Efficient Clustering for Improving Network Performance in Wireless Sensor Networks

Tal Anker[1,2], Danny Bickson[1⋆], Danny Dolev[1] and Bracha Hod[1]

[1] The Hebrew University of Jerusalem, Israel
{anker, daniel51, dolev, hodb}@cs.huji.ac.il
[2] Marvell Semiconductor, CA, USA
tala@marvell.com

Abstract. Clustering is an important mechanism in large multi-hop wireless sensor networks for obtaining scalability, reducing energy consumption and achieving better network performance. Most of the research in this area has focused on energy-efficient solutions, but has not thoroughly analyzed the network performance, e.g. in terms of data collection rate and time.
The main objective of this paper is to provide a useful fully-distributed inference algorithm for clustering, based on belief propagation. The algorithm selects cluster heads, based on a unique set of global and local parameters, which finally achieves, under the energy constraints, improved network performance. Evaluation of the algorithm implementation shows an increase in throughput in more than 40% compared to HEED scheme. This advantage is expressed in terms of network reliability, data collection quality and transmission cost.

Keywords: wireless sensor networks, clustering, belief propagation

1 Introduction

Organization of large multi-hop wireless networks into clusters is essential for achieving basic network performance. In wireless sensor networks (WSN), the clustering is primarily characterized by data aggregation by each cluster head, which significantly reduces the traffic cost. The hierarchial model requires two main methods: (1) periodic selection of cluster heads (CHs); and (2) assignment of each node to one or multiple clusters.

Optimal clusters' selection is equivalent to the minimum dominating set problem which is an NP-complete problem. The literature is extremely rich with many approximation algorithms based on several heuristics. The reader is referred to [1] and [2] for a review of previous work.

While most efforts thus far have focused on an energy-efficient clustering scheme, the attention to the performance of the multi-hop network was quite

⋆ Danny Bickson has been partially supported by EVERGROW, IP 1935 of the EU Sixth Framework.

limited. An energy-efficiency algorithm may select a few CHs for energy-saving, but if these CHs do not have good connectivity or if they are not stable, the retransmission and the dropped packets may significantly degrade the network performance and the total energy wasted may end up to be higher. Therefore, taking reliable communication into account is essential for any clustering algorithm which aims to reduce the energy consumption in a network.

Moreover, the network lifetime should be measured not only by the time that the first or the last node dies, but also by the period of time that the network is available for providing services and operating appropriately. Since the network is usually dense and many nodes are redundant, the death of a few nodes does not affect the network. Thus, network lifetime is tightly coupled with the network performance.

The work presented in this paper uniquely addresses the clustering problem in multi-hop networks with a special focus on network performance, using the belief propagation (BP) algorithm. BP is an iterative algorithm for computing marginal probabilities on trees, by local message passing [3]. Mostly, it is used for efficiently solving inference problems. BP is a popular method for distributed inference because of its properties, such as fast convergence, accurate results, and good performance in asynchronous environment etc.

The main advantage of this method over existing algorithms for clustering is that BP considers not only local properties of a node, such as residual energy or degree, but also takes into account joint characteristics of a group of nodes, such as link quality and topology information. Utilization all available data, while maintaining small constant message and time overhead, leads to considerable increase in network performance and balanced power consumption among the nodes.

The contribution of the paper is two-fold. First, it introduces a new algorithm for efficient clustering that considers not only the power balancing among the nodes, but also the total transmission power aggregated in the multi-hop routing. The algorithm is fully decentralized and asynchronous, have fixed small convergence time and scales to large networks. Extensive simulation of the algorithm in environment of interferences, packet loss and node failures, which covers other synchronization issues, such as active node's duty cycle, demonstrates its robustness as well. In contrast to many algorithms in this area, our algorithm makes no a priori assumptions regarding the network size and distribution of nodes, link symmetry or topology.

Moreover, the paper presents a scalable and practical implementation of BP in WSN for inference goals. We propose a new broadcast variation that is tailored to fit Min-Sum algorithm, efficient implementation in hardware and effective network transmission. The message passing routine is highly energy-aware and provides distinctive combination of energy-efficient features. Our novel approach of using a broadcast communication paradigm and the use of only integer calculations, without any scheduling or message ordering, considerably decrease the general overhead relative to other BP frameworks that are used for WSN ([4], [5], and [6]).

