Flow Simulation Based Energy Efficient Clustering in Wireless Sensor Network

Yi JIANG
Dept. Of Communication Engineering, Harbin University of Science and Technology, China
jasonj@live.cn

Keywords: flow simulation, clustering, wireless sensor network.

Abstract. The paper introduces a flow simulation based clustering algorithm for wireless sensor network. The algorithm is fully decentralized. The proposed algorithm is inspired by simulating flow in the network. The algorithm clusters the nodes that are at distance of at most k hops from the cluster head while balance the energy dissipation among sensor nodes. By grouping sensor nodes into clusters, the performance of the mobile network, such as power dissipation, can be optimized. The preliminary simulation results show that the proposed algorithm is efficient.

Introduction

In wireless sensor network, the energy dissipation *e* to transmit a packet to a receiver at distance *d* can be estimated by formula [5]: $e_{tx} = k_1 \times d + k_2 \times d^c$, where c($2 \le c \le 4$) and *k* are constants for a specific wireless system. To receive a data packet at distance *d* the energy can be estimated by formula: $e_{rx} = k_1 \times d$. From those two formulas, we can see that:

- The energy used to transmit messages are larger than to receive messages.
- The energy used is greatly depend on the distance between the source and the destination of the communication channel.

To reduce the energy overhead and prolong the lifetime of the whole sensor network, itwas proposed to partition the network nodes into clusters [5,7,8]. Suppose thereis centroid node in each cluster that are closest to all of the clustered nodeswithin the same cluster, we call this node cluster head.

In our study, we propose a flow simulation based algorithm to clusterthe wireless sensor node. The algorithm is inspired by [1] and [4]. The rest of thepaper is organized as follows. First, the paper describes the algorithms of flow simulation,thenprovides the details about the clustering algorithm in wireless sensornetwork. Finally, a preliminary simulation about this algorithm will be presented.

Related works

MCL(Markov cluster algorithm) [4] is a centralized clustering algorithm forgraphs based on flow simulation. In MCL algorithm, the graph G is mapped to aMarkov matrix. Then the set of transition probabilities is iteratively recomputedvia expansion and inflation. Via expansion, nodes are able to see new neighbors.Via inflation, favoredneighbors are further promoted and less favoredneighborsare further demoted. The heuristic behind this algorithm is that a flowbetween sparsely connected dense regions evaporates after MCL process. Therefore,it is easy to detect dense regions in the original graph, and the dense regionsare returned as clusters. MCL algorithm is considered a scalable and efficientnetworking clustering scheme, which can achieve high clustering accuracy.

CDC(Connectivity-based Distributed Node Clustering) [1] is a distributed andscalable algorithm for discovering connectivity-based clusters in peer-to-peer networks.The idea of the CDC scheme is to simulate flows in the network andit is inspired by MCL. In CDC, the algorithm is starts from

initiating messagesfrom set of nodes that are the originators of the clustering algorithm. The flowis simulated by messages sent from originator.We think this algorithm is scalable and it can handle dynamic network well.

But in wireless sensor network, the connectivity based algorithm isnot well suitable. Because, in such environments, the network topologyis changing because of node dying. Besides, the CDC algorithm consider the peer-to-peer network asan un-weighted graph, which is different to wireless sensor network. In wireless sensor network, the distance between nodes are important parameterswhen selectingthe next hop route.

The flow simulation based clustering algorithm

Thealgorithm is inspired by MCL, and similar to CDC. But there are differences between the two algorithms. The main difference is that the algorithmconsidering the physical distance between nodes along with the hop distance. Let usfirst introduce the requirements and assumptions about the algorithm.

The cluster algorithm running in mobilesensor network must meets the following requirements:

1. Fully decentralized andasynchronized. This is the basic requirement. Theasynchronized requirement is because that the mobile network is more dynamicthan peer-to-peer network.
2. Avoid message overhead in network. Most of the mobile nodes are batterypowered or energy limited device. The more the message forwarded, the morelikely the node dead.
3. Heuristic and producing sub-optimal output. The cluster algorithm is runningdecentralized; only local information can be used to guide the algorithm.The heuristic algorithm is more suitable in this environment and produces sub-optimal output.
4. The selection of the cluster head nodes must consider the load balancing.The cluster head has high work load among all the cluster nodes, so itconsumes energy faster than other nodes, if the cluster head selectionalgorithm cannot balance the work load among clusters, the clusterhead will die quickly. As a consequent, the life time of the whole sensornetwork will be shortened.

