Maximal Clique Based Clustering Scheme for Wireless Sensor Networks

Kamanashis Biswas, Vallipuram Muthukkumarasamy, Elankayer Sithirasanen

*School of Information and Communication Technology, Griffith University
Gold Coast, Australia*

kamanashis.biswas@griffithuni.edu.au

v.muthu@griffith.edu.au

e.sithirasanen@griffith.edu.au

Abstract—In this paper we present a self-organizing, singlehop clustering scheme, which is based on partitioning sensor networks into several disjoint cliques. Clustering sensor nodes into small groups is an effective method to achieve scalability, fault tolerance, load balancing, routing etc. Here, we develop and analyze maximal clique based cluster-first technique where each node obtains a list of its neighbours' connectivity as well as their degree of connection at first. Then, the node with highest degree of connection initiates clique formation process and makes the cluster. Among all the members of the cluster, the node with maximum energy is selected as cluster head (CH). The proposed technique has a number of advantages. For example, it requires only the knowledge of one-hop neighbours to form clusters. Furthermore, the clustering algorithm is robust for topological change caused by node failure, node mobility, CH change and even for node insertion or removal. Simulation results show that our proposed clustering scheme gives better performance in terms of cluster size, variance of cluster size, and number of single node clusters than the existing clustering algorithms such as Secure Distributed Clustering, LEACH and LCA.

I. INTRODUCTION

The rapid advances in low-cost hardware and low-power design have fostered the development of small-sized battery operated sensors that are capable of detecting and monitoring ambient conditions such as temperature, pressure, humidity, sound etc. Sensors are basically simple processing devices with limited computational capability, memory and transmission range. Wireless sensor networks usually contain thousands of such small, inexpensive sensors, which are randomly deployed in open, unprotected and harsh environment for long time to collect data. Given the vast area to be covered, the short lifespan of the battery-operated sensors and the possibility of having damaged nodes during deployment, sensors are expected to be densely distributed in most WSNs applications. Designing, operating and maintaining such large size network would require scalable architectural and management strategies [1]. Moreover, sensors are powered by batteries, which cannot normally be recharged. Hence, grouping sensor nodes into clusters and designing energy-efficient algorithms have become a significant factor in wireless sensor networks.

In this paper, we propose a maximal clique-based clustering (MCC) scheme that forms non-overlapping clusters of size n ;

where n is the maximum cluster size. By exchanging information of 1-hop neighbours, all sensor nodes in the network are grouped into a number of disjoint cliques, in which all the nodes can directly communicate with each other. Among all the nodes in a cluster, the node with maximum energy becomes cluster head. The key features of MCC are: self-configuration and localized coordination, maximum energy cluster head, periodical rotation of cluster head, fault tolerance and scalability. This clustering scheme can be used in data fusion, routing and implementing security services.

II. RELATED WORK

A randomly deployed wireless sensor network requires a cluster formation protocol to partition the network into clusters. Moreover, most of the WSN applications need data aggregation to reduce communication overhead. Data aggregation combines data from different sources by using functions such as *suppression*, *min*, *max*, *average*. In some network models, all aggregation functions are assigned to more powerful and specialized nodes known as cluster heads. Based on the order in which cluster formation and CH selection are performed, the clustering protocols can be divided into two categories: i) leader-first approach and ii) cluster-first approach [2]. In leader-first approaches (e.g. [3-9]), cluster heads are selected first based on certain metrics (e.g. best energy level, maximum connectivity, data attributes) and then they collaborate on how to assign other nodes to different clusters. On the other hand, in cluster-first approaches (e.g. [2, 10, 11]), all the nodes first form clusters and one of the nodes from each cluster is elected as cluster head based on some specific features (least distance, maximum power). In these clustering techniques, sensor nodes are almost always divided into a number of cliques in order to achieve direct communication with each other.

Linked Cluster Algorithm (LCA) is a leader-first based cluster formation protocol in which a node declares itself to be a CH if it has the lowest ID among the non-covered neighbour nodes [3-5]. A node is covered if it is in 1-hop neighbourhood of another node, which has declared itself to be a CH. Unlike LCA, Low Energy Adaptive Clustering Hierarchy (LEACH) forms clusters based on the received signal strength and uses the CH nodes as routers to communicate with base stations [8].