The rest of the paper is organized as follows. Section 2 briefly presents relevant previous work. Section 3 describes the network model and formalizes the clustering problem. Efficient clustering, using belief propagation, is described in Section 4. Section 5 analyzes the algorithm using simulation. Section 6 concludes the paper with a discussion and directions for future work.

2 Related Work

Many research projects in the last few years have explored clustering in WSN from different perspectives. LEACH [7], is the first clustering algorithm that was proposed for reducing power consumption. In LEACH, the clustering task is rotated among the nodes, based on duration. Direct communication is used by each CH to forward the data to the base station (BS).

HEED [8] extends the basic scheme of LEACH by using residual energy and node degree or density as a metric for cluster selection to achieve power-balancing. It operates in multi-hop networks, using an adaptive transmission power in the inter-clustering communication.

Both schemes are fully-distributed, terminate in constant number of iterations and incur low message overhead. However, the cluster selection deals with only a subset of parameters, which can possibly impose constraints on the system. These methods are suitable for prolonging the network lifetime rather than for the entire needs of WSN.

VCA [9] is a voting-based clustering algorithm that enhances the criteria for cluster selection and combines load balancing consideration together with topology and energy information. VCA addresses inefficient cluster formation using a voting scheme, which enables the nodes to exchange information about their local network view. This method assumes a synchronization among the nodes. Similar to WCA [10], the time required for the nodes to gather information about all other nodes depends on the network size and is not constant.

In EEUC [11], the hot-spot problem in multi-hop networks is solved using clusters with unequal size. CHs that are closed to the BS tend to die faster, because they relay much more traffic than remote nodes. Setting smaller cluster sizes to the close CHs preserves their energy. Additional improvement for multi-hop networks is presented in [12], using a separation between the data gathering and aggregation task and the forwarding task.

All these algorithms try to prolong the network lifetime and to balance the load among the nodes, using some metrics for cluster selection and maintenance. Network performance of a multi-hop network is beyond the scope of these papers. A broader perspective is presented in [13], where three fundamental characteristics of multi-hop networks are clarified: power consumption distribution, the effect of the distribution on data collection rate, and data collection time. This work examines the network performance of direct communication, LEACH and HEED. It provides new metrics for measuring the quality of a clustering algorithm in multi-hop WSN. These metrics are used for evaluation of our algorithm as well.

3 System Modeling and Problem Formulation

We model the sensor network as a directed graph $G = (V, E)$, where V is a set of nodes, where each one is assigned a local unique identifier. E is a set of wireless links connecting two adjacent nodes. Nodes are defined as adjacent if and only if they are within each other's transmission range. The links may be asymmetric. A special node, v_0, is defined to be the base station (BS). The BS is distinguished from other nodes by its unlimited energy supply. The network is multi-hop, where nodes closer to the BS relay traffic of other remote nodes and probably consumes much more energy [11]. There are no assumptions about the distribution of the nodes, their homogeneity, location information etc.

The challenge of a clustering scheme is to efficiently form and maintain a connected disjointed groups of nodes in a local and distributed manner. Each group contains a single leader and several ordinary nodes.

The connectivity requirement may be achieved using one of two basic methodologies: either by an adaptive transmit power, where the CH increases its transmission power to reach the next CH or by the assignment of a set of nodes, covered by several CHs, to be gateway nodes. In this work, the second approach is used. This approach is more general because it does not assume any distribution of the nodes and it also takes into consideration interferences in the area.

An efficient scheme is used to select CHs that: (1) minimize the total transmission power aggregated over all nodes in the selected path; (2) balance the load among the nodes to prolong the network lifetime. These two requirements may contradict; e.g. a long path that consumes more energy than a short path may be selected in order to avoid battery depletion at some nodes. The network performance itself is obtained, in part, by the first requirement, where minimizing the total transmission cost results in a decrease of retransmissions as well as the data transmission time.

In order to achieve a scalable and feasible framework, the overhead of the scheme should have a constant message and time complexity per node, with low maintenance cost. Additionally, it should work well under constraints as topology changes, asynchronous environment, failures and duty cycle.

