

HAUSDORFF CLUSTERING AND MINIMUM ENERGY ROUTING FOR WIRELESS SENSOR NETWORKS

Xiaorong ZHU Lianfeng SHEN
the Department of Radio Engineering
Southeast University
Nanjing, China

Tak-Shing Peter YUM
the Department of Information Engineering
The Chinese University of Hong Kong
Hong Kong, China

ABSTRACT

We present a new method for data gathering that maximizes lifetime for wireless sensor networks. It involves three parts. First, nodes organize themselves into several static clusters by the Hausdorff clustering algorithm based on location, communication efficiency and network connectivity. Second, clusters are formed only once but the role of cluster-head is optimally scheduled among the cluster members. We formulate the cluster-head scheduling that maximizes the network lifetime as an integer programming problem and propose a greedy algorithm for its solution. Third, after cluster-heads are selected, they form a backbone network to periodically collect, aggregate, and forward data to the base station, where a minimum energy (cost) routing is used. Comparing with other known methods, significant lifetime extension is obtained with the use of this method.

I. INTRODUCTION

Wireless sensor networks (WSNs) represent a new paradigm for extracting data from the environment for many agricultural, industrial and scientific applications. Typical sensor nodes have limited power supply and therefore power management is a key design issue in WSNs. It is observed that wireless transmission is the main consumption of battery power, and therefore sensor nodes should minimize communication activities through local collaborations among the nodes [1]-[3].

In sensor networks, a typical application is the gathering of sensed data to a distant base station (BS) [4]-[7]. A sensor can communicate directly only with other sensors within its range. To go beyond, sensors need to form multi-hop links. In multi-hop networks clustering is very effective in reducing communications, i.e., the data gathered by the sensors is combined at the cluster-heads before sending to the BS. Clustering is particularly crucial for scaling the network to hundreds or thousands of nodes. Many clustering algorithms have been proposed [8]-[13] for wireless ad hoc networks. Most of these algorithms [8]-[10] are specifically designed for generating stable clusters in mobile networks. But in sensor networks, the locations of nodes are mostly fixed and instability is not important. In sensor networks, clustering is mainly for communication efficiency. A distributed, single-hop clustering algorithm called LEACH is proposed in [6]. The cluster head role is periodically rotated among the sensor node to balance energy consumption. It is assumed that all nodes can perform long distance transmissions to the base station. The authors derived the optimum number of clusters heads by taking into account the energy used by all clusters. HEED [7] is one of the effective data gathering protocols without location support. In HEED, cluster-head selection is

based on the candidate's residual energy and a secondary parameter, such as proximity to its neighbors. The process of clustering is divided into many cycles. In each cycle, some nodes are selected to be cluster-heads while neighboring nodes join the cluster-heads to form clusters. This repeated clustering introduces communication and processing overheads and taxing the sensor energy as a result [6]. On the other hand, a number of non-location based [1]-[7], [14]-[15] and location based [16]-[20] routing protocols have been proposed for sensor networks. Among them, the location based solutions have received more attention due to their inherent scalability and power-efficiency. In most cases location information is needed in order to calculate the distance between two particular nodes so that energy consumption can be estimated. For instance, if the region to be sensed is known, using the location of sensors, the query can be diffused only to that particular region.

The Minimum Energy Communication Network (MECN) is proposed in [16] by using low power GPS. The main idea is to find a sub-network, which requires less power for transmission between any two particular nodes. However, in MECN, it is assumed that every node can transmit to every other node. Geographic Adaptive Fidelity (GAF) [17] is an energy-aware location-based routing algorithm. GAF identifies redundant nodes within each virtual grid and switches off their radios to achieve energy savings. However, the active node in GAF does not aggregate traffic as other hierarchical protocols [6]-[7]. The Greedy Other Adaptive Face Routing (GOAFR) is proposed in [18]. It always picks the closest neighbor for routing towards the BS. However, it can easily stuck at some local minimum (i.e., no neighbor is closer to a node than the current node, namely "dead end"). Another location-based data dissemination protocol is called LAF [19]. The sensor network is divided into virtual grids and each node associates itself with a virtual grid based on location. Sensor nodes within a virtual grid are classified as either gateway nodes or internal nodes. While gateway nodes are responsible for forwarding the data across virtual grids, internal nodes forward the data within a virtual grid. LAF achieves energy savings by reducing the redundant transmissions of the same packet by a node. However, grids are fixed and gateway nodes are also fixed, which leads to gateway nodes consuming much energy and dying earlier.

