International Journal of Electronics Communication and Computer Engineering
Volume 4, Issue 3, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

Node Scheduling In Wireless Sensor Network to Remove Data Conflict

Rahul Karmakar
Assistant Professor, Deptt. of CSE
IMPS College of Engg.& Technology
Email: rahulkarmakar6@gmail.com

Debasis Mandal
Assistant Professor, Deptt. of CSE
IMPS College of Engg.& Technology
Email: debimps07@gmail.com

Akhil Kumar Das
Assistant Professor, Deptt. of CSE
IMPS College of Engg.& Technology
Email: dasakhi@gmail.com

Abstract **- Recently, the idea of wireless sensor networks has attracted a great deal of research attention due to wide-ranged potential applications that will be enabled by wireless sensor networks, such as battlefield surveillance, machine failure diagnosis, biological detection, home security, smart spaces, inventory tracking, etc. A wireless sensor network consists of tiny sensing devices, deployed in a region of interest. Each device has processing and wireless communication capabilities, which enable it to gather information from the environment and to generate and deliver report messages to the remote base station (remote user)[6]. The base station aggregates and analyzes the report messages received and decides whether there is an unusual or concerned event occurrence in the deployed area. Considering the limited capabilities and vulnerable nature of an individual sensor, a wireless sensor network has a large number of sensors deployed in high density (high up to 20nodes/m3 and thus redundancy can be exploited to increase data accuracy and system reliability.**

Keywords **- Sensor Network, Base Station.**

I. INTRODUCTION

The objectives of this paper is to remove the data conflict that means more than one sensor will never send data to another sensor or Base Station (BS) at same time. Because if occurs so then the data will be lost or corrupted. Sensor collects data from other sensor to send it to Base Station or other sensor towards the Base Stations. Base Stations collects the data from other sensor to calculate some value and/or take a decision and monitor it. So if data confliction occurs then the monitoring device can't work accurately

II. ALGORITHM

1. Input the no of nodes of the Spanning Tree , sensing range of the nodes and area region of the network *(x* x *y*)[7].
2. Randomly generate the locations of the nodes (in x and y co-ordinate value) and calculate the distance between each pair of the nodes. Those, which will be inside the range of a node, will be considered as neighbor of this node.
3. Find the entire minimum component-spanning tree from the network using Prim's algorithm[2].
4. Select the one which has maximum no of nodes.
5. Represent the spanning tree as an adjacency matrix[2].
6. Find the leaf nodes the spanning tree and store it in an array. Perform level wise traversal (breadth first) to all the nodes. If a node has no child then includes it in the list of leaf node.
7. Create the communication graph using these leaf nodes. If two leaf nodes are adjacent of a same node then there will be an edge between these two nodes. If a leaf nodes is adjacent of a node n and another is neighbor of n or vice versa then there will be an edge between these two leaf nodes[1][5].
8. Find the maximal independent set from the communication graph[4]. Select all the leaf nodes which have the highest level in the spanning tree. If more than one node is selected then check which one has the minimum degree (degree in communication graph). If the level and the degree both are same of two nodes then select any one. Enlist this node in the independent set. Delete this node (say v_i) and all the other nodes (v_j) if there is an edge (v_i, v_j) in the communication graph and also delete the edges (v_i, v_j) and (v_j, v_k) from the communication graph. [v_i, v_j, v_k are the vertices of the communication graph]. repeat step 1 to 5 until the communication graph is empty[1].
9. Delete the members of the independent set from the spanning tree.
10. Repeat step 4 to step 7 until all the nodes of spanning tree is deleted (except the root).

III. DESCRIPTION WITH AN EXAMPLE

First the no of nodes (say 14) is given then the sensing range (say 30) and the region of the network is given (say 100 x 100)[7]. According to the distance between two nodes it is decided whether they are neighbor of each other or not.

