8.1 Introduction to Maximal Independent Set:

In graph theory concepts, independent set [106] is a set which contains non-adjacent elements. In the graph, the elements which are not having edges between them are called as non-adjacent elements. An independent set which is having maximum number of such non-adjacent elements is called as Maximal Independent Set (MIS). In network view, the non-adjacent elements are one which are not directly connected with one another (i.e.) no direct communication link.

Figure 8.1: Sample graph

In Figure 8.1 (a), a sample network graph is shown. In Figure 8.1 (b), an independent set of the same graph is shown and Figure 8.1 (c) depicts the maximal independent set of non-adjacent nodes.
8.2 Coverage Pattern in WSN:

Coverage [107] is the process of monitoring the physical phenomena around the sensor location. The coverage model reflects the sensor capability and its quality in measuring the physical phenomena.

8.2.1 Coverage Model:

Based on the pattern, the coverage model can be classified into two types: (i) Disk model and (ii) Directional model. Figure 8.2 depict the disk coverage and directional coverage models. In disk model as shown in Figure 8.2 (a) all the points within the sensing radius $R_s$ of a node will be sensed by the nodes. But in directional model, the sensor node will cover the points in a particular direction only which is up to its length of $R_s$ as shown in Figure 8.2 (b).

![Disk coverage model](image-a)

(a) Disk coverage model

![Directional coverage model](image-b)

(b) Directional coverage model

Figure 8.2: Coverage models
8.2.2 Coverage Types:

Coverage type represents the level of coverage. It can be classified into three types [108], they are: (i) Area coverage, (ii) Point coverage and (iii) Barrier coverage.

**Area coverage:** The entire Region of Interest (ROI) is under monitoring by the sensor nodes. It is also referred as full coverage in literature.

**Point coverage:** In the entire ROI, certain key location will be more important than others. The sensor nodes can be scheduled in such a way that these locations (points) should alone be monitored continuously.

**Barrier coverage:** Whenever an event occurs in the surrounding environment of a node, till the end of the event its activities should be monitored continuously. This kind of coverage is called as barrier coverage.

8.2.3 Redundant Data in a Cluster:

In the conventional clustering architecture, the Cluster Members (CMs) are closely located with each other. The data generated by these members will be more redundant in nature. These redundant data are combined into a single piece of data at the Cluster Heads (CHs) using appropriate aggregation techniques. Aggregation cost is also one of the major factors of energy consumption in CHs. The best way of solving this redundant data problem is by avoiding the generation of redundant data than aggregating it.

In this chapter, a new clustering algorithm Energy Efficient Coverage aware Data Collection (EECDC) is proposed for solving the redundant data generation in WSN. In conventional clustering algorithms, a cluster will contain the nodes geographical closer to
each other. In EECDC logical clusters are used to avoid redundant data. A logical cluster is one, which contains non-adjacent sensor nodes in the network. The nodes whose sensing radius $R_s$ is not overlapping with each other are classified as non-adjacent nodes. This logic cluster is inspired based on the MIS concept in graph theory. Figure 8.3 shows three sensor nodes A, B and C with sensing radius $R_s$. Since A’s and B’s sensing or coverage range overlaps with each other A and B can’t be accommodated in the same cluster set. But C’s coverage range do not overlap with A and B, so C can be added to the cluster set where A or B is there.

Figure 8.3: Identifying non overlapping members
8.3 Energy Efficient Coverage aware Data Collection (EECDC):

EECDC has two phases: (i) Cluster Formation phase and (ii) Data Collection phase.

8.3.1 Cluster Formation phase:

In this phase, maximum number of non overlapping nodes will be accommodated in a logical cluster. A node $S_i$ will be randomly selected from the set $S = \{S_1, S_2, S_3...S_N\}$ and added in a new cluster set $C_x$ where $N$ represents the maximum number of nodes in the network. Then the nodes whose coverage or sensing areas do not overlap with each other will be added in the cluster set $C_m$. After the nodes are added in the new cluster set, these will be removed from the set $S$. Again the above explained process continues till all the nodes get accompanied within a cluster. Finally, each cluster set will be containing the nodes whose coverage areas are non-overlapping and the redundant data generation is avoided within a cluster. Then within a cluster set, the member nodes weight $w$ value is computed based on the Equation (8.1).

\[
    w = \alpha.RE + \beta \cdot \frac{1}{TD}
\]  

(8.1)