In this paper, we propose a three-part distributed method for maximizing the network lifetime of a stationary set of nodes with a base station. First, nodes organize themselves into several static clusters by the Hausdorff clustering algorithm that is based on location, communication efficiency and network connectivity. Hausdorff distance is typically used to compute the distance between two node sets and we use it as the clustering metric. Second, a cluster is formed

only once but the role of cluster-head is optimally scheduled among the cluster members. We formulate cluster-head scheduling that maximizes network lifetime as an integer programming problem. Then a greedy algorithm is proposed to select cluster-heads, based on residual energy and proximity. Third, cluster-heads form a backbone network to periodically collect, aggregate, and forward data to the BS. They find the path with the least power consumption by using Dijkstra's shortest path algorithm [21]-[22]. In case of tie, they choose the path with the node having more energy. Therefore, this protocol achieves energy saving by reducing the number of transmissions and using the least power consumption routing. In this paper, we choose HEED [7] and MECN [16] for comparison for their superior performance. They are both distributed data gathering protocols.

The rest of the paper is organized as follows. Section II describes the problem and the network model. Section III presents Hausdorff clustering algorithm. Section IV gives the cluster-head scheduling algorithm. Section V presents minimum cost routing algorithm for wireless sensor networks. Section VI presents the performance of comparison with other well-known algorithms. Finally, Section VII gives the conclusions.

II. PROBLEM STATEMENT AND ENERGY CONSUMPTION MODEL

Consider a WSN with a single base station and N nodes. The location information of nodes may be provided by the GPS [23], or some localization schemes [24]-[25]. Let $V = \{v_i = (x_i, y_i), i = 1, \dots, N\}$ be the set of node coordinates. Let $e_i(0)$ be the initial energy allocation of node i and let L_i denote the lifetime of node i . Let the network lifetime, denoted as L , be the time elapsed until the first node in the network depletes its energy. In other words, $L = \min(L_1, L_2, \dots, L_N)$. The objective of the problem is to maximize L , and this requires the energy of all nodes be used economically and uniformly.

We use an adaptive node energy model proposed in [6]. It includes three parts as follows.

1) Transmit a data unit from node i and node j needs $E_{i,j}^T$, where

$$E_{i,j}^T = \begin{cases} E_{elec} + \epsilon_{fs} d_{ij}^2, & d_{ij} < d_0 \\ E_{elec} + \epsilon_{mp} d_{ij}^4, & d_{ij} \geq d_0 \end{cases} \quad (1)$$

Here, E_{elec} is the energy needed for coding, modulation, filtering, etc. $\epsilon_{fs} d_{ij}^2$ and $\epsilon_{mp} d_{ij}^4$ are energy terms needed for combating path loss.

2) Receive a data unit at node j needs E_j^R .

3) Data aggregation needs E_{DA} .

In addition, we also take power used when idling and sleeping as P_I and P_S respectively.

Note that besides transmission and reception, cluster-heads need to aggregate data and relay them to cluster-heads or directly to the BS. Experiments have been reported on the power consumption of different operations in a TinyOS systems [26]. The experiments show that transmitting a single bit of data consumes 800 times more energy than executing an instruction. The focus of energy management therefore should be on minimizing the number of transmissions.

III. HAUSDORFF CLUSTERING ALGORITHM

In this section, we introduce the Hausdorff distance [27], define the parameters used in the clustering process, present Hausdorff clustering algorithm and discuss the properties of clusters generated.

A. The Hausdorff Distance

Let the set of sensor nodes S be partitioned into M clusters $\{G_1, G_2, \dots, G_M\}$. Let the Euclidean distance between node m and node n be denoted as

$$d(v_m, v_n) = \|v_m - v_n\| = \sqrt{(x_m - x_n)^2 + (y_m - y_n)^2} \quad (2)$$

Then, the smallest distance from node m of one cluster to another cluster G_j is

$$d^*(v_m, G_j) = \min\{d(v_m, v_n) : v_n \in G_j\} \quad (3)$$

The directed Hausdorff distance from cluster G_i to G_j , denoted as $h(G_i, G_j)$, is the largest value for the v_m 's concerned, or

$$h(G_i, G_j) = \max\{d^*(v_m, G_j) : v_m \in G_i\} \quad (4)$$

The Hausdorff distance between cluster G_i and G_j is simply the larger of the two directed distance, or

$$H(G_i, G_j) = \max(h(G_i, G_j), h(G_j, G_i)) \quad (5)$$

Intuitively, if the Hausdorff distance is d , then every node in G_i must be within a distance d from some node in G_j and vice versa. The function $H(G_i, G_j)$ can be computed in time $O(n) \log(n)$ for two clusters with a total size of n [28].