It uses a function to determine the probability p of a node that wants to be a cluster head (CH). A candidate node with probability p broadcasts its decision of candidature to other nodes. Each non-CH node finds out its cluster by selecting the CH that can be reached by using the lowest communication energy. LEACH also performs periodical rotation of cluster heads in order to balance the load. The main problem of both LCA and LEACH is that the selection of cluster head is arbitrary in LCA and probabilistic in case of LEACH. Hence, there is a good chance that a node with very low energy gets selected as a CH. When this node dies, the whole cluster becomes dysfunctional. In most of the energy-efficient clustering algorithms [7-9], CH selection process is probabilistic rather than deterministic.

Beside a large number of leader-first approaches, few cluster-first approaches are also proposed which mainly emphasize on routing and security issues. In [11], the authors present a routing protocol for ad-hoc networks that divides the network into a number of overlapping clusters. The basic goal of this protocol is to minimize communication overhead during topology updates. In [12], the authors describe a clique based distributed group formation algorithm for autonomous agent coalitions. In this protocol, each node computes its own clique of pre-defined size and then interchanges the information to form mutually strong inter-group communication connectivity. This cluster formation scheme is adopted in [2, 13].

The cluster first approach described in [2] is interesting one where the authors claim that most of the existing clustering protocols assume benign environments and cannot survive attacks from malicious nodes in hostile environment. Secure distributed clustering process ensures security mechanisms by dividing the network into multiple small groups and providing guarantee that all the normal nodes in each small group agree on the same group membership. But it faces a number of problems such as communication overhead, computation overhead as well as storage overhead.

In this paper, we present a distributed cluster formation protocol MCC, which is slightly different from the above protocols. In [2], each node computes its local maximum clique and then exchanges the information to form final clique which is time consuming. On the other hand, in MCC, only the nodes with highest degree of connection computes clique of n size. The nodes, which become members of a cluster never get the chance to initiate clique formation process.

III. MAXIMAL CLIQUE BASED CLUSTER FORMATION SCHEME

We first describe the network model and then our proposed clustering scheme.

A. Network Model

Consider a set of sensors dispersed in the field. We assume the following properties about the sensor network:

- Every node is assigned a unique ID.
- Links are bidirectional, i.e. two nodes u_1 and u_2 are connected if both of them can communicate to each other.

- A message sent by a node is received correctly within a finite time by all of its one-hop neighbours.
- Network topology does not change during cluster formation process.
- The network consists of multiple mobile/stationary nodes, which implies that energy consumption is not uniform for all nodes.

B. The MCC Protocol

In this section we describe the proposed MCC protocol. First, we present our protocol step by step and then illustrates with examples. The first five phases describe the cluster formation process while the rest two specifies CH rotation, addition and deletion of node respectively.

1) *Constructing Connectivity Matrix Phase*: Each node receives the connectivity list from its all one-hop neighbours and computes the degree of connectivity from that list. This can be provided by a physical layer for mutual location and identification of radio nodes.

2) *Cluster Formation Phase*: This phase is continued until all nodes join in a cluster. Each node checks its connectivity matrix and finds out whether it holds highest degree of connection or not. If it does not, then it waits until it is added to a cluster by neighbour nodes or it becomes the highest degree node after some iteration. Otherwise, it waits for random time t , and starts clique formation phase.

3) *Clique Formation Phase*: It is well known that finding out maximum clique in a random graph is an NP-complete problem [14]. To reduce the time complexity, we use greedy algorithm to compute maximal clique as shown in the following *algorithm 1* [15]. We assume that the algorithm is executed by node m .