4 Efficient Clustering using Belief Propagation

The idea of using BP for clustering was recently introduced in [14]. The affinity propagation method was set in that paper in a very general context and not in a practical manner for WSN. In this section we construct a novel BP framework for WSN and describe the algorithm for clustering.

4.1 Belief Propagation

In a probabilistic graphical model, an undirected graph $G = (V, E)$ is a set of nodes V and arcs E which represent dependencies among random variables. We denote by x_i the variable representing the set of possible states of a node i. $\psi_i(x_i)$

corresponds to a local (prior) distribution function of node i and $\psi_{ij}(x_i, x_j)$ refers to a joint function of two connected nodes i and j. These functions are also called potential functions.

In the BP method [15], [16], the inference is carried out in a local and distributed manner by each node, using a message passing technique. $m_{ij}(x_j)$ is a message from node i to node j about the state that node j should be. Node i calculates the massage using previous messages it receives from its adjacent neighbors $N(i)$. The message update rule performed by a node i in round t is:

$$m_{ij}(x_j)^t = \sum_{x_i} \psi_i(x_i)\psi_{ij}(x_i, x_j) \prod_{k \in N(i) \setminus j} m_{ki}(x_i)^{t-1} \ .$$

The update rule refereing to state x_j of node j is a sum over all the possible states x_i of node i. On each state, three elements are incorporated together: the local prior information $\psi_i(x_i)$, the joint function $\psi_{ij}(x_i, x_j)$ and the direct neighbors information $m_{ki}(x_i)^{t-1}$.

Upon termination, after round $\bar{t}$, the belief at a node i (the marginal of the variable) is the product of the local evidence together with all the incoming messages and a normalization constant α:

$$b_i(x_i) = \alpha\psi_i(x_i) \prod_{k \in N(i)} m_{ki}(x_i)^{\bar{t}} \ .$$

The BP algorithm for trees is an exact inference algorithm, which means that the belief converges to the correct marginal values in a finite number of iterations equals to the diameter of the tree.

Min-Sum Algorithm For energy efficiency, a variation of the original BP algorithm, known also as the Min-Sum (MS) [17], is used. This algorithm uses only addition and subtraction operations, so it works well with integer values and saves the overhead of floating-point calculations. Additionally, the algorithm uses broadcast messages [18], in order to preserve communication resources.

The MS algorithm computes inference in the negative log domain, which can be equivalently viewed from the physics point of view as an energy, or cost minimization. According to this viewpoint, the goal of the MS algorithm is to minimize the overall cost over all the nodes in the network, based on the local cost functions and the constraints between the nodes. The algorithm is intuitive. Each node transmits to its neighbors a message with its local and joint costs. Each neighbor that receives the message updates its own belief accordingly and transmits the new belief, so gradually the information is propagated through the network until the nodes converge to a common belief. This convergence point minimizes the overall cost in the network. The algorithm, in its broadcast form has three basic steps:

1. *Message Passing*
   Each node i transmits its local evidence on the initial round, and its belief,

based on incoming messages, on the successive rounds. Every broadcast message m_{i*} from node i includes a combined information for all its neighbors, replacing multiple unicast messages. The receivers extract the information intended for them.

$$m_{i*}(x_i)^0 = \psi_i(x_i) \ ,$$

$$m_{i*}(x_i)^t = \psi_i(x_i) + \sum_{k \in N(i)} m_{ki}(x_i)^{t-1} \ .$$

2. *Message Update Rule*
Upon a reception of message $m_{j*}(x_j)^t$ from node $j \in N(i)$, node i updates its local belief by extracting the unicast information from the broadcast message of node j, using the following calculation:

$$m_{ji}(x_i)^t = \min_{x_j}\{\psi_{ij}(x_i, x_j) + m_{j*}(x_j)^t - m_{ij}(x_j)^{t-1}\} \ .$$

The value of every message at round $t < 0$ is 0.

