of prime search process in cluster formation and also distribution of suitable tracks in transmission of the data sensed. M. Aslam et al [2012] [11] have proposed a Centralized Energy Efficient Clustering (CEEC) routing protocol. The CEEC was designed for three level heterogeneous WSN. CEEC can be applied to the WSNs with multi-level heterogeneity. In CEEC, entire network area is separated into three equivalent sections such that nodes with equal energy are located in same section. CEEC has improved throughput and network lifetime. A.A. Khan et al. [2012][12] have proposed Heterogeneity-aware Hierarchical Stable Election Protocol (HSEP) with two protocols. In clustering protocols farther the CS is from BS more is the amount of energy consumed during transmission. Proposed protocol is designed to lower the amount of energy consumed during transmission from CH to BS. Simulation proved that the proposed protocol increases network lifetime and stability period as compared to other protocols.

4. PROPOSED WORK

The proposed approach uses the concept of cuckoo search algorithm to create a cluster of a WSN. The cluster formation is based on the cuckoo’s Egg Laying Radius (ELR). Each cluster consists of two types of nodes namely a trigger node and optimum nodes, where the trigger nodes are responsible to form a cluster. Usually the triggered node is selected based on the highest residual energy and which is relatively close to the event. These trigger nodes are also termed as Cluster Head (CH). Once the cluster is formed around the trigger node considering the ELR. Whenever any event occurs, the data forwarding is taken care by the optimum nodes. Selection of optimal node is based on multiobjective genetic algorithm with pereto ranking concept. The flowchart of the proposed work is presentding figure 1.

![Flow chart of the proposed system](image-url)
4.1 CLUSTER FORMATION USING CUCKOOS EGG LAYING RADIUS (ELR)

One of the natural tendencies of cuckoos is to lay eggs within a maximum distance away from their habitat. This distance from the habitat to the host bird’s nest is termed as “Egg Laying Radius (ELR)” as shown in figure 2 and cuckoos normally lay their eggs within this ELR. When an event occurs (say earthquake, fire, etc.) a node which is having the highest residual energy and which is near to the event is selected as a trigger node. Considering this trigger node as the center of the cluster and ELR as the radius, a cluster is formed.

To explain an optimization problem, the problem variables should be in the form of a collection, which is termed as “habitat” in Cuckoo Optimization Algorithm. In a \( N_{var} \)-dimensional optimization problem, the habitat is represented as a collection of \( 1 \times N_{var} \), which is the present living location of the cuckoo. The collection is stated as

\[
\text{Habitat} = [x_1, x_2, \ldots, x_{N_{var}}]
\]

Where \([x_1, x_2, \ldots, x_{N_{var}}]\) are floating point numbers.

Another natural phenomenon of cuckoos is that, each cuckoo lays about five to twenty eggs. This range is termed as the upper limit \( Var_{hi} \) and lower limit \( Var_{low} \) of the egg dedication to each cuckoo at different iterations. Therefore ELR is calculated as

\[
ELR = \alpha \times \frac{\text{current number of eggs}}{\text{total number of eggs}} \times (Var_{hi} - Var_{low})
\]

Where \( \alpha = \text{integer} \) handles the maximum value of ELR.

![Figure 2: The Egg Laying Radius with six eggs.](image)
4.2 Finding of Optimum Node Using Multi-Objective Genetic Algorithm

For the discovery of optimum node, the multi-objective genetic algorithm based on Pareto ranking is taken into consideration. The efficiency of multi-objective genetic algorithm mainly depends on the selection of fitness functions. Here for the selection of fitness functions say $F_1$ and $F_2$ are considered, where $F_1$ takes care for maximization problems and $F_2$ takes care for minimization problems. The same can be compared with WSN, where we need to maximize ($F_1$) the lifetime of the network and minimize ($F_2$) the delay in data transmission. In this we need to take care of more than one objective say ($F_1$) and ($F_2$), so we need to use multi-objective genetic algorithm to find the optimum node, as the multi-objective genetic algorithm gives very good performance to solve the maximization and minimization problems.

Following analysis are carried out on ($F_1$) in 4.2.1 and ($F_2$) in 4.2.2 for fitness so that ($F_1$) takes care for maximization problem and ($F_2$) for minimization problem.