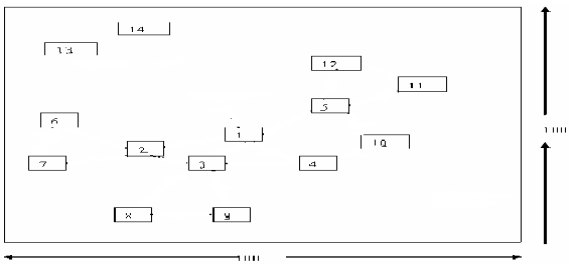

Fig.1

In the figure below those nodes are connected to each other (by dotted line or flat line) are neighbor of each other. Next a spanning tree is created from this wireless sensor network. Suppose the following is a wireless sensor network where all the nodes are sensor and node 1 (fulfilling the criteria of having at least one neighbor) is

International Journal of Electronics Communication and Computer Engineering
Volume 4, Issue 3, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

considered as the root of the spanning and we get a component-spanning tree which contains the nodes 1, 2, 3, upto 12. But still two node 13 and 14 are remaining so next 13 is selected as root to find another component spanning tree. Then from this network we get only one component-spanning tree and it is given as input to this scheduling algorithm. So the target of all the sensor nodes is to send the information to the root. Since it is a multi hop sensor network so all the nodes can't send data directly to the root, they will send to their parent. The fat line is indicating the spanning tree and dotted line indicating the neighbors. Here two nodes are not included in the spanning tree because they don't have any neighbor so they can't send data to any node.

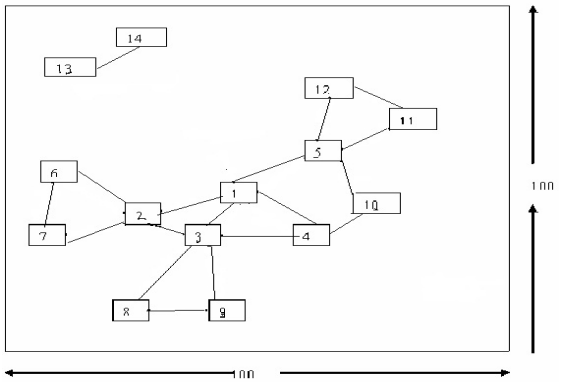

Fig.2

We will take the maximum component graph if there are no. of graphs. Here we have two graphs one is formed by node 1 to 12 and another is Formed by node 13 and 14.So according to our algorithm we take the First graph ie node 1 to 12.

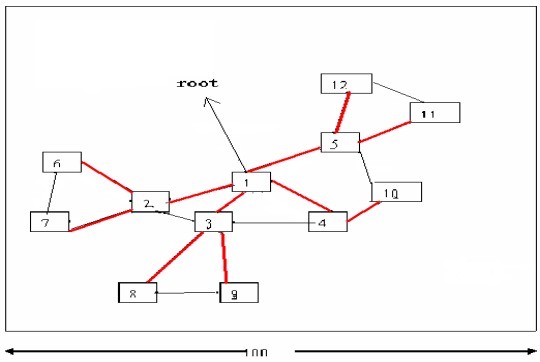

Fig.3

Round 1:

The leaf nodes and their corresponding level are :

Table:1

Node	Level
6	2
7	2
8	2
9	2
10	2
11	2
12	2

Communication Graph:

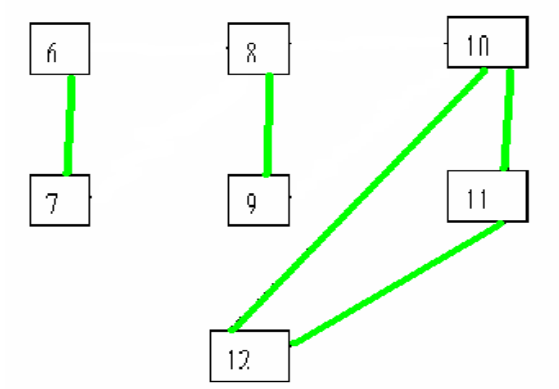

Fig.4

Communication Graph in Round 1:

Maximal independent set:- First 12 is taken as an element of the independent set because all the leaf nodes of round 1 is of same level (level= 2) and the degree of these nodes in communication graph are also same. So according to the algorithm first 12 is taken. Then 12 is deleted from the communication graph and since there is edge between 10-12 and 11-12 this is why 10 and 11 are also deleted from communication graph. Then among the remaining one node is enlisted into the independent set in the same way. So next 9 has been selected and then 7. After that there are no node remaining in the communication graph so 12 9 and 7 are the elements of the independent set in round 1.