The weight $w$ of a node depends on residual energy ($RE$) and total sum of distance with all other members ($TD$) in a cluster set. $\alpha$ and $\beta$ in Equation (8.1) are the co-efficient values of $RE$ and $TD$ respectively. $\alpha$ and $\beta$ varies between 0 and 1. The highest weight node is elected as CH in a cluster set.
All the nodes will be informed about its status like whether it is CH or CM. For a CM node the details such as its membership to which cluster and Time Division Multiple Access (TDMA) schedule will be communicated directly by the BS. Along with that, a token will be transmitted to the CH of first cluster set to be activated in the current round. The token contains the details such as next cluster sets to be activated in sequence. The cluster set which is having the token will alone be involved in monitoring the environment. The other cluster sets will be in sleeping state to conserve its energy and to avoid redundant data.

8.3.2 Data Collection phase:

A threshold value $T = s.E_{init}$ is defined in all the CHs, where $s$ is the user defined value between 0 and 1 and $E_{init}$ is the initial energy of the CH when it get elected as CH. After receiving the token from the BS or from previous CH, the CH for the current cluster set will check its energy whether it is greater than the threshold value $T$. If the residual energy of CH is greater than $T$, the data collection phase continues, otherwise the token will be transmitted to the BS to initiate re-clustering in the network.

The CMs in a cluster communicates its data to the CH during its time slot. The CH after receiving the data from its CM has to aggregate the data but the pure aggregation followed for the other proposed algorithms cannot be applied in EECDC. The pure aggregation is one which transforms all the received data messages from the CMs into a single piece of information whose length is equal to the data message sent by a CM. The pure aggregation is suitable for a cluster whose CMs data are more redundant. But in the case of EECDC, data from the CMs are unique and non-redundant. So it should be
aggregated accordingly where the uniqueness of the data should not be lost. So the aggregation ratio is followed in the CHs for aggregating the CM data. Equation (8.2) depicts the length of the aggregated data.

\[ L_{agg} = L_{rec} + (L_{rec} \cdot R_{agg} \cdot m) \]  

(8.2)

The CHs of the respective cluster set communicated the aggregated data message of length \( L_{agg} \) to the BS. \( L_{rec} \) is the length of the data bits from CMs, \( R_{agg} \) is the aggregation ratio and \( m \) is the number of nodes in a cluster set.

At the end of the data collection phase, the token held by the current CH will be transferred to the next CH mentioned in the token. Thus, in EECDC only one cluster set will be in active state during a communication round so that other cluster set member conserve their remaining energy level. But due to this way of activation, some regions in ROI may be uncovered during a communication round, but it will be covered during the subsequent cluster set activations. So at discrete time interval all the points in the ROI will be covered in EECDC execution. EECDC is more suitable for applications where continuous monitoring of all the points in the ROI is not required. Figure 8.4 depicts the EECDC algorithm.

**8.3.3 Time Complexity of EECDC:**

In EECDC, during the cluster formation phase a node is selected randomly and it checks the non-overlapping nodes from the list and forms a cluster. Thus \( n(n-1) \) processing occurs. Then in each cluster set, the weight of the member node is computed which takes \( mk \) processing time.
\[ T(n) = n(n - 1) + mk \quad (8.3) \]

\[ T(n) = n^2 - n + mk = O(n^2) \quad (8.4) \]

Where \( n \) = number of nodes in the network, \( K \) = number of cluster set, \( m \) = number of members

---

**EECDC Algorithm:**

1. \( S \) = set of all sensor nodes
2. \( MI \) = Maximal Independent Set
3. \( \) while \((S!={})\) //empty independent set
4. \( \) status=true
5. \( \) Initialize \( MI_j = \{ \}\)
6. \( \) select a random node \( s_i \) from \( S \)
7. \( \) add node ‘\( s_i \)’ to set \( MI_j \)
8. \( \) for \( h=1:|S| \)
9. \( \) for \( g=1:|MI_j| \)
10. \( \) if \( d(s_g, s_h) > R_S(s_g) + R_S(s_h) \)
11. \( \) status=true
12. \( \) else
13. \( \) status=false
14. \( \) end
15. \( \) end
16. \( \) if \( (status==true) \rightarrow \) Join \( M_j \)
17. \( \) end
18. \( \) \( j=j+1 \)
19. \( \) \( S=\text{setdiff} \ (S, MI_j) \)
20. \( \) end
21. \( \) for \( k = 1 : |MI| \)
22. \( \) for \( q = 1 : |MI_k| \)
23. \( \) \( s_q = MI_k (q) \)
24. \( \) \( w(s_q) = \alpha.R_E + \beta.(\frac{1}{r_D}) \)
25. \( \) end
26. \( \) elect node \( MI_k \) having highest weight as CH
27. \( \) end
28. \( \) BS sends \textit{CH\_ELECTION} to all CH nodes
29. \( \) BS sends \textit{CM\_MEMBERSHIP} to all member nodes

