Load-Balancing Through Enhanced Clustering Technique in Wireless Sensor Network

Kamla Adhikari
Department of Software Engineering
NIET, Gr.Noida, Uttar Pradesh, India

Dr.Rajdev Tiwari
Department of Computer Science Engineering
NIET, Gr.Noida, Uttar Pradesh, India

Abstract- A wireless sensor network(WSN) consists of spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure, etc and to cooperatively pass their data through the network to the main location. Clustering of the sensor nodes are very important in order to achieve scalability, to prolong lifetime of sensor nodes etc. There are many algorithms that have been proposed earlier for clustering in WSN. In this paper an algorithm is described different from earlier algorithms which will also consider the cluster radius as well as residual energy of the sensor nodes in order to form clusters. .

Keywords – ***cluster, connectivity density, cluster radius, wireless sensor networks***

I. INTRODUCTION

In most wireless sensor network (WSN) applications now a days the entire network must have the ability to operate unattended in harsh environments in which pure human access and monitoring cannot be easily scheduled or efficiently managed or its even not feasible at all. Based on this critical expectation, in many significant WSN applications the sensor nodes are often deployed randomly in the area of interest by relatively uncontrolled means (i.e., dropped by a helicopter) and they form a network in an ad hoc manner. Moreover, considering the entire area that has to be covered, the short duration of the battery energy of the sensors and the possibility of having damaged nodes during deployment, large populations of sensors are expected; it's a natural possibility that hundreds or even thousands of sensor nodes will be involved. In addition, sensors in such environments are energy constrained and their batteries usually cannot be recharged. Therefore, its obvious that specialized energy-aware routing and data gathering protocols offering high scalability should be applied in order that network lifetime is preserved acceptably high in such environments. Naturally, grouping sensor nodes into clusters has been widely adopted by the research community to satisfy the above scalability objective and generally achieve high energy efficiency and prolong network lifetime in large-scale WSN environments. The corresponding hierarchical routing and data gathering protocols imply cluster-based organization of the sensor nodes in order that data fusion and aggregation are possible, thus leading to significant energy savings. In the hierarchical network structure each cluster has a leader, which is also called the cluster head (ch) and usually performs the special tasks referred above (fusion and aggregation), and several common sensor nodes (sn) as members. The cluster formation process eventually leads to a two-level hierarchy where the ch nodes form the higher level and the cluster-member nodes form the lower level. the sensor nodes periodically transmit their data to the corresponding ch nodes. The ch node aggregates the data (thus decreasing the total number of relayed packets) and transmits them to the base station (bs) either directly or through the intermediate communication with other ch nodes. However, because the ch nodes send all the time data to higher distances than the common (member) nodes, they naturally spend energy at higher rates.

A common solution in order to balance the energy consumption among all the network nodes is to periodically re-elect new chs (thus rotating the ch role among all the nodes over time) in each cluster. A typical example of the implied hierarchical data communication within a clustered network (assuming single- hop intra-cluster communication and multi-hop inter-cluster communication) is further illustrated in figure 1. The bs is the data processing point for the data received from the sensor nodes, and where the data is accessed by the end user. It is generally considered fixed and at a far distance from the sensor nodes. the ch nodes actually act as gateways between the sensor nodes and the bs.The function of each ch, as already mentioned, is to perform common functions for all the nodes in the cluster, like aggregating the data before sending it to the bs. In some way, the ch is the sink for the cluster nodes, and the bs is the sink for the chs.

A. Objectives and design challenges of clustering algorithm in WSN

As was mentioned at the beginning, hierarchical clustering in WSNs can greatly contribute to overall system scalability, lifetime, and energy efficiency. Hierarchical routing is an efficient way to lower energy consumption within a cluster, performing data aggregation and fusion in order to decrease the number of transmitted messages to the BS. On the contrary, a single-tier network can cause the gateway to overload with the increase in sensors density. Such overload might cause latency in communication and inadequate tracking of events. In addition, the single-tier architecture is not scalable for a larger set of sensors covering a wider area of interest because the sensors are typically not capable of long-haul communication. Hierarchical clustering is particularly useful for applications that require scalability to hundreds or thousands of nodes. Scalability in this context implies the need for load balancing and efficient resource utilization. Applications requiring efficient data aggregation (e.g., computing the maximum detected radiation around a large area) are also natural candidates for clustering. Routing protocols can also employ clustering. In addition to supporting network scalability and decreasing energy consumption through data aggregation, clustering has numerous other secondary advantages and corresponding objectives. It can localize the route setup within the cluster and thus reduce the size of the routing table stored at the individual node.