4.2.1 Selection of ($F_1$) for Maximization

The objective here is that, we need to set the parameters of ($F_1$) in such a way that the output of ($F_1$) is maximized. There we need to prove that

$$F_1 = \text{Maximize} \ (C_1 + C_2 + \cdots + C_n)$$

Where $C_n = \text{Sequence Set Cover (SSC)}$.

The SSC is represented as $C_i = \{(N_1r_1e_1), \ldots \ldots (N_ie_i)\}$, where $i < \infty, N_i$ is using sensing range $r_i$ to sense the target $t_i$ with energy $e_i$.

Suppose we deploy $n$ sensors $N_1, N_2, N_3, \ldots \ldots, N_n$ to sense a total number of target $m$ as $(t_1, t_2, \ldots \ldots, t_m)$, where each node has multiple sensing range $r_1, r_2, r_3, \ldots \ldots, r_p$ and energy consumption $e_1, e_2, e_3, \ldots \ldots, e_p$ for each node for each sensing range. Assuming that the maximum energy associated with each node is $E$.

Our main objective was to find the efficient sequence of node and sensing range so that utilization of power is maximized and at the same time all targets should be covered.

For example let’s consider three sensor nodes $N_1, N_2$ and $N_3$ as shown in the figure 3. Node $N_1, N_2$ and $N_3$ has sensing range $r_1, r_2$ and $r_3$ respectively which are overlapped with each other. As we can see from figure 3, there are two targets $t_1$ and $t_2$ which come under the range of more than one sensor. Suppose $t_1$ is the first target to sense which comes under the range of $N_1$ and $N_2$ sensors. In this case for sensing the same target by $N_1$ and $N_2$ both will utilize their energy which is waste of energy and will lead to reduce the network lifetime. To overcome this problem we assumed the following techniques:

First we will try to identify dissimilar sequence of nodes and sensing ranges to cover the set of targets. Therefore $C_i = \{(N_1, r_1, e_1), \ldots \ldots (N_i, r_i, e_i)\}$, where $i < \infty, N_i$ is using sensing range $r_i$ to sense the target $t_i$ with energy $e_i$.
Continue with figure 3, we observe that target $t_1$ is in the range of $N_1$ and $N_2$ and target $t_2$ is in the range of $N_2$ and $N_3$. In this scenario we need to calculate two Sequence Set Cover as $C_1$ and $C_2$. Therefore $C_1 = \{(N_1, r_i, e_j), (N_2, r_j, e_j)\}$ and $C_2 = \{(N_2, r_j, e_j), (N_3, r_k, e_k)\}$. The residual energy of $N_1, N_2$ and $N_3$ are $(E - e_j), (E - e_j)$ and $(E - e_k)$ respectively.

Now suppose we assume each Sequence Set Cover as a single round in WSN, then the network lifetime will be maximum if and if only when the network can sustain for the more number of rounds. In other word we can say that lifetime of network will be more if $C_1 + C_2 + \cdots C_n$ is maximum. So, our first objective function

$$F_1 = \text{Maximize} \ (C_1 + C_2 + \cdots C_n)$$

Subject to:

i. Energy spent by every sensor node is $\leq$ total initial battery energy of the sensor nodes.

ii. If the cover $k$ consists of a sensor node, then exactly one of its $P$ sensing ranges are positioned.

iii. This assures that each of the targets $t_j$ gets covered by each set $c_k$.

### 4.2.2 SELECTION OF ($F_2$) FOR MINIMIZATION

The issue of delay minimization is an instance of the stochastic shortest route issue, wherein the node, which holds the packet, assesses the present state. The resultant disturbance relates to the cost which needs to be minimized. In the event that there is just a single source creating the event-reporting packets, the end-to-end disturbance of the first packet is figured as an element of any cast strategy $(X, Y)$ and furthermore, the rest the sleep-wake schedules policy $sp$. The any cast strategy $(X, Y)$ is defined by forwarding candidate node sets $(X)$ and priority of node $(Y)$. The responsibility of the forwarding node set, which is a set of candidates nodes is to forward a packet at node. Matrix $X$ contains all the forwarding set of entire nodes denoted by

$$X = [x_{kl}, k = 1, \ldots, M, l = 1, \ldots, M]$$

Where $x_{kl} = 1$ if $l$ is in node $k$'s forwarding set

$$x_{kl} = 0 \text{ otherwise}$$
The priority of matrix of nodes are represented as

\[ Y = [y_{kl}, k = 1, \ldots, M, l = 1, \ldots, M] \]

Assume \( n_1 > n_2 > \ldots > n_h \) to be the order of nodes that continually transmit the packet from the source node \( s \) to base stations.