Before describingthe algorithm, there are some assumptions about the mobilenetwork and mobile nodes.

1. The network has not been clustered.
2. Every mobile node knows its own position.

If node does not know its position, the node should have the ability tomeasure the signal strength with which node can determine the distance betweenthe other node according to the Friis transmission equation.

As stated previously, the power consumption is largely related to thedistance between local node and neighbors. In our study, the sensor nodeselects the one which is close to the itself while have more residualenergy as the cluster head. The main idea of the algorithm is simulating flowsin the network as [1] does. Image the nodes in the network are intersection ofthe edges(links between nodes), and messages are flows. The message flow in thenetwork, at each intersection, it drops some of its weight and continue its flowat random direction along the links. Considering the following fact: a randomwalk in a graph that visits a dense cluster will likely not leave the cluster untilmany of its vertices have been visited [4]. So we can infer that the nodes within the same density area will receive most of the weights. Then the node that one receives most weighted weights from can be selected as the cluster head.

The algorithm starts out by initiating messages from nodes randomly. Incontrast to CDC algorithm, in the algorithm, there are no such THP(Two HopsProbability) process to determine the originator nodes. In our study, the aim ofthe algorithm is to reduce the energy dissipation, we think the process of theTHP is not needed in sensor network, because the cluster message can be usedto determine the originator as well.At first, nodes are not gathered into clusters. But if there are nodes needto transfer packets to some other nodes, there are flooding in the network. Wechoose to piggy back the cluster message to every packets. In this way, there isno extra packetneeded to transfer cluster message.

Neighbors. After power on, nodes turn on its radio transceiver. At this time,it broadcast a message through the wireless connection. All nodes in the areaof the radio transmission respond to this message. In this way, nodes canknow the neighbors in the sensor network. There are various reasons that a nodewill died without notification in sensor network, so the neighbors of node arechanging at time. Nodes need to probe the network periodicallyto maintain the neighbors' list. This is done by neighbor protocol. The neighborprotocol is very simple.

Cluster message. By neighbor protocol, node knows the nodes in thearea of the radio transmission. Those nodes are the one hop neighbors of thelocal node. The algorithm will to do is to gather all the nodes in onehop distance into clusters. Thus thesensor network can be partitioned into clustersaccording to the one hop distance. The cluster messages send from nodes tonodes are mainly within the one hop distance. Because there may be more thanone node within the one hop distance from local node, the cluster message maybe forwarded more than one time.Similar to the data structures used in algorithm CDC, in the algorithm thereare similar cluster message structure. Each cluster message is a tuple consistingof the following fields:

1. Source ID (SID): A field uniquely identifying the source of the cluster message.
1. Message ID (MID): A filed distinguishing each message from all other messagesfrom the same source.
2. Message Weight (MWeight): The weight carried by the message.
3. Previous Hop ID (PHID): This field indicating the most recent node themessage traversed.
4. Max Hop Distance (MHD): This field indicating the max hop the messagecan be reached. This filed is always set to 2 in our study.

Message weight. The algorithmconsiders not only the distances between nodesbut also the energy distribution of the sensor nodes. Clustering according to thedistances between nodes will reduce the energy dissipation in the sensor network.But the cluster head will out of power more quickly than the cluster member. After several iterations, the network may not alive because many of the nodes are died. Clustering according to the energydistribution only cannot make the energy usage optimal too. So we choose to clusterthe sensor network on those two factors. The initial message weight in thealgorithm is calculated from the following formula:

$$Message.Weight = \frac{Energy_i}{NumNeighbors_i} \tag{1}$$

Where $Energy_i$ is the energy level of the sensor node, which is a real number in [0..1]. $NumNeighbors_i$ is the number of nodes within the one hop distance.