B. Cluster Conditions

Let each node has the choice of six transmission power levels just like the Berkeley Motes [29], say by specifying the standard ioctl() system call. The lowest transmission power level, with a range R_1 , is used to cover the intra-cluster transmission. The higher power levels are for reaching neighboring cluster-heads. Let one of the higher power levels

have a range R_2 . We assume the highest power level should be at least $4R_1$ to meet inter-cluster communication requirements.

To find the network coverage requirement, consider two neighboring clusters G_i and G_j as shown in Fig. 1. The worst coverage condition is that the closest two nodes (i.e. node B and C) from these two clusters are of distance $2R_1$ apart. On the other hand, to maintain network connectivity, the worst condition is that when nodes A and D are respectively heads of cluster i and j . If they are reachable from each other, their minimum transmission range should be $4R_1$ as shown. In order to save energy and decrease inter-cluster interference, we choose $R_2 = 4R_1$.

Let d_i be the minimum distance between node i and all other nodes in the network. To maintain network connectivity, the minimum value of R_1 , denoted as $R_{1,min}$, should be

$$R_{1,min} = \max \{d_1, d_2, \dots, d_N\}$$

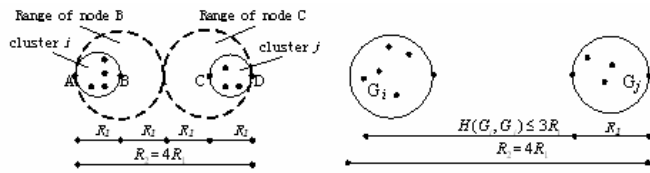


Figure 1: Transmission range requirements.

To summarize, the two necessary conditions for a node to join a cluster are:

- 1) The Hausdorff distance between the node and the cluster must be smaller than R_1 .
- 2) If the node is admitted, the Hausdorff distance between this cluster and neighbor clusters must be no larger than $4R_1$, to guarantee network connectivity during inter-cluster communication. In other words, $H(G_i, G_j) \leq 3R_1$ so as to guarantee $H(G_i, G_j) + R_1 \leq R_2$ as shown in Fig. 2.

C. Hausdorff Clustering Algorithm

We now describe a distributed clustering algorithm where nodes make autonomous decisions. We assume that nodes use traditional RTS/CTS-based collision avoidance mechanisms during clustering. Our goal is to design a clustering algorithm according to clustering conditions mentioned above. These clusters will remain unchanged throughout the network lifetime. Moreover, to evenly use the energy among all the nodes, cluster-head is rotated among cluster members. Details are in section IV.

At the beginning, each node broadcasts a topology discovery message with the lowest power level to find all its neighbors. Then, the base station appoints an initiator for starting the clustering operation. The initiator broadcasts a clustering message and awaits join-requests from neighboring nodes. It then admits cluster members according to the clustering conditions. If the applying node receives an admission message, it sends back a confirmation message.

Upon receiving the confirmation message, the initiator updates its membership list and broadcasts a membership-update message with a higher power level for informing neighboring clusters. If the candidate node is rejected by all its neighboring clusters, it organizes a new cluster with itself being the initiator.

The termination of the clustering algorithm depends on two cases. If it is an initiator, it will terminate the clustering procedure until all its neighboring nodes within the lowest power range have joined clusters. If it is a non-initiator, it terminates the algorithm after successfully joining a cluster. After the cluster is formed, the initiator of each cluster broadcasts cluster membership information to all cluster members for the preparation of cluster-head selection. A summary of the procedures for the base station, initiators and non-initiators is shown in Fig. 3.