The order is irregular on the grounds that at each hop, the primary node in the forwarding set that awakens is chosen as the next-hop node.

If the packet arrives at sink \( s \) after \( H \) hops, then we have \( n_h = s \) for \( h \geq H \). Let \( d_i(s, X, Y) \) be the expected one-hop delay at node \( i \) under the anycast strategy \( (X, Y) \), i.e. the expected delay from the time the packet arrives at node \( i \) to the time it is forwarded to the next-hop node.

Then, the end-to-end delay for packet \( D_i(s, X, Y) \) from node \( i \) to sink \( s \) is expressed as

\[ D_i = (s, X, Y) = E\left[ \sum_{k=0}^{\infty} d_{ik}(s, X, Y) \right] \]

So, our second objective function \( F_2 \) is to minimize \( D_i = \text{Minimize}(E[\sum_{k=0}^{\infty} d_{ik}(s, X, Y)]) \).

After finding both the objective function \( F_1 \) and \( F_2 \), now we need to discover the pareto ranking. In this calculation we try to locate the rank of the points, which is equivalent to the quantity of points by which it is dominated in the present population. Pareto ranking algorithm is demonstrated in algorithm 1.

Algorithm 1: Pareto Ranking Algorithm

\[ \text{PR}(p_{np}, d_n) \] // \( p_{np} \): total number of points in the population, \( d_n \): number of dimensions

1. \{ \( T_1[p_{np}] = 0 \), \( T_2 = 0 \), \( T_3 = 0 \). \} // Initialization of variables & array
2. \( A_1[p_{np}][d_n] = \text{Population} \)
3. for \( i = 0 \) to \( (p_{np} - 1) \)
4. \{ for \( j = 0 \) to \( (p_{np} - 1) \)
5. \{ if \( j != i \)
6. for \( k = 0 \) to \( (p_{np} - 1) \)
7. \{ if \( f_1(A_1[j,k]) \leq f_2(A_1[i,k]) \)
8. if \( f_1(A_1[j,k]) < f_2(A_1[i,k]) \)
9. \{ \( T_2 = T_2 + 1 \) \}
10. else
11. \{ \( T_3 = T_3 + 1 \) \}
12. if \( (T_2 + T_3) == d_n \)
13. \{ \( T_1[i] = T_1[i++] \) \}
14. \( T_2 = T_3 = 0 \)
15. \} \}
Once the front is discovered, we can say that the optimal node will be a member of the front and then we can apply genetic algorithm to this front i.e. encoding, cross over, mutation. The adaptive strategy algorithm is demonstrated in algorithm 2.

Algorithm 2: Adaptive Strategy Algorithm

1. \( N_b[N]=0 \)  // \( N_b \) is an array //
2. Int \( s_1=0, s_2=0; \)
3. Float \( z=0.0, q=0.0; \)
4. Enter number of nodes \( N \)
5. For(\( i=0 \) to \( N-1 \))
6. \( \{ \) Enter \( N_b[i] \)  // \( N_b[i] \) is the no of mutation done by an \( i^{th} \) mutation operator in a generation //
7. For( \( j=0 \) to \( (N_b[i]-1) \))
8. \( \{ \) \( p[i,j]=Mx[f(rank(j)),f(mut(Rk(j)))]-f(Rk(j)); \)
9. \( \} \)
10. for( \( i=0 \) to \( (N-1) \))
11. \( \{ \) for( \( k=0 \) to \( (N_b[i]-1) \))
12. \( \{ \) for( \( j=0 \) to \( (N_b[i]-1) \))
13. \( \{ \) \( s_1=s_1+p[k,j]; \)
14. \( z=(s_1/N_b[k])+z; \)
15. \( s_2=s_2+p[i,k]; \)
16. \( s_1=0; \)
17. \} \)
18. \( q=(s_2/N_b[i]); \)
19. \( t[i]=q/z; \)
20. \} \)
21. for( \( i=0 \) to \( (N-1) \))
22. \( \{ \) \( R[i]=t[i]*(P_{mut}-N) \)
23. \} \}