Beyond the typical challenges mentioned above (limited energy, limited capabilities, network life-time) some additional important considerations in the design process of clustering algorithms for WSNs should be the following: Cluster formation: The CH selection and cluster formation procedures should generate the best possible clusters (well balanced, etc.). However they should also preserve the number of exchanged messages low and the total time complexity should (if possible) remain constant and independent to the growth of the network. This yields a very challenging trade-off. Application Dependency: When designing clustering and routing protocols for WSNs, application robustness must be of high priority and the designed protocols should be able to adapt to a variety of application requirements. Secure communication: As in traditional networks, the security of data is naturally of equal importance in WSNs too. The ability of a WSN clustering scheme to preserve secure communication is ever more important when considering these networks for military applications.

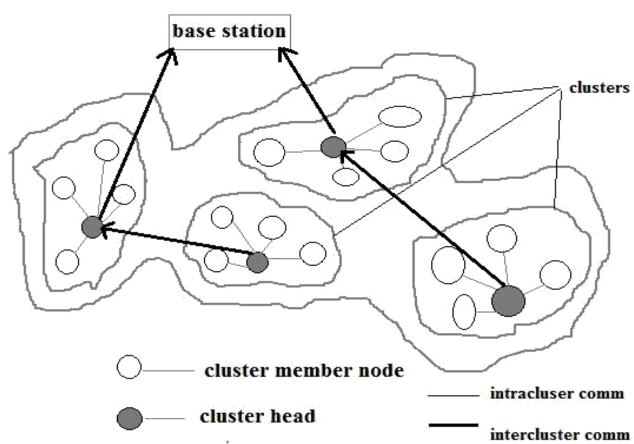

Fig.1. Data communication in a clustered network

II. LITERATURE SURVEY

1) Low Energy Adaptive Clustering Hierarchy:

One of the first and most popular clustering protocols proposed for WSNs was LEACH (Low Energy Adaptive Clustering Hierarchy). It is probably the first dynamic clustering protocol which addressed specifically the WSNs needs, using homogeneous stationary sensor nodes randomly deployed, and it still serves as the basis for other improved clustering protocols for WSNs. Its an hierarchical, probabilistic, distributed, one-hop protocol, with main objectives (a) to improve the lifetime of WSNs by trying to evenly distribute the energy consumption among all the nodes of the network and (b) to reduce the energy consumption in the network nodes (by performing data aggregation and thus reducing the number of communication messages). It forms clusters based on the received signal strength and also uses the CH nodes as routers to the BS. All the data processing such as data fusion and aggregation are local to the cluster. LEACH forms clusters by using a distributed algorithm, where nodes make autonomous decisions without any centralized control. All nodes have a chance to become CHs to balance the energy spent per round by each sensor node. Initially a node decides to be a CH with a probability p and broadcasts its decision. Specifically, after its election, each CH broadcasts an advertisement message to the other nodes and each one of the other (non-CH) nodes determines a cluster to belong to, by choosing the CH that can be reached using the least communication energy (based on the signal strength of each CH message). In Figure 2 the cluster formation scheme is given in a more clear view.

Generally, LEACH can provide a quite uniform load distribution in one-hop sensor networks. Moreover, it provides a good balancing of energy consumption by random rotation of CHs. Furthermore, the localized coordination scheme used in LEACH provides better scalability for cluster formation, whereas the better load balancing enhances the network lifetime. However, despite the generally good performance, LEACH has also some clear drawbacks. Because the decision on CH election and rotation is probabilistic, there is still a good chance that a node with very low energy gets selected as a CH. Due to the same reason, it is possible that the elected CHs will be concentrated in one part of the network (good CHs distribution cannot be guaranteed) and some nodes will not have any CH in their range. Also, the CHs are assumed to have a long communication range so that the data can reach the BS directly.