3. *Belief Calculation*
At the end of round $\bar{t}$, where $\bar{t}$ can be chosen to be the network diameter or any other predefined limit, node i determines its final state x_i to be the one which minimizes the total cost.

$$b(x_i) = arg \min_{x_i}\{\psi_i(x_i) + \sum_{k \in N(i)} m_{ki}(x_i)^{\bar{t}}\} \ .$$

4.2 Cost Metrics

Basic metrics for energy-efficient and reliable communication are formulated in [19] for minimum energy path and maximum lifetime. Their analysis shows that an incorporation of the link error rates is required for reliable packet delivery, in both constant-power and variable-power scenarios. Using a similar method, two cost functions are defined. These cost functions consider residual energy, degree, topology and link quality, distance from BS (in terms of hops) and overall transmission cost, as the following.

A self cost of a potential CH is denoted by $C_i = \frac{E_i}{B_i}$, which is basically defined by the expected energy consumption in a period E_i and it's residual battery power B_i. The expected energy consumption is an estimation of the power used in the routing if that node becomes a CH. The estimation is based on the network topology: the degree of the node determines the expected reception and transmission; the distance from the BS in terms of hop count estimates the further transmission cost to the BS.

Transmission cost among two nodes or along a path is a function of the radio power level and the number of transmitted bits. Previous work [20], [21] has shown that the overall transmission cost cannot be estimated by the distance between the nodes, e.g. because of interferences, nor can be estimated by the received signal strength indicator (RSSI), due to in-correlation between low RSSI and reception rate. Link quality can evaluate the expected number of transmissions along the path. Each node estimates the quality of the links by observing packet success and loss events. Accordingly, the transmission cost between two neighbors $C_{ij} = \frac{E_{ij}}{B_i}$ is defined as a function of the energy consumption over the link E_{ij} and the remaining battery power of the transmitting node B_i.

4.3 Algorithm Description

Let x_i be a CH candidate of node i, i.e. $x_i = i$ or $x_i \in N(i)$ and x_i has a valid route to the BS and appropriate link quality.

We define $\psi_i(x_i)$ to be a local cost function of connecting node i to CH x_i.

$$\psi_i(x_i) = \begin{cases} C_i \;\; for\ x_i = i \\ C_{ij}\ for\ each\ x_i = j \in N_i \end{cases} .$$

$\psi_{ij}(x_i, x_j)$ represents the constraints between two neighbors i and j to eliminate improper assignment of CH association. The constraints are: (1) two neighbors cannot be both CHs; (2) a node can select another node to be its CH only if that node announces that it is a CH.

$$\psi_{ij}(x_i, x_j) = \begin{cases} \infty\ one\ of\ the\ constraints\ is\ applied \\ 0 \;\; otherwise \end{cases} .$$

Cluster selection is possible at each node after a period of initialization, when a route to the BS is constructed. The process is asynchronously triggered by two events: (1) when a regular node does not find a CH among its neighbors, e.g. because of topology changes; and (2) periodically, by a CH, to balance the power among the nodes in a local area. The second event also ensures that the number of CHs will not be too large, by preventing a CH from assuming that role if it is not re-selected.

The message passing algorithm is performed on a tree structure, which is a sufficient condition for convergence. The algorithm is executed in a restricted region of a 1-hop neighborhood, and as a result, it requires a constant number of messages. It stabilizes when the entire network is not affected by local changes anymore. The tree is a subtree of the general routing tree that is used in the network. In the first event, once a node triggers a clustering process because of no CH, it announces itself as a temporary CH and its 1-hop neighbors, which get its message and find it as an appropriate CH, selects it as a parent and performs the message passing on the resulting 1-hop tree. In the second event, the node is already a CH, so the message passing tree is already constructed, where all the children of that node participate in the message passing.

Each node i starts the process by broadcasting the message $m_{i*}(x_i)^0$. This message contains its cost for being a CH (infinite if it is not a valid CH) and the cost to connect other CH candidates among its neighbors. These costs are transmitted as 16-bit integer numbers together with 16-bits of identification.

The rest of the packet processing is performed according to the MS algorithm described above, where unordered messages are stored in a buffer until computation. The timer between the rounds is large enough to support asynchronous operation, but not too large, for not to adversely impact effective operation. Topology changes during the message passing are taken into consideration as follows: (1) Cost of new neighbors is not added in the middle of the message passing operation; (2) Node who loses its parent during the message passing cannot converge with its new parent, so all its messages are ignored. The node should wait until the end of the process to find out a new CH; (3) Link breaks are marked by updating the joint cost to be infinite. A node determines which of its neighbors are in its routing subtree by inspection the messages of its parent and its descendants. A node discards cost information of nodes that are not in its subtree, because it does not have complete information about them. Messages with errors or those which are not synchronized with the messages of the node, are discarded as well.