Indpnt [ ] = { 12 9 7 }

So 12, 9, 7 will send data at the same time in round 1. After that these node will be deleted from the spanning tree.

After deleting these nodes the spanning tree is as follows:

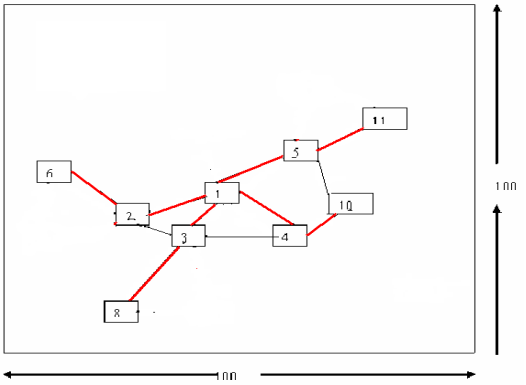

Fig.5

Round 2:

The leaf nodes and their corresponding level are:

Table: 2

Node	Level
6	2
8	2
10	2
11	2

International Journal of Electronics Communication and Computer Engineering
Volume 4, Issue 3, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

Communication Graph:

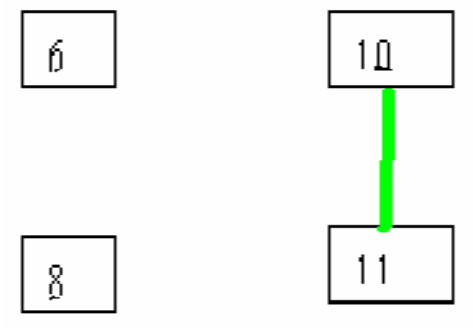

Fig. 6

Communication Graph in Round 2:

Maximal independent set:- First 11 is taken as an element of the independent set because all the leaf nodes of round 1 is of same level (level= 2) and the degree of these nodes in communication graph are also same. So according to the algorithm first 11 is taken. Then 11 is deleted from the communication graph and since there is edge between 10-11 this is why 10 is also deleted from communication graph. Then among the remaining one node is enlisted into the independent set in the same way. So next 8 has been selected and then 6. After that there are no node remaining in the communication graph so 11 8 and 6 are the elements of the independent set in round 2.

Indpnt [ ] = { 11 8 6 }

So 11, 8, 6 will send data at the same time in round 2. After that these node will be deleted from the spanning tree.

After deleting these nodes the spanning tree is as follows:

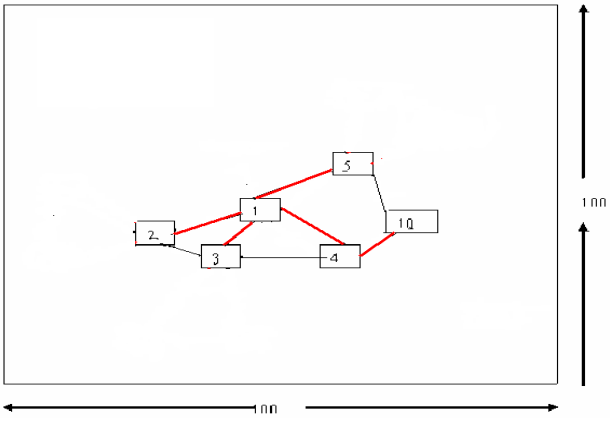

Fig. 7

Round 3:

The leaf nodes and their corresponding level are:

Table: 3

Node	Level
2	1
3	1
5	1
10	2

Communication Graph:

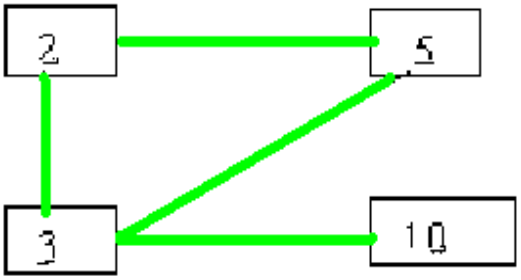

Fig.8

Communication Graph in Round 3:

*Maximal independent set:-*First 10 is taken as an element of the independent set because 10 is of highest level (level= 2) among the leaf nodes of round 3. So according to the algorithm first 10 is taken. Then 10 is deleted from the communication graph and since there is edge between 10 - 3 this is why 3 is also deleted from communication graph. Then among the remaining, one node is enlisted into the independent set in the same way. So next 5 has been selected and then 5 is deleted from the communication graph and since there is edge between 5 - 2 this is why 2 is also deleted from communication graph.. After that there are no node remaining in the communication graph so 10 and 5 are the elements of the independent set in round 3[4].

Indpnt [ ] = {10 5}

So 10 and 5 will send data at the same time in round 3. After that these node will be deleted from the spanning tree.

After deleting these nodes the spanning tree is as follows:

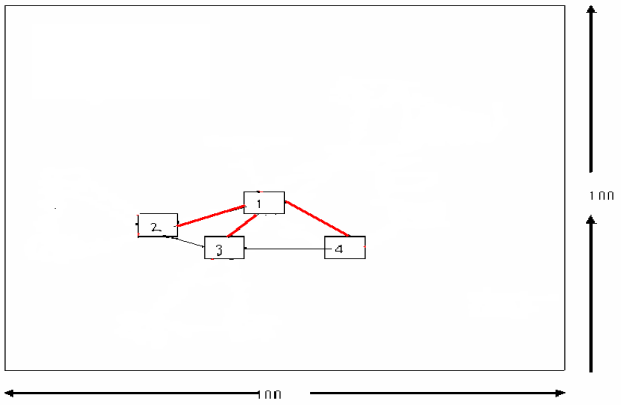

Fig.9

Round 4:

The leaf nodes and their corresponding level are:

Table: 4

Node	Level
2	1
3	1
4	1

International Journal of Electronics Communication and Computer Engineering
Volume 4, Issue 3, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

Communication Graph:

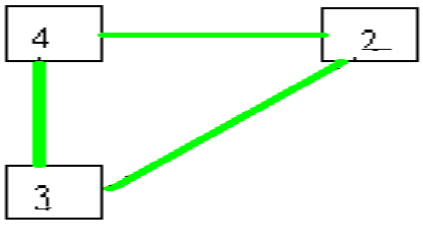

Fig. 10

Communication Graph in Round 4:

Maximal independent set:-

Indpnt [ ] = { 4 }

So only 4 will send data in Round 4. After that 4 will be deleted from the spanning tree . After deleting this node the spanning tree is as follows:

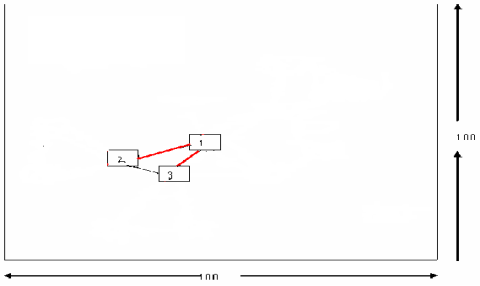

Fig. 11

Round 5 :

The leaf nodes and their corresponding level are:

Table: 5

Node	Level
2	1
3	1

Communication Graph:

Fig.12

Communication Graph in round 5:

Maximal independent set:-

Indpnt [ ] = { 3 }

So node 3 will send data at the same time in round 5. After that the node will be deleted from the spanning tree.