This is not always a realistic assumption because the CHs are usually regular sensors and the BS is often not directly reachable to all nodes. Moreover, LEACH forms in general one-hop intracluster and intercluster topology where each node should transmit directly to the CHs and thereafter to the BS, thus normally it cannot be used effectively on networks deployed in large regions.

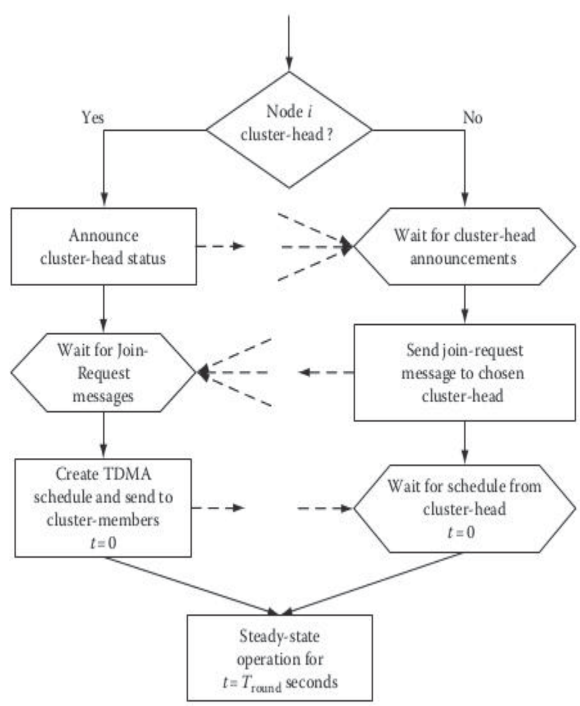

Fig.2. Flowchart of the cluster formation process of LEACH.

2) Hybrid Energy-Efficient Distributed Clustering:

Another improved and very popular energy-efficient protocol is HEED (Hybrid Energy-Efficient Distributed Clustering) . HEED is a hierarchical, distributed, clustering scheme in which a single-hop communication pattern is retained within each cluster, whereas multi-hop communication is allowed among CHs and the BS. The CH nodes are chosen based on two basic parameters, residual energy and intra-cluster communication cost. Residual energy of each node is used to probabilistically choose the initial set of CHs. On the other hand, intra-cluster communication cost reflects the node degree or nodes proximity to the neighbor and is used by the nodes in deciding to join a cluster or not. Thus, unlike LEACH, in HEED the CH nodes are not selected randomly. Only sensors that have a high residual energy are expected to become CH nodes. Also, the probability of two nodes within the transmission range of each other becoming CHs is small. Unlike LEACH, this means that CH nodes are well distributed in the network. Moreover, when choosing a cluster, a node will communicate with the CH that yields the lowest intra-cluster communication cost. In HEED, each node is mapped to exactly one cluster and can directly communicate with its CH. Also, energy consumption is not assumed to be uniform for all the nodes. The algorithm is divided into three stages. At the beginning, the algorithm sets an initial percentage of CHs among all sensors.

This percentage value, Cprob, is used to limit the initial CHs announcements to the other sensors. Each sensor sets its probability of becoming a CH, CHprob, as follows

CHprob = Cprob Eresidual / Emax

where Eresidual is the current energy in the sensor and Emax is the maximum energy, which corresponds to a fully charged battery. CHprob is not allowed to fall below a certain threshold pmin , which is selected to be inversely proportional to Emax. The main body of the algorithm consists of a (constant) number of iterations. Every sensor goes through these iterations until it finds the CH that it can transmit to with the least transmission power (cost). If it hears from no CH, the sensor elects itself to be a CH and then sends an announcement message to its neighbors informing them about the change of status. Finally, each sensor doubles its CHprob value and goes to the next iteration of this phase. It stops executing this phase when its CHprob reaches 1. Therefore, there are two types of CH status that a sensor could announce to its neighbors: (a) The sensor becomes a tentative CH if its CHprob is less than 1 (it can change its status to a regular node at a later iteration if it finds a lower cost CH). (b) The sensor permanently becomes a CH if its CHprob has reached 1. At the end, each sensor makes a final decision on its status. It either picks the least cost CH or announces itself as CH. Note also that for a given sensors transmission range, the probability of CH selection can be adjusted to ensure inter-CH connectivity. Generally, HEEDs mechanism to select the CHs and form the clusters produces a uniform distribution of cluster heads across the network through localized communications with little overhead. It also clearly outperforms LEACH with regard to the network lifetime and the desired distribution of energy consumption. However, synchronization is required and the energy consumed during data transmission for far away cluster heads is significant, especially in large-scale networks. Also, a knowledge of the entire network is normally needed to determine reliably the intra-cluster communication cost and configuration of those parameters might be difficult in practical world.