One round before termination, a node calculates the belief about its final state - a CH or an ordinary node, and attaches the appropriate announcement to the message. After the last round a node operates according to its announcement; If it has previously announced itself as a CH it becomes a CH. Otherwise, it joins the cluster that minimizes the overall cost, according to the information it holds. In case of errors or convergence problem, it is possible that no node would declare itself as a CH. In such a scenario, nodes that do not have any alternative CH in their area start the clustering process again.

In contrast to the cost messages, which are propagated over the routing tree to avoid loops, the decision of a selected cluster is made by the information spread in the entire 1-hop neighborhoods, i.e. a node can select a CH that does not appear in its current subtree. Each node updates its clusters map according to all the broadcast messages it gets.

Once the clustering process is done, the routing tree is changed, where CHs operate as parents of the nodes who join them. Using the gateway approach to connect two clusters, a CH may choose a regular node to be its parent, if it does not have any CH that could operate as its parent. The hop metric is used to detect and avoid cycles, so after the process there is a new routing tree.

Convergence Time BP has a fast convergence property, but when too many errors are involved, it is likely that the convergence will be more slow and into a wrong value. WSN are exposed to a large amount of communication and node failures, so the convergence to a correct state is not guaranteed. Therefore, in order to avoid impact of the physical and the MAC layers as well as other environment factors, we limit by design the number of rounds until termination to be a predefined small fixed value. On ideal environment, the convergence of the algorithm to a common belief, not including the CH announcement, is 2 rounds, equal to the diameter of the 1-hop vicinity graph. Actually, the predefined round number was set to 5. This value is robust against some synchronization and packet loss and it is sufficient in most of the cases to reach a convergence via three steps: detection of the nodes in the routing tree for correct cost calculations, computation of the belief based on cost functions and publication of the CH announcement. This number of rounds is very small in compared to other schemes and is not affected by the network size, therefore providing a scalable solution in large networks. Moreover, the limitation on the number of messages means low delay and small message overhead.

Main

(1) If CH and timer expires or if ordinary node with no CH
  (1.1) Start clustering process with propagation limit of 1;
(2) Upon reception a first-round BP message from parent or from CH candidate and when the propagation limit is 1
  (2.1) Update your parent to be the sender node for the message passing;
  (2.2) Start clustering process with propagation limit of 0;

Clustering Process

(1) Compute local cost function and joint cost function of all the neighbors;
(2) Run the MS algorithm with the following rules:
  (2.1) Unordered messages will be stored in a buffer until computation;
  (2.2) Upon topology changes update the cost;
  (2.3) Messages with errors or synchronization problems are discarded;
(3) One round before termination attach the belief about final state to the message;
(4) Ending steps:
  (4.1) Set the power level according to the final state and update timers;
  (4.2) Select a parent: if ordinary node, select CH that minimizes the cost; if CH, select other CH if possible, otherwise choose an ordinary node as a gateway.

Fig. 1. Sketch of the Algorithm

5 Performance Evaluation

To evaluate the performance of clustering using BP, it has been compared with the clustering process of HEED [8], in a network model that uses gateway nodes to connect between the clusters, when two CHs cannot communicate directly.

In HEED, a node initially sets its probability to become the CH according to its residual energy. During each iteration, a node arbitrates among the CHs announcements it has received to select the lowest cost CH. If it has not received any announcements, it elects itself to become a CH with probability it has. If successful, it sends an announcement indicating its willingness to become CH. The node then doubles its probability, waits for a short iteration interval, and begins the next iteration. A node stops this process one iteration after its probability reaches the value of 1. Simulation results have shown that HEED is effective in prolonging the network lifetime and in supporting scalable data aggregation.

5.1 Simulation Model

TOSSIM, TinyOS simulator [22], was used for the analysis of the clustering algorithm. Link Estimation and Parent Selection (LEPS) [21] was used as the routing protocol in the multi-hop network. In this method, each node monitors all traffic received within the single hop range, including route updates from neighbors. Using shortest path heuristic, it manages