Algorithm 1: Greedy Algorithm to find maximal clique

INPUT: $G_i = \{V_i, E_i\}$, $i \in V_i$ // Connectivity matrix of node m
OUTPUT: C_i // Cluster node list, initially holds m
STEPS:
 $CS = n$; // Set cluster size, it equals to node size by default
 $CN = m$; // Set current node to m
if G_i of node m is null,
 then return // indicates single node cluster
else
 while (size of $C_i \leq CS$ && exists nodes in common)
 Find k , having maximum common connection with CN
 Add k to C_i
 $CN = k$;
 end while
end if-else

4) *Cluster Head Selection Phase*: After finishing clique formation process, the node which initiated clustering process will select the cluster head. It will check the received signal strength of all other nodes in the cluster and choose the best node as CH [16]. In case of tie, the lowest node ID will be selected as CH.

5) *Cluster Announcement Phase*: The nodes which have joined in the cluster, will inform neighbours regarding its

membership. Hence, the non-clustered nodes will be able to update their connectivity matrix list by removing the clustered nodes.

6) *Cluster Head Rotation Phase*: The cluster heads of all clusters will periodically check the energy level of its neighbour nodes. If it finds a node with more energy level then the current CH will request the node to become CH and will convey the information to all other nodes in the cluster.

7) *Node Insertion and Deletion Phase*: Node insertion phase is executed by the cluster head if it receives an add request containing neighbour list from a new node. On receiving such request, the CH checks whether this node is connected to all other members in the clique or not. If the CH gets positive confirmation from all its neighbours then the new node is added to the cluster and every node would update their clique information immediately. A node will announce itself as CH if it fails to join in any cluster.

Every CH periodically checks connectivity links to find out the status of clique agreement. Inconsistency may occur if the sensor network contains mobile nodes or due to node failures. In such case, the inconsistent node will be removed from the cluster. The node which is removed from the cluster will initiate insertion phase to join in another cluster after discovering its neighbours in new location. If a cluster head becomes unavailable due to mobility or failure, the other nodes of the cluster select the new CH on the basis of energy level and remove the information of the previous CH from their neighbour lists.

In the following, we explain cluster formation process in details. To facilitate the discussion, we use simple graph as shown in Fig. 1. It shows a sensor network consisting of six sensor nodes and nine edges. A bidirectional edge from node i to node j indicates that both node are directly connected to each other.

Fig. 1: Sensor network with six nodes

In this network, when *node 1* receives the connectivity list $\{1, 2, 3\}$ from *node 5*, it ignores the link between *node 5* and *node 2* as *node 1* is not directly connected to *node 2*. But it does not decrease the degree of connectivity. Thus, each node forms its connectivity matrix using only common links.

From the connectivity matrix, each node knows whether it contains highest degree of connection or not. Any node containing equal or highest degree of connections can start cluster formation process after random time t to avoid collision. For example, in the above figure, both *node 1* and *node 3* have 4 connections. So, both of them are eligible for starting cluster formation process. But, one of them will be

selected depending on the random timer. Suppose, *node 1* gets the chance, then *node 1* will inform all its neighbours about its clustering process. The neighbour nodes will wait for confirmation message from *node 1* to know whether it is selected or not to join in the cluster formed by *node 1*.

TABLE I
CONNECTIVITY MATRIX OF NODE 1

CM of node 1	3	4	5	6	DoC
1	1	1	1	1	4
3	0	1	1	0	4
4	1	0	0	1	3
5	1	0	0	0	3
6	0	1	0	0	2

During this waiting period, if any other node attempts to initiate clustering process, the candidate node will receive a negative acknowledgement from the waiting node. As a result, the node trying to form new cluster will also go to wait state until it gets any feedback from that neighbour node. As soon as *node 1* has been formed its cluster, it will send a message ($C_{info}: \{1, 3, 4\}$) to its neighbours including the list of nodes who are nominated as members of that cluster. Each node selected to join in *node 1*'s cluster will inform all its neighbours so that they can remove the clustered nodes from their lists. It ensures that a node which is already joined in a cluster cannot be added to another cluster anymore. In this way, all the nodes will be member of a cluster. If a node does not have any node in its connectivity list, then the node itself form a cluster. For example, when *node 1* sends cluster information to *node 6*, *node 6* will discard *node 1* and *node 4* from its connectivity list. As a result, it will be a single node without any link. Hence, *node 6* will form a single node cluster itself. The following figure shows the final output of the cluster formation process. The dotted circle in Fig. 2 denotes cluster heads selected on the basis of energy level.