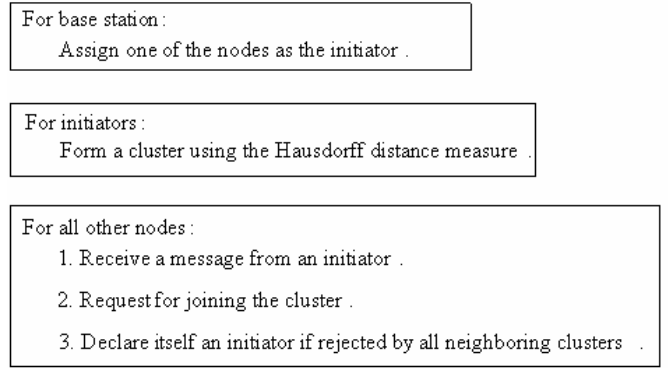


Figure 3: A Summary of procedures for base station, initiators and non-initiators

D. Properties of generated clusters

- (1) All nodes are clustered, or $G_i \cup \dots \cup G_M = S$.

Proof Consider a sensor node. If it is an initiator of a cluster, it belongs to this cluster. If it is an isolated node (having no neighboring nodes), it forms a single member cluster itself. If it is not an isolated node, it must be a neighbor of an initiator and will join that cluster.

- (2) All G_i 's are connected.

Proof For intra-cluster communication, one node can directly communicate with other nodes with the lowest power level. For inter-cluster communication, a cluster-head can directly communicate with neighboring cluster-heads with a higher power level and can reach other non-neighboring cluster-heads by hopping. Therefore, all clusters are connected.

- (3) $G_i \cap G_j = \emptyset$.

Proof The algorithm allows each node to join only one cluster.

IV. THE CLUSTER-HEAD SCHEDULING

Once a cluster is formed, the cluster-head needs to be scheduled. Our goal is to find a schedule that maximizes the network lifetime.

A. Integer Programming Formulation

We assume each node generates a packet in each round, which is collected, forwarded, and eventually transmitted to the base station. Let

$$q_{ij}^s = \begin{cases} 1, & \text{if packet from node } s \text{ goes through link } i \rightarrow j \\ 0, & \text{otherwise} \end{cases}$$

$$i, s = 1, 2, \dots, N, \quad j = 0, 1, \dots, N \quad (6)$$

where node 0 represents the base station.

Node s generates L packets in lifetime L altogether, which are eventually forwarded to the base station by node i . Therefore, we have

$$L = \sum_{i=1}^N q_{i0}^s \quad (7)$$

Let p_{ij} be the total number of packets that node i transmits to node j . It is given by

$$p_{ij} = \sum_{s=1}^N q_{ij}^s \quad i = 1, 2, \dots, N, \quad j = 0, 1, \dots, N \quad (8)$$

According to energy constraints, for each node i we require

$$\sum_{j=0}^N p_{ij} E_{ij}^T + \sum_{j=1}^N p_{ji} E_i^R \leq e_i(0) \quad i = 1, 2, \dots, N \quad (9)$$

Besides, for each node i , packet flow conservation requires the number of packets node i generates in the whole network lifetime plus the number of incoming packets forwarded by node i be equal the total number of packets out of node i . In other words,

$$L + \sum_{j=1}^N q_{ji}^s = \sum_{k=0}^N q_{ik}^s \quad s = 1, 2, \dots, N \quad (10)$$

Therefore, the problem of maximizing network lifetime is equivalent to maximizing $L = \sum_{i=1}^N q_{i0}^s$, subject to constraints (6), (8)-(10).

This is a formidably complex optimization problem given the large number of integer variables. In the following we introduce a greedy algorithm for a heuristic solution.

B. A Greedy Algorithm for Cluster-head Scheduling

In order to maximize the network lifetime, it is necessary to maximize the lifetime of each cluster. Cluster lifetime can be extended by rotating the role of cluster-head among the nodes in a cluster. Cluster-head selection is primarily based on the residual energy. To increase energy efficiency and further prolong network lifetime, we also consider proximity of neighbors as a secondary selection parameter.

We use a four nodes cluster to illustrate this algorithm. Let time be divided into periods of duration T . At the beginning of the first period, the initiator of each cluster acts as

temporary cluster-head to select the new one. It will broadcast a message to request residual energy of each member to prepare for cluster-head selection. At the beginning of other periods, the old cluster-head runs the greedy algorithm to select the node with the largest residual energy as the new cluster-head. Fig. 4 shows that at the beginning of period 1, node 3 is so selected. At the beginning of period 2, node 2 is calculated to have the largest residual energy and therefore is selected as the new cluster-head. Continuing, the cluster lifetime is seen to extend to 6 periods.