III. PROPOSED ALGORITHM

We know that the sensor nodes closer to the base station consume much more energy due to the relaying network traffic near the base station. Hence, the sensor nodes closer to the base station may quickly exhaust battery. Besides the residual energy, during the cluster head election we will also discuss the distance to the base station. Main idea is to decrease the size of the cluster closer to the base station.

A. System model

Energy required to transmit l-bit message over a distance d is,

ET (l,d) = ET elec(l) + ET amp(l,d)

where, ET elec(l) is electronics energy required to transmit l-bit message and ET amp(l,d) is amplifier energy required to amplify l-bit message over distanced.

B. DSBCA: Distributed self-organizing load balanced clustering algorithm

Earlier algorithm like LEACH, HEED assume uniform distribution of sensor nodes which is not the case in real life . Sensor nodes are normally distributed non-uniformly. If we do not take distance of nodes from base station in

consideration then it will cause unbalanced structure. This algorithm generates more balanced clusters than earlier algorithm. Cluster heads near the base station also forward data from farther clusters so they consume more energy than the cluster heads farther from the base station. And we also know that if a cluster contains too many nodes then there required too much energy due to communication overhead. Hence, based on the above concerns, DSBCA algorithm considers the connectivity density and the location of the node, trying to build a more balanced clustering structure.
In this algorithm we calculate the clustering radius by using connectivity density and distance from the base station. Fig. 3 shows DSBCA clustering in uniform distribution. DSBCA forms different clustering layers in which the radius of farther clustering layers are larger, and in the same layer the clustering radius is identical.

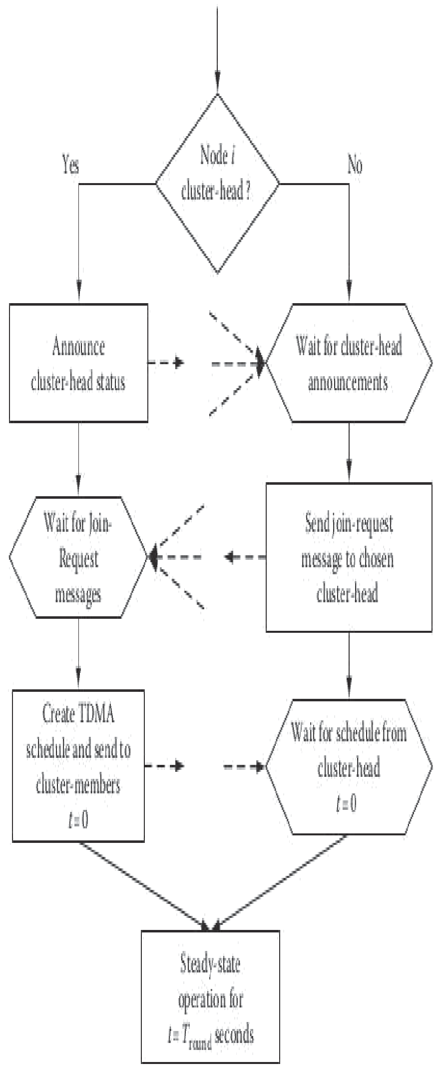

Fig. 3. DSBCA clustering in uniform distribution

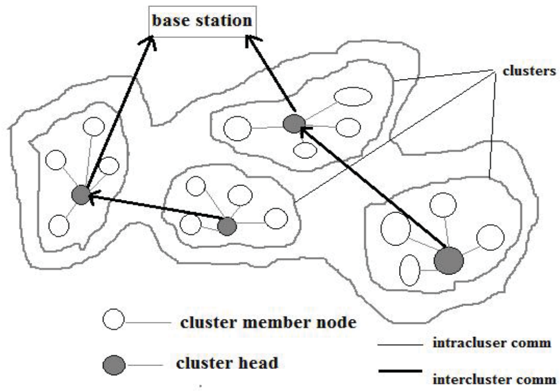

Fig. 4. DSBCA clustering in nonuniform distribution.