The message weight is designedto drop at the intermediate node j in the traversalpath of the message. The dropped weight and the message weight can be calculated by the Eq.2 and Eq.3:

$$DroppedWeight_j = \frac{Message.Weight}{2^{hops} \times e^{\frac{D_{i,j}}{D_i}}} \tag{2}$$

$$Message.Weight_i = Message.Weight_i - DroppedWeight_j \tag{3}$$

where $hops$is the number of hops the message has traversed, D_i is the effectiveradio transmission diameter, $D_{i,j}$ is the distance between sensor node *i* and *j*.Because node knows its own position, and the position of the source nodecan be carried in cluster message, so the distance between nodes can becalculated simply.

If the reset of the message weight in this cluster message is greater than zero, thenthe message will be broadcasted to all the neighbors of node *j* with message weight as denoted by Eq.4.

$$Message.Weight_j = \frac{Message.Weight_i}{NumNeighbors_j} \tag{4}$$

The algorithm. The pseudo code of the algorithm is shown in Fig.1. The main process of the algorithm is similar to CDC. The differences are the weight computation and the amount of the weight to drop. Obviously, message from sensor node with high energy level carry more weight than others. Those messages drop more weight on nodes it traverses. The node far away from the candidate cluster head node receives less weight from the message. Nodes within the radio transmission range of the high energy level sensor nodes will choose the closest high energy node as cluster head. Those cluster head nodes will use up their energy more quickly than the other nodes. The algorithm will always choose the high energy node as the cluster head.So the energy dissipation is balanced in the whole network, and prolongs thelife time of the sensor network.

```
TotalWeight ← TotalWeight + Msg.Weight
Msg.Weight ← Msg.Weight + Message.Weight / (2^hops × D_{i,j}/D_i)
Msg.Hops ← Msg.Hops − 1
Msg.Range ← Msg.Range − D_{i,j}
if TotalWeight(Msg.SourceID) ≥ Threshold then
        Clusterhead ← Msg.SourceID
else
        if Msg.Hops > 0 and Msg.Range > 0 then
            for all n such that n ∈ Neightbors_i do
                Msg.Weight ← Msg.Weight / NumNeightbors_i
                forward Msg to n
            end for
        end if
end if
```

Fig.1 The algorithm executed on each sensor node*i*when clustermessage received

Candidate cluster heads. This is the last section of the algorithm. Thissection describes the selection of candidate cluster heads. The algorithm describedin Fig.1 is executed by the candidate cluster heads. If all thesensor nodes send cluster messages to all the neighbors, there will be messageoverhead in the network. Before the execution of the cluster algorithm, the sensornodes need to identify itself as the

candidate.The node has more residual energy and low degree is the candidate of the cluster head, and the more residual energy and the lower degree a node has, the more probability the node will be cluster head.Choosing the node with low degree may not wisefor energy-constrained sensor network because it penalizes specific nodes in thenetwork irrespective of their battery lifetimes [6,7]. Choosing the node withhigh degree as the cluster head may result in quickly draining the battery oflarger-degree nodes. In our study, Eq.5 is used to identify the candidate clusterhead.Where *RadioRange* is the radio transmission range of the sensor node. Eq.5 balances the energy level and the density of the sensors.

$$\text{CandidateProbability}_{\text{i}} = \frac{\text{Energy}_{\text{i}}}{\text{NumNeighbors}_{\text{i}} \times \text{RadioRange}^2} \tag{5}$$

In the algorithm,node whose $\text{CandidateProbability}_{\text{i}} = Max_{j \in Neighbors_i} \text{CandidateProbability}_{\text{j}}$ is the cluster headcandidate.

Performance evaluations

The simulation usesa network in which 100 nodes are located randomly in 100m×100m dimensions, andassumes that the energy consumptionis symmetric between two arbitrary nodes in simulation. The base station is located 100m away from the closest sensor node and the base station is considered to be high-energy node.

We use the same radio model as [5] used. That is $E_{elec} = 50nJ / bit / m^2$ and $\varepsilon_{amp} = 100\,pj / bit / m^2$. Each node was initially given 0.5J of energy, and eachmessage are 2000bits.