/**** Function for maximum ****/

1. \( Mx(int a, int b) \)
2. \{ \)
3. \( \) If(\( a>=b \))
4. \( \) Return \( a \)
5. \( \) Else
6. \( \) return \( b \)
7. \} \}

/**** Function for ranking ****/

1. \( Rk(int y) \)
2. \{ \)
3. \( \) Int\( z[n][2]=0, temp=0, c=0; \)
4. \( \) For(\( i=0 \) to \( (n-1) \) )
5. \( \} \)
6. \( Ar[i][0] = r[i] \);
7. \( Ar[i][1] = i \);
8. }
9. For\((i=0\) to \((n-2))\)
10. {
11. For\((j=(i+1)\) to \(n-1)\)
12. {If\((z[i][0] < z[j][0])\)
13. {
14. Temp = \(z[i][0]\);
15. \(Ar[i][0] = z[j][0]\);
16. \(Ar[j][0] = Tp\);
17. \(Tp = z[i][1]\);
18. \(z[i][1] = z[j][1]\);
19. \(z[j][1] = Tp\);
20. }}
21. \(C = z[y][1]\);
22. Return \((a[c])\);
23. }

After some iteration, we will notice that the nodes of the front have become fixed. Now we can choose any member of node as an optimal node. Now once this is done the data can be sent through the optimal to the sink.

5. PERFORMANCE ANALYSIS OF PROPOSED SYSTEM

The execution of the proposed algorithm has been simulated in Matlab. Table 1 lists the simulation parameter to create a simulation environment.

5.1 SIMULATION PARAMETERS

<table>
<thead>
<tr>
<th>General Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area</td>
<td>500 ( \times ) 500( \text{M}^2 )</td>
</tr>
<tr>
<td>No. of nodes</td>
<td>500</td>
</tr>
<tr>
<td>Initial Energy</td>
<td>100 Joules</td>
</tr>
<tr>
<td>Data Packet Size</td>
<td>40 bytes</td>
</tr>
<tr>
<td>Sleep Power</td>
<td>0.0006 mW</td>
</tr>
<tr>
<td>No. of rounds</td>
<td>100</td>
</tr>
<tr>
<td>No. of Nest</td>
<td>100</td>
</tr>
<tr>
<td>No. of eggs in the nest</td>
<td>1-5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GA Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. Generation</td>
<td>500</td>
</tr>
<tr>
<td>Population size</td>
<td>200</td>
</tr>
<tr>
<td>Length of chromosomes</td>
<td>100</td>
</tr>
<tr>
<td>Mutation rate</td>
<td>0.007/0.1</td>
</tr>
<tr>
<td>Crossover rate</td>
<td>0.7</td>
</tr>
</tbody>
</table>

Table 1: Simulation Parameters
5.2 Results and Discussions

The hybrid model consisting of CS and Multiobjective genetic algorithms yields better results as compared to LEACH in terms of better execution of network lifetime, packet loss ratio, and throughput.

![Figure 4. Network Lifetime](image)

Figure 4 shows the graph of residual energy of network with respect to the time in milliseconds (Network lifetime) for proposed algorithm and LEACH. From Figure 4 it is evident that our proposed algorithm has the capability to keep alive the network for an extended period as compared to LEACH.

![Figure 5. Packet Loss Ratio](image)

Figure 5 shows the average Packet Loss Ratio (PLR) with respect to the time in milliseconds for the proposed algorithm and LEACH. Figure 5 shows that the proposed algorithm is better in terms of average PLR (which is less than LEACH).
Figure 6 illustrates the gained efficiency in terms of throughput of the network. From Figure 6, it is evident that our proposed algorithm is superior to LEACH in terms of improved throughput.

6. CONCLUSION

In this paper, a novel opportunistic routing protocol has been presented, which introduces hybrid method based on cuckoo search and multiobjective genetic algorithm with pareto ranking for clustering and routing. The proposed algorithm is energy efficient as it considers the effect of each transmission and reception of packets on node's total energy. The simulation and performance analysis has been done by comparing proposed algorithm with LEACH algorithm. The results show that proposed algorithm has good performance in the presence of nodes. The proposed method optimizes the network lifetime, packet loss ratio and throughput in WSN. In the future we can consider more parameters and metrics components to improve network performance by considering other properties of the WSN, cuckoo search and multiobjective genetic algorithm. The result also proves that the proposed method is better than LEACH algorithm and increases the lifetime of the network.

REFERENCES