Fig. 2: Cluster formation

C. Effects of Defining Clique Size

Since cluster nodes are energy-constrained, the network's lifetime is a major concern, especially for applications of WSNs in harsh environments. Setting almost equal sized clusters becomes crucial for improving the network lifetime since it prevents the exhaustion of the energy of a subset of CHs at high rate. In our proposed scheme, maximum cluster size can easily be defined by setting limit on clique size. Thus it is possible to divide large networks into smaller clusters and

ensure that CHs are not overburden. The following set of clusters is obtained when we set maximum clique size '2'.

Fig. 3: Cluster with maximum clique size {2}

IV. EXPERIMENTS AND ANALYSIS

We simulated our proposed MCC algorithm described in Section III-B using MATLAB and compared the results with two leader-first protocols: LCA and LEACH as well as with one cluster-first protocol: Secure Distributed Clustering (SDC). LCA is one of the earliest clustering schemes while LEACH is the most popular and widely used one. On the other hand, SDC is a clique based approach like MCC. For our experiments, we randomly deploy 100, 200, 300, 400 and 500 sensor nodes in a 100×100 (m²) simulation area respectively. The transmission range of all the sensor nodes is fixed to 25 meters except LEACH. In LEACH, the energy level is set to 0.5 joules initially. The following graphs presenting simulation outcome is the average result of 1000 experiments. In addition to LEACH, LCA and SDC, we have also analysed the effect of defined clique size of our MCC algorithm in cluster formation process. Here, we limit the cluster size on the basis of the assumption that no cluster will contain more than 10% of total nodes in wireless sensor networks. This strategy protects the cluster heads from overburden and thus improves the network stability. The metrics that we have used to evaluate cluster characteristics are: average cluster size, coefficient variance of cluster size, number of single node clusters and maximum cluster size for MCC with and without defined cluster size.

Fig. 4(a) presents the average cluster size of the four protocols. In sensor networks, it is preferred that average

cluster size should not be too small due to ensure optimal clustering. Even, it is not desirable to include too many nodes in a cluster due to the increasing message collisions and transmission delay. The average cluster size of the four protocols increases with the node density of the network. The increment is almost linear and significant in both LEACH and LCA while it is comparatively smaller in MCC and SDC. The reason is that all nodes in a cluster in MCC protocol are located at one-hop distance but in LCA, the maximum distance between any two nodes in one cluster is two hops. As a result, MCC has a smaller average cluster size than the others. Like LCA, same trend is observed in LEACH. This is happened due to probabilistic cluster head selection procedure. If the selected cluster heads are located very near to each other, there will obviously some clusters with large number of nodes [16]. Fig. 4(b) presents the variance of cluster sizes. We measure Coefficient of Variance using the formula: $100 \times (\text{Standard deviation})/(\text{Mean value of set})$. The result shows that MCC has a smaller coefficient variance than the other three protocols which indicates our protocol creates more uniform clusters. Fig. 4(c) shows the single node clusters generated by the four protocols. It shows that MCC has fewer single node clusters than LEACH, LCA and SDC with the increment of node density in the network. This is because LCA attempts to generate the largest cluster first and thus leave some nodes into small clusters. Similarly, in LEACH, if boundary nodes or many nodes in the same region are selected as CHs then the number of single node clusters increases. In SDC, maximum cliques are obtained to form the clusters which lead a number of single nodes in the network. For densely located sensor nodes in a network, MCC provides minimum single node clusters according to the experiment.

Finally, we have examined the effect of undefined and defined clique size for our proposed protocol. The result indicates that a cluster may contain more than sixty nodes, if clique size is not defined. This will require the cluster head to perform huge in-network processing. This situation can easily be avoided by defining clique size as it is not possible to form a cluster greater than the pre-defined size.