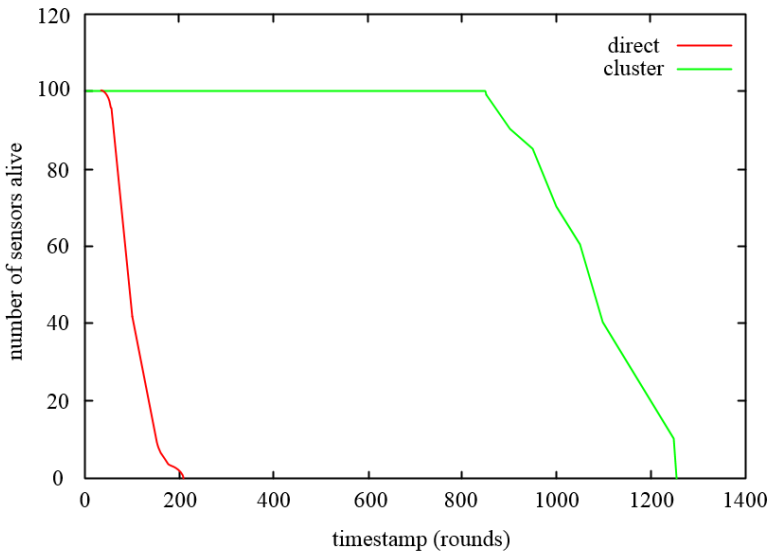

Fig.2 Number of sensors alive

Fig.2 shows the result of the simulation, the first sensor died at850 rounds, the last sensor died at 1257 rounds. The simulationstates that theproposed algorithm has comparable performance to LEACH.

Conclusion

The paperintroduces the flow simulation based clustering for energy efficientdata gathering algorithm for wireless sensor network. The algorithm extends thegeneric connectivity based algorithm for peer-to-peer clustering to accommodatethe wireless sensor network environments. The algorithm in this paper considernot only the distance the cluster member away from cluster head but

also theenergy level of each sensor node. The cluster based communication in wirelesssensor network can save a lot of energy in wireless sensor network, thusprolongs the lifetime of the system. The simulation shows that the flow simulationbased clustering has good performance.

References

[1] L. Ramaswamy, B. Gedik, and L. Liu. Connectivity Based Node Clustering in DecentralizedPeer-to-Peer Networks. Technical report, College of Computing, Georgia Tech, 2003

[2] Li, Q., Aslam, J., Rus, D: Hierarchical Power Aware Routing in Sensor Networks.Proceedingsof the DIMACS Workshop on Pervasive Networking 2001.

[3] Heinzelman, W. R., Chandrakasan A. Balakrishnan, H. Energy efficient CommunicationProtocols for Wireless Microsensor Networks. Proc. Hawaaian Intl Conf. onSystems Science, 2000.

[4] S. van Dongen. A new cluster algorithm for graphs. Technical Report INS-R0010,National Research Institute for Mathematics and Computer Science in the Netherlands,Amsterdam, May 2000.

[5] Heinzelman, W. R., Chandrakasan A., Balakrishnan, H. Energy efficient CommunicationProtocols for Wireless Microsensor Networks. Proc. Hawaaian Intl Conf. onSystems Science, 2000.

[6] Ossama Younis, Marwan Krunz and Srinivasan Ramasubramanian, Node Clusteringin Wireless Sensor Networks: Recent Developments and Deployment Challenges,IEEE Network Magazine (special issue on wireless sensor networking), May 2006

[7] D. J. Baker and A. Ephremides. The architectural organization of a mobile radionetwork via a distributed algorithm. IEEE Transactions on Communications,29(11):1694-1701, 1981.

[8] O. Younis and S. Fahmy. Distributed clustering in ad-hoc sensor networks: A hybrid,energy-efficient approach. In Proceedings of IEEE INFOCOM, Hong Kong, March2004. An extended version appeared in IEEE Transactions on Mobile Computing,3(4), pp. 366379, Oct-Dec 2004.