After deleting this node the spanning tree is as follows:

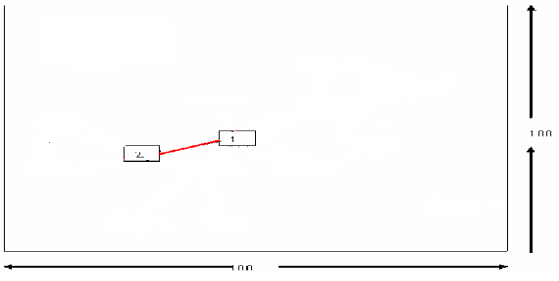

Fig.13

Round 6:

The leaf nodes and their corresponding level are :

Table: 6

Node	Level
2	1

Since there is only one leaf node in this round so no need to create communication graph, it will be included into the independent set.

Maximal independent set:

Indpnt [ ] = { 2 }

So node 2 will send data in round 6. After that this node will be deleted from the spanning tree.

After deleting this node the spanning tree is as follows :

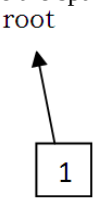

Fig.14

Since only the root is remaining, the algorithm will stop here.

The Scheduling Results is: In round 1 node (12,9,7) in round 2 node (11,8,6) in round 3 (10,5) in round 4 node (4) in round 5 node (3) and in round 6 node (2) will send data.

IV. Result of Simulation

Table wise data of number of rounds for particular no of sensor nodes with specific sensing range.

Table: 7

S. No.	No. of Nodes	Sensing Range	No. of Rounds	Average
1	40	50	18	
2	40	50	20	**19**
3	40	50	19	
1	80	30	17	
2	80	30	16	**16**
3	80	30	15	
1	50	30	19	
2	50	30	21	**20.66**
3	50	30	22	
1	100	20	37	
2	100	20	27	**33**
3	100	20	35	
1	120	18	34	
2	120	18	39	**35.33**
3	120	18	33	

V. CONCLUSION

In this paper, we proposed a node-scheduling scheme, which can remove data conflict, therefore reducing the data loss or erroneous result, by making a scheduling which sensor will send data when. We presented a basic scheduling algorithm which guaranties that data confliction will not occur but it may increase the delay to gather data and to send it to the Base Station (BS).

There may be more than one scheduling algorithm but we have tried here to give the optimal solution. It is not possible to give 100% because it is a NP hard problem but have to find out the algorithm, which will give the maximum possible. Although our algorithm achieves the goals of guaranteeing fully removal of data confliction, it still has improvement space. Furthermore, our simulation is based on the static networks. Node mobility may cause the inaccurateness of position information previously obtained in neighbor information obtaining step, thus affect the correctness of scheduling algorithm. The effect caused by node mobility needs further modification.

REFERENCES

[1] Graph Theory By Narsingh Deo
[2] Data Structure By Seymour Lipschutz.
[3] Wikipedia.com & Google.com
[4] The Independent Set Algorith Ashay Dharwadker, H-501 Palam Vihar, District Gurgaon ,Hariyana
[5] Wireless Network Coding and Node-Based Scheduling, M. Bama, Andrew Thangaraj, Srikrishna Bhashyam, Department of Electrical Engineering, Indian Institute of Technology, Madras, Chennai, India 600036
[6] Algorithms For Wireless Sensor Networks, Sartaj Sahni and Xiaochun Xu Department of Computer and Information Science and Engineering,University of Florida, Gainesville, FL 32611
[7] Distributed Algorithms For Improving Wireless Sensor Network Lifetime With Adjustable Sensing Range By Aung Aung.

AUTHOR'S PROFILE

Mr. Rahul Karmakar
Completed M.Tech. Degree in Computer Science & Engineering in 2009 from University of Calcutta, Currently working as Assistant Professor in the CSE Department at IMPS College of Engineering & Technology, Malda (West Bengal).

Mr. Debasis Mandal
Completed M.Tech. Degree in Multimedia & Software System in 2010 from National Institute of Technical Teachers' Training & Research, Kolkata. Currently working as Assistant Professor in the CSE Department at IMPS College of Engineering & Technology, Malda (West Bengal).

Mr. Akhil Kumar Das
Completed M.Tech. Degree in Multimedia & Software Systems in 2007 from National Institute of Technical Teachers' Training & Research, Kolkata. Currently working as Assistant Professor in the CSE Department at IMPS College of Engineering & Technology, Malda (West Bengal).