

TLEACH in NetSim

Software: NetSim Standard v13.2, Visual Studio 2022

Project Download Link:

https://github.com/NetSim-TETCOS/TLEACH_in_WSN_v13.2/archive/refs/heads/main.zip

Follow the instructions specified in the following link to download and setup the Project in NetSim:

<https://support.tetcos.com/en/support/solutions/articles/14000128666-downloading-and-setting-up-netsim-file-exchange-projects>

Low-energy adaptive clustering hierarchy ("LEACH") is a MAC protocol which is integrated with clustering and a simple routing protocol in wireless sensor networks (WSNs). The goal of LEACH is to lower the energy consumption required to create and maintain clusters in order to improve the lifetime of a wireless sensor network.

This Cross Layer Protocol is implemented in NetSim in MAC layer which involves ZigBee Protocol and Network layer which involves DSR protocol. The clustering of sensors happens in the Network layer and the Cluster head election involves interacting with the MAC layer to obtain the remaining power of the sensors.

TLEACH is Threshold-based LEACH in which nodes with remaining energy greater than a threshold value are considered to elect the cluster head. The sensor that is closer to the sinknode is elected as the cluster head. Whenever all sensors in a cluster reach energy levels lower than the threshold, the threshold is reduced.

A **TLEACH.c** file is added to the DSR project.

1. For this implementation of TLEACH, the number of Clusters is fixed as 4 and all the 4 clusters are equal. If the user wants to change it, then he/she must also change the static routing for the Cluster Heads and the ClusterElement array accordingly.

```
28
29 #include "main.h"
30 #include "DSR.h"
31 #include "List.h"
32 #include "../BatteryModel/BatteryModel.h"
33 #include "../ZigBee/802.15.4.h"
34 #define NUMBEROFCLUSTERS 4
35 #define SIZEOFCLUSTERS 16 //SIZEOFCLUSTERS can be 1,4,9,16,25
36 #define THRESHOLD_PROPORTION 0.7 //Threshold is set to 70% of the maximum battery level in the cluster
37
38 static int CHcount[NUMBEROFCLUSTERS];
39 static int prevCH[NUMBEROFCLUSTERS];
40 static int CHthreshold[NUMBEROFCLUSTERS]; //Contains the energy threshold value for each cluster
41
42 int sinknodeID;
43 #define FTI* fp:
44 //For 100 sensors and SIZEOFCLUSTERS = 25, uncomment this
45 //int ClusterElements[NUMBEROFCLUSTERS][SIZEOFCLUSTERS] = {{1,2,3,4,5,11,12,13,14,15,21,22,23,24,25,31,32,33,34,35,41,42,43,44,45}, \
46 {6,7,8,9,10,16,17,18,19,20,26,27,28,29,30,36,37,38,39,40,46,47,48,49,50}, \
47 {51,52,53,54,55,61,62,63,64,65,71,72,73,74,75,81,82,83,84,85,91,92,93,94,95}, \
48 {56,57,58,59,60,66,67,68,69,70,76,77,78,79,80,86,87,88,89,90,96,97,98,99,100}};
49
50 //For 64 sensors and SIZEOFCLUSTERS = 16, uncomment this
51 int ClusterElements[NUMBEROFCLUSTERS][SIZEOFCLUSTERS] = { {1,2,3,4,9,10,11,12,17,18,19,20,25,26,27,28}, \
52 {5,6,7,8,13,14,15,16,21,22,23,24,29,30,31,32}, \
53 {33,34,35,36,41,42,43,44,49,50,51,52,57,58,59,60}, \
54 {37,38,39,40,45,46,47,48,53,54,55,56,61,62,63,64} };
55
56 //For 36 sensors and SIZEOFCLUSTERS = 9, uncomment this
57 //int ClusterElements[NUMBEROFCLUSTERS][SIZEOFCLUSTERS] = {{1,2,3,7,8,9,13,14,15}, {4,5,6,10,11,12,16,17,18}, {19,20,21,25,26,27,31,32,33},
58
59 //For 16 sensors and SIZEOFCLUSTERS = 4, uncomment this
60 //int ClusterElements[NUMBEROFCLUSTERS][SIZEOFCLUSTERS] = {{1,2,5,6}, {3,4,7,8}, {9,10,13,14}, {11,12,15,16}};
61
62 //For 4 sensors and SIZEOFCLUSTERS = 1, uncomment this
63 //int ClusterElements[NUMBEROFCLUSTERS][SIZEOFCLUSTERS] = {{1}, {2}, {3}, {4}};
64
```