---

Figure 8.4: EECDC algorithm
8.4 Simulation Setup and Results:

Since EECDC is also a centralized clustering algorithm, its performance is compared with other centralized algorithms such as Low Energy Adaptive Clustering Hierarchy – Centralized (LEACH – C), Adaptive Decentralized Re-clustering Protocol (ADRP) and Energy Efficient Clustering Algorithm Based on Neighbors (EECABN). The MATLAB environment is used for simulations. In total, 30 simulations are conducted for all the three scenarios mentioned in Chapter 3. For each scenario, 10 simulations are conducted by varying network deployment to ensure the reliability of the results. The network size of 100 m x 100 m square area is taken for these simulations. Like in Chapter 6 and 7, the BS is located at (50, 50) in scenario 1, (100, 100) in scenario 2 and (50, 150) in scenario 3. Area coverage model is used in these simulations (i.e.) a node will cover all the points in all the direction within its sensing radius $R_s$. In these simulations, equal importance has been given for residual energy and the total distance for electing the CH of a cluster set. The co-efficient $\alpha, \beta$ is set at 0.5 and $s$ at 0.8 in all the simulations. The aggregation ratio $R_{agg}$ is only applied for EECDC clusters and the value is set at 0.5 and $R_s=15m$ for all the simulations. There is no need of aggregation ratio for LEACH-C, ADRP and EECABN since the data generated within a cluster will be more redundant in nature.
Figure 8.5 depicts the total energy consumption per communication round in all the simulated algorithms. In the case of EECDC only one cluster set is activated during a communication round. But in the case of LEACH-C, ADRP and EECABN, all the clusters are activated during a communication round, where lots of redundant data will be there. Figure 8.6, 8.7 and 8.8 depict the network lifetime of LEACH-C, ADRP, EECABN and EECDC till last node death in the network. In all the scenarios EECDC shows improved performance by avoiding redundant data generation and processing it in a cluster. Comparing with other algorithms, EECDC achieves 2000, 1700 and 1500 more communication rounds than others in scenario 1, 2 and 3 respectively.
Figure 8.6: Network lifetime in scenario 1

Figure 8.7: Network lifetime in scenario 2
Figure 8.9 shows the First Node Die (FND) condition. It is observed from Figure 8.9 that EECDC increases the network lifetime by 150% than LEACH-C, 121% than ADRP and 108% than EECABN in scenario 1. In scenario 2, EECDC is better by 130% than LEACH-C, 107% than ADRP and 101% than EECABN. Considering scenario 3, EECDC shows betterment by 130% than LEACH-C, 114% than ADRP and 112% than EECABN.

Figure 8.10 depicts the Half Node Die (HND) of all the algorithms. EECDC shows network lifetime improvement by 198% than LEACH-C, 170% than ADRP and 167% than EECABN in scenario 1. On considering scenario 2, EECDC is better than 150% than LEACH-C, 119% than ADRP and 116% than EECABN. In scenario 3, EECDC is better by 120% than LEACH-C, 108% than ADRP and 103% than EECABN.
Figure 8.9: First Node Die in various scenarios

Figure 8.10: Half Node Die in various scenarios

Figure 8.11 depicts Most Node Die (MND) conditions for all the compared algorithms. But MND condition is not considered for analysis because after HND, the cluster set loses its reliability in EECDC. This is because of an increase in the percentage of uncovered points in ROI.
8.5 Conclusion:

In this chapter, a new clustering algorithm EECDC based on the MIS concept is proposed. EECDC is designed to reduce the energy wastage due to redundant data processing. EECDC forms logic clusters whose member node coverage area is non-overlapping. Also EECDC uses a token for cluster set activation. The cluster set which is having the token will alone be in the active state to monitor the environment and all other cluster set will be in the sleep state. The uncovered points in the ROI during a round can be covered during subsequent cluster set activations. The simulation results also indicated the energy conservation achieved using EECDC.

Figure 8.11: Most Node Die in various scenarios