Fig. 4(a): Average cluster size

Fig. 4(b): Coefficient of variance on cluster size

Fig. 4(c): Number of single node clusters

Fig. 4(d): Maximum cluster size

V. CONCLUSION

In this paper, we proposed a maximal clique based cluster formation protocol which discovers its connected neighbour sets and initiates clustering process on the basis of degree of connectivity. The proposed algorithm can be widely used in sensor network domain such as data fusion, routing, conformity checking etc. Currently, we implement the algorithm in uni-processor system. We believe that parallel clique computation by several independent nodes will make the clustering process faster. Our future plan is to enhance the protocol to provide efficient routing. We will also investigate the energy dissipation and network lifetime of MCC protocol to compare it with conventional clustering schemes. Moreover, we would like to implement security mechanisms to prevent malicious attacks that are very common to wireless sensor networks.

ACKNOWLEDGMENT

We wish to thank the anonymous reviewers for their valuable comments and suggestions.

REFERENCES

- [1] A. A. Abbasi and M. Younus, "A Survey on Clustering Algorithms for Wireless Sensor Networks," *Journal of Computer Communications*, Special Issue on Network Coverage and Routing Schemes for Wireless Sensor Networks, Vol. 30, pp. 2826 - 2841, 2007.
- [2] K. Sun, P. Peng and P. Ning, "Secure Distributed Cluster Formation in Wireless Sensor Networks," in *22nd Annual Computer Security Applications Conference*, ACSAC '06, Miami Beach, FL, USA, pp. 131 - 140, Dec. 2006,
- [3] D. J. Baker and A. Ephremides, "The Architectural Organization of a Mobile Radio Network via a Distributed Algorithm," *IEEE Transactions on Communications*, Vol. 29(11), pp. 1694 - 1701, 1981.
- [4] D. J. Baker, A. Ephremides and J. A. Flynn, "The design and simulation of a mobile radio network with distributed control," *IEEE Journal on Selected Areas in Communications*, pp. 226 - 237, 1984.
- [5] A. Ephremides, J. E. Wieselthier and D. J. Baker, "A design concept for reliable mobile radio networks with frequency hopping signaling," in *Proc. of IEEE*, Vol. 75(1), pp. 56 - 73, 1987.
- [6] K. Xu and M. Gerla, "A heterogeneous routing protocol based on a new stable clustering scheme," in *Proc. of IEEE Military Communications Conference*, Vol. 2, Anaheim, CA, pp. 838 - 843, 2002.
- [7] S. Bandyopadhyay, and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *Proc. of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, Vol.3, San Francisco, California, pp. 1713 - 1723, April 2003.
- [8] W. B. Heinzelman, A. P. Chandrakasan and H. Balakrishnan, "Application specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Networking*, Vol. 1(4), pp. 660 - 670, 2002.
- [9] O. Younis and S. Fahmy, "HEED: A Hybrid, Energy-Efficient, Distributed clustering approach for Ad Hoc sensor networks," *IEEE Transactions on Mobile Computing*, Vol. 3(4), pp. 366-379, 2004.
- [10] H. Ishii and H. Kakugawa, "A self-stabilizing algorithm for finding cliques in distributed systems," in *21st IEEE Symposium on Reliable Distributed Systems (SRDS'02)*, Oct. 2002.
- [11] P. Krishna, N. H. Vaidya, M. Chatterjee and D. K. Pradhan, "A Cluster-based Approach for Routing in Dynamic Networks," in *ACM SIGCOMM Computer Communication Review*, Vol. 27(2), pp. 49 - 64, Apr. 1997.
- [12] P. Tossic and G. Agha, "Maximal clique based distribution group formation for autonomous agent coalitions," in *3rd Int'l joint Conf. on Agents and Multi Agent Systems*, 2004.
- [13] A. B. Bomgni and J. F. Myoupo, "An Energy-Efficient Clique-Based Geocast Algorithm for Dense Sensor Networks," in *Journal of Communications and Network*, SciRes., pp. 125-133, 2010.
- [14] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, 1979.
- [15] A. Dharwadker, *The Clique Algorithm*, Amazon, pp. 48, ISBN: 978-1466391215, Oct. 2011.
- [16] R. Chang and C. Kuo, "An energy efficient routing mechanism for Wireless Sensor Networks," in *20th International Conference on Advanced Information Networking and Applications (AINA '06)*, Vol. 2, Apr. 2006.