Figure 1: TLEACH.c file

- To make 4 equal clusters the number of sensors must be 4,16,36,64,100. Depending on the number of sensors, the ClusterElements array must be defined. Here, it has been defined and commented for 4,16,36,64,100 sensors. Uncomment the one you want to use.

The file contains the following functions.

- **fn_NetSim_TLEACH_CheckDestination();** // This function is used to check whether the current device is the destination (i.e) the sinknode or not. Else the packet will be forwarded to the next hop.
- **fn_NetSim_TLEACH_GetNextHop();** // This function is used to identify the next hop in cases where the current device is either a sensor within the cluster or the cluster head. Static routes are defined in this function. It returns the Device id of the next hop.
- **fn_NetSim_TLEACH_AssignClusterHead();** // This function is used to dynamically assign cluster heads within a cluster based on the residual energy. The sensor with higher remaining power in comparison to other sensors within the same cluster will be elected as the cluster head.
- **fn_NetSim_TLEACH_IdentifyCluster();** // This function is used to determine the cluster to which a sensor belongs. It returns the cluster id of the cluster.
- **fn_NetSim_TLEACH_init();** // It is used to initialize TLEACH parameters such as the sinknode ID and the initial Threshold value.
- **fn_NetSim_TLEACH_set_threshold();** // It is used to reduce the threshold value for clusters whenever all sensor in a cluster reach energy levels less than the threshold.

Example

- The **Workspace_TLEACH_in_WSN** comes with a sample network configuration that are already saved. To open this example, go to Your work in the Home screen of NetSim and click on the **TLEACH_in_WSN_Example** from the list of experiments.
- The example consists of a WSN network with 64 sensors placed uniformly along with a sinknode as shown below:

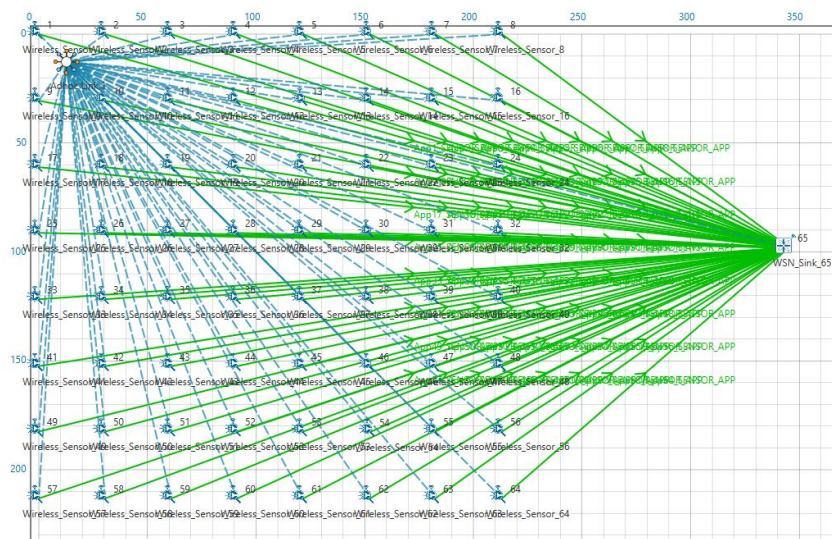


Figure 2: Network Topology

- Run the simulation.

Results and discussion: View the packet animation. You will note that the sensors directly start transmitting packets without route establishment since the routes are statically defined in TLEACH. You will also note that the cluster heads keep changing dynamically.