Fig. 4 shows DSBCA clustering in non-uniform distribution. In the case of non-uniform distribution, the cluster radius are determined by the distance from the base station and connectivity density of nodes. With farther distance from the base station and lower connectivity density, the cluster radius is larger; on the contrary, with closer distance from the base station and lower connectivity density, the cluster radius is smaller.

1) Cluster head selection phase: DSBCA selects the random nodes to trigger clustering process first. Then the triggered nodes calculate connectivity density and distance from the base station in order to calculate cluster radius k, and become the temporary cluster head.
k = floor(D(u) / Dk(u))

where D(u) is distance form base station and it is calculated by taking into consideration RSSI(received signal strength indicator) and Dk(u) is the connectivity density.

DSBCA follows a distributed approach to build hierarchical structure in self-organizing mode without central control. In this phase, the node with the highest weight in k-hop neighbors of Ut is elected as cluster head. The weight of the node is calculated by taking into consideration following parameter : residual energy, connection density, and times of being elected as cluster head. Thus, we can generate clusters more balanced in energy and position.

Initially selected temporary cluster heads sends hello message to all the k-hop neighbors. The k-hop neighbors calculate their respective weight and then the node with highest weight become the new cluster head. the newly elected cluster heads broadcast head message to their k-hop neighbors to declare itself as cluster head and asks them to join the cluster. Head message contains HID(cluster head id), SID(sender id), HD (number of hops from the cluster head). The algorithm discards head message if HD is greater than k. When a neighbor node receives Head message, even if it is already in a cluster, it sends Join message to the cluster head to request joining the new cluster as long as its weight is lower. It is possible that some nodes do not receive any head message. In this algorithm if a node does not receive any head message in T(w) T(r) time it declare it self as cluster head. Where T(w) is waiting time and T(r) is refreshing time.

2) Cluster building phase: If the number of nodes in a cluster is very large then it causes extra overhead as well as extra communication charge. So in this algorithm there is a upper limit for number of nodes in a cluster. If a cluster head receives join message form an ordinary node it will check the count of cluster node if it is less than the upper limit it will accept the message and increase the counter otherwise it will simply discard the message.

Every node of cluster maintains a information table, which contains the HID, HD, SID and other information. If a node receives transmitting packet, it will update its information table correspondingly. For example, the node checks HD in a newly received packet, if HD is smaller, then it updates the value of HD in table, with SID updated. That is to say, it has found a shorter path to cluster head and sets the new SID as its forwarding node. There is only a single HID entry in the ordinary node because it belongs to one cluster head.

3) Cycle phase: IN DSBCA cluster heads are not fixed because that will caused the early node death. So cluster heads are being altered periodically to balance the node energy consumption. Reelection of cluster head is triggered after every T(r) (refreshing time) time. The current cluster head collect the weight of every nodes in its cluster then it will check the node with highest weight and elect that node to new cluster head.

IV.CONCLUSION

In this paper, we have discussed a balanced clustering algorithm with distributed self-organization for WSNs of non-uniform distribution, taking into account optimal configuration of clusters. Compared with traditional clustering algorithms, the proposed algorithm can form more stable and reasonable cluster structure, and also improve the network life cycle significantly.

REFERENCES

[1] H. C. Lin and Y. H. Chu, A clustering technique for large multihop mobile wireless networks, in 51st Int. Conf. Veh. Technol., May 2000, pp. 15451549

[2] W. R. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, Energy- efficient communication protocol for wireless micro sensor networks, in Proc. 33rd Annu. Hawaii Int. Conf. Syst. Sci., Jan. 2000, pp. 110.

[3] Ying Liao, Huan Qi, and Weiqun Li, “Load-Balanced Clustering Algorithm With Distributed Self-Organization for Wireless Sensor Networks”,IEEE SENSORS JOURNAL, VOL. 13, NO. 5, MAY 2013

[4] O. Younis and S.Fahmy, “HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks, “IEEE Trans.Mobile Comput., vol. 3, no.4,pp.366- 379,Dec. 2004

[5] http://en.wikipedia.org/wiki/Low_Energy_Adaptive_Clustering_Hierarchy