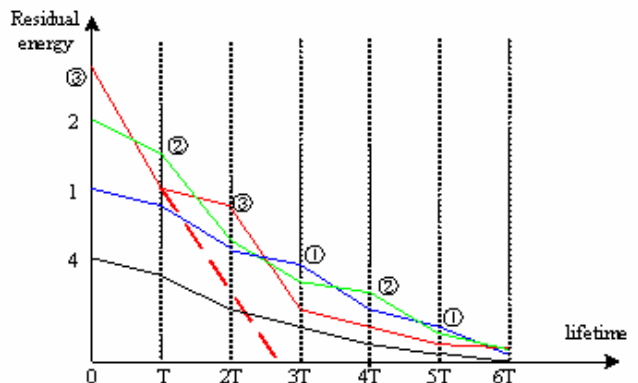


Figure 4: Cluster-head scheduling in a cluster.

In general, let $L(R_1)$ be the network lifetime for given R_1 . The greedy algorithm for cluster-head scheduling is stated as follows:

Begin

$n=1$

Repeat

1. For each cluster, select a node with maximum residual energy as the cluster-head. In case of tie, select the one with minimum root mean-square distance to neighbors.
2. Update the residual energy for all nodes at

$$t = nT$$

$$L(R_1) = nT$$

$$n = n + 1$$

Until (the residual energy of any node is depleted)

End

In each beginning of a round, the old cluster-head carries out the greedy algorithm and selects the new one. After the new cluster-head is selected, the old one will announce with the lowest power level within the cluster. The new cluster-head broadcasts a message with a higher power level to its neighboring cluster-heads and gets itself connected for routing.

V. INTER-CLUSTER ROUTING

After the network is clustered, specific methods for intra-cluster and inter-cluster communications depend on applications. For intra-cluster communication, the nodes can directly send data to the cluster-head using TDMA schedule just as in LEACH [6]. For inter-cluster communication, we

propose a heuristic routing algorithm for the forwarding of traffic from cluster-heads to the base station.

The objective of route optimization is the extension of network lifetime. We apply the distributed Dijkstra's shortest path algorithm [21] among cluster-heads using power consumption as the cost metric. Here, the link cost for link $i \rightarrow j$ is defined as follows

$$C(i, j) = P_t(i, j) + P_r \tag{11}$$

where $P_t(i, j)$ is the power used for transmission and P_r is the fixed power used for receiving.

For tie breaking, i.e., when the costs of several links equal, the node having the most residual energy can be picked. The convergence of the algorithm is guaranteed, as discussed in [21].

After several iterations, the path from cluster head i to the base station can be built, which is the global minimum power path. And the path cost can be computed by the summation of link costs on the path.

VI. PERFORMACE EVALUATION

We have developed a simulator in Visual C++ to evaluate the performance of proposed protocol and compared it with other data gathering algorithms. We distribute 100 sensor nodes randomly in a 100m×100m field. Most of our simulation parameters are similar to those in [7]. The parameters are listed in Table 1. We assume that all the messages received from the cluster members can be aggregated into a single message. We run the data gathering protocols 20 times and average the results. In each run, a randomly selected node initiates the clustering protocol.

Table 1: Simulation Parameters

Type	Parameter	Value
Network	Network Grid	From (0,0) to (100,100)
	Base station	At (50,175)
Application	Initial energy	2J
	R_1	50 m
	Data packet size	100 bytes
	Broadcast packet size	25 bytes
	Packet header size	25 bytes
	Round	5 frames
	T	10 s
Radio model	E_{elec}	50nJ/bit
	ϵ_{fs}	10pJ/bit/m ²
	ϵ_{mp}	0.0013pJ/bit/m ⁴
	E_{DA}	5nJ/bit/signal
	Threshold distance (d_0)	75 m
	P_t	14.88 mW
	P_s	0.016 mW

A. Transmission Range R_1

Network lifetime is the time until the first node dies and is measured in "rounds". A round is of duration T during which a burst of data is collected and forwarded to the base station.

Fig. 5 shows the network lifetime of 8 cases with R_1 ranges from 30m to 100m. Also shown is k , the number of clusters formed for each R_1 . It is seen that maximum lifetime can be achieved from a broad range of R_1 . This is because a smaller R_1 value (smaller intra-cluster communication cost) could result in a larger number of hops to reach the base station (larger inter-cluster communication cost) and vice-versa.

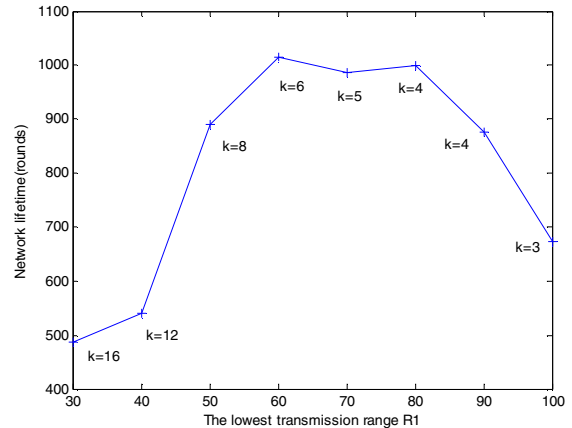


Figure 5: Network lifetime for different value of R_1 .

B. Period Length T

Fig. 6 shows the network lifetime as a function of period length T from 10 to 110 seconds, where the number of frames transmitted in each round is 5. It shows that the network lifetime (in seconds) increases with T. The tradeoff being that a larger T value means larger latency in reporting sensor data to the base station.

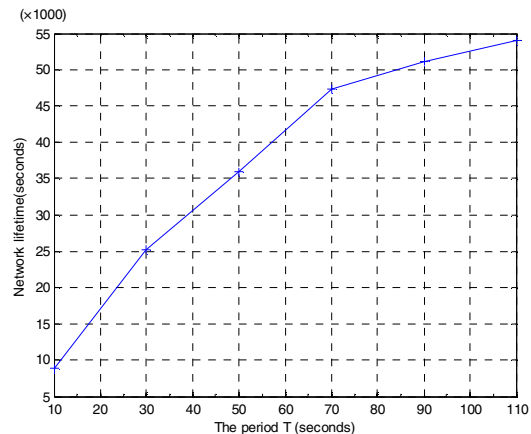


Figure 6: Relation between network lifetime and period T.

C. Performance Comparision with HEED

HEED [7] is one of the effective data gathering protocols without location support. Fig. 7 (a) compares the network lifetime of Hausdorff clustering and HEED for number of nodes ranging from 300 to 600. It is seen that Hausdorff

clustering is uniformly better independent of node density. This is due to the following reasons. First, HEED utilizes a dynamic virtual topology with variable cluster sizes hence incurring higher overhead, while our scheme uses a fixed and simple architecture that eliminates reclustering. Second, in Hausdorff clustering, when cluster-heads forward data to the base station in each round, minimum energy routing is used. In HEED, an ad hoc routing protocol, such as Directed Diffusion or Dynamic Source Routing [7], is used for data forwarding by cluster heads. Fig. 7 (b) compares the “lifetime” until the last node dies. Note that the above results energy is consumed in sleeping and idling states according to Table 1 for Hausdorff clustering. For HEED, these states do not consume energy.

Fig. 8 compares the fractional energy used for clustering. It shows that for extending network lifetime, fixed clustering with rotating cluster-heads proposed in this paper is a better alternative to reclustering.

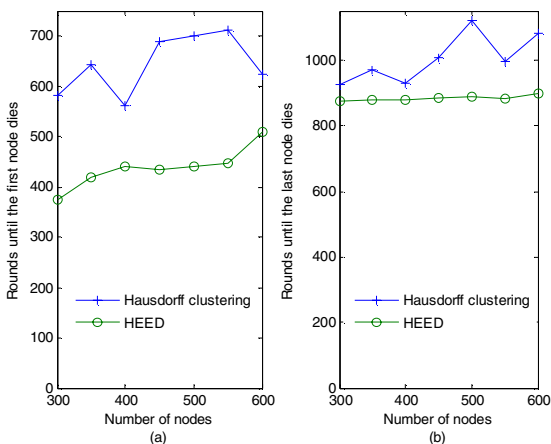


Figure 7: Network lifetime comparison. (a) until the first node dies. (b) until the last node dies.

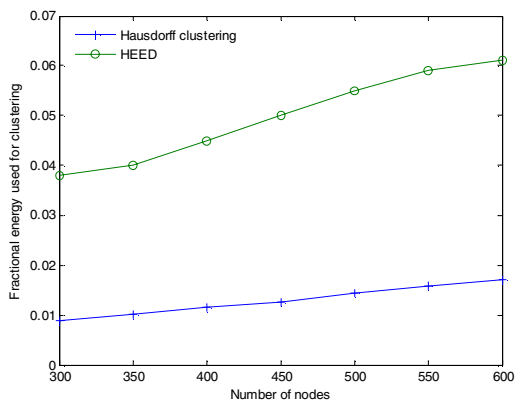


Figure 8: Fractional energy used for clustering.

D. Network Lifetime Benchmark with MECN

Next, we compare Hausdorff clustering with Minimum Energy Communication Network (MECN) [16], a typical location-based data gathering protocol for sensor networks. In the simulation, we assume there are 200 sensor nodes

randomly distributed in a 1000m×1000m field. Each node has the lowest transmission range of 125 meters. Sensors periodically transmit data to the base station located at (1000, 1000). The broadcast control packets are 50 bytes and traffic packets are 512 bytes. The traffic packet rate is 0.5 per second. We assume that each node has an initial energy of 5 Joule.

We compare the two methods by the number of nodes still alive over time. As shown in Fig. 9, Hausdorff clustering performs consistently better than MECN. Several reasons attribute to it. First, as nodes die (due to running out of power), MECN needs to reconstruct sub-network, which consumes much energy. This is not needed in Hausdorff clustering. Second, MECN is not a clustering protocol and therefore cannot reap the benefit of traffic aggregation.

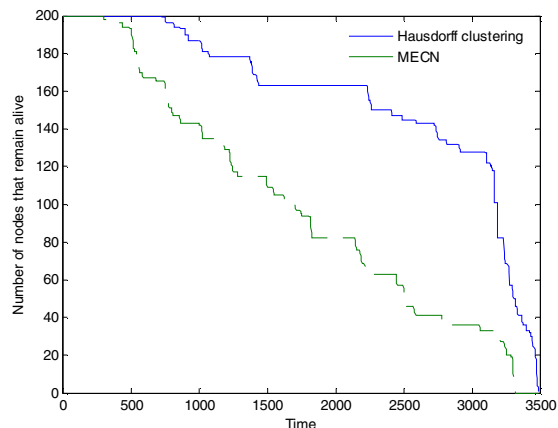


Figure 9: A trace of network lifetime.

VII. CONCLUSIONS

In this paper, the Hausdorff clustering is proposed for wireless sensor networks. Simulation results show that Hausdorff clustering can significantly extend the network lifetime when compared to other approaches. Hausdorff clustering is equally applicable to cases where the sensor node initial energy distribution is non-uniform. The detail performance evaluation of these cases is beyond the scope of this paper.

REFERENCES

- [1] Q. Liang, “Cluster head election for mobile ad hoc wireless network,” *Proc. 14th IEEE International Symposium on Personal Indoor and Mobile Radio Communications*, Sept.2003, pp.1623 – 1628.
- [2] Lindsey S and Raghavendra C.S, “PEGASIS: Power-Efficient Gathering in Sensor Information Systems,” in *IEEE Aerospace Conference Proceedings*, Big Sky, Montana, 2002, vol.3, pp. 1125-1130.
- [3] Yunxia Chen and Qing Zhao, “On the Lifetime of Wireless Sensor Networks,” *IEEE Communication Letters*, vol.9, no.11, pp.976-978, Nov.2005.
- [4] S. Bandyopadhyay and E. Coyle, “An Energy-Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks,” *Proc. IEEE INFOCOM*, Apr.2003, pp.1713-1723.
- [5] Wei-Peng Chen, Jennifer C. Hou, and Lui Sha, “Dynamic Clustering for Acoustic Target Tracking in Wireless Sensor Networks,” *IEEE Trans. Mobile Computing*, vol.3, no.3, pp. 258-271, Jul-Sep.2004.
- [6] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “An Application-Specific Protocol Architecture for Wireless Sensor

- Networks," *IEEE Trans. Wireless Comm.*, vol.1, no.4, pp.660-670, Oct.2002.
- [7] Ossama Younis and Sonia Fahmy, "HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad hoc Sensor Networks," *IEEE Trans. Mobile Computing*, vol.3, no.4, pp. 367-379, Oct.-Dec.2004.
- [8] A. D. Amis and R. Prakash, "Load Balancing Clusters in Wireless Ad Hoc Networks," in *Proceedings of ASSET*, 2000, pp. 25-32.
- [9] A. D. Amis, R. Prakash, T. H. P. Yuong, and D. T. Huynh, "Max-Min D-Cluster Formation in Wireless Ad Hoc Networks," in *Proceedings of IEEE INFOCOM*, 2000, vol.1, pp. 32-41.
- [10] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: A Weighted Clustering Algorithm for Mobile Ad hoc Networks," *Journal of Cluster Computing*, Special Issue on Mobile Ad hoc Networking, pp.193-204, 2002.
- [11] V. Kawadia and P. R. Kumar, "Power control and Clustering in Ad hoc Networks," in *Proceedings of IEEE INFOCOM*, 2003, vol.1, pp. 459-469.
- [12] S. Banerjee and S. Khuller, "A Clustering Scheme for Hierarchical Control in Multi Hop Wireless Networks," in *Proceedings of IEEE INFOCOM*, 2001, vol.2, pp. 22-26.
- [13] M. Gerla, T. J. Kwon, and G. Pei, "On demand Routing in Large Ad hoc Wireless Networks with Passive Clustering," in *Proceedings of WCNC*, 2000, vol.1, pp. 23-28.
- [14] A. B. McDonald and T. F. Znati, "A Mobility Based Framework for Adaptive Clustering in Wireless Ad-Hoc Networks," *IEEE Journal on Selected Areas in Communications*, vol.17, no.8, pp.1466-1487, Aug.1999.
- [15] Johnson D and Maltz D, *Mobile Computing*. New York, Kluwer Academic Publishers, 1996.
- [16] V. Rodoplu and T.H. Ming, "Minimum energy mobile wireless networks," *IEEE Journal on Selected Areas in Communications*, vol.17, no.8, pp.1333-1444, Aug.1999.
- [17] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in *Proceedings of the 7th Annual ACM/IEEE International Conference on mobile Computing and Networking*, 2001, pp. 70-84.
- [18] F. Kuhn, R. Wattenhofer, and A. Zollinger, "Worst-case Optimal and Average-case Efficient Geometric Ad Hoc Routing," *Proc. 4th ACM Int'l Conf. Mobile Comp. and Net.*, 2003, pp. 267-278.
- [19] H. Sabbineni and K. Chakrabarty, "Location-Aided Flooding: An Energy-Efficient Data Dissemination Protocol for Wireless Sensor Networks," *IEEE Trans.Computers*, vol. 54, no.1, pp. 36-45, Jan. 2005.
- [20] Y. Yu, D. Estrin, and R. Govindan, "Geographical and Energy-Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks," *UCLA Comp. Sci. Dept. tech. rep.*, UCLA-CSD TR-010023, May 2001.
- [21] N. A. Lynch, *Distributed algorithms*. San Mateo, CA: Morgan Kaufmann, 1996, pp. 510-80.
- [22] <http://www.cs.sunysb.edu/~skiena/combinatorica/animations>.
- [23] B. Hoffman-Wellenhof, H. Lichteneger, and J. Collins, *Global Positioning System: Theory and Practice*, fourth ed. Vienna: Springer-Verlag, 1997.
- [24] T. Clouqueur, K.K. Saluja, and P. Ramanathan, "Fault Tolerance in Collaborative Sensor Networks for Target Detection," *IEEE Trans.Computers*, vol. 53, pp. 320-333, Mar. 2004.
- [25] N. Patwari and R.J. O'Dea, "Relative Location in Wireless Networks," *Proc. IEEE Vehicular Technology Conf.*, 1991, pp. 1149-1153.
- [26] Samuel Madden, Michael J. Franklin, Joseph M. Hellerstein, and Wei Hong, "Tag: a tiny aggregation service for ad-hoc sensor networks," *Proc. of OSDI*, 2002, pp. 131-146.
- [27] Daniel P. Huttenlocher, Gregory A. Klanderman, and William J. Rucklidge, "Comparing Images Using the Hausdorff Distance," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.15, no.9, pp.850-863, Sep.1993.
- [28] H. At, B. Behrends, and J. Blomer, "Measuring the Resemblance of Polygonal Shapes," *Proc. Seventh ACM Symp. Comput. Geometry*, 1992, pp.102-109.
- [29] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D.E. Culler, and K.S.J. Pister, "System Architecture Directions for Networked Sensors," *Architectural Support for Programming Languages and Operating Systems*, 2000, pp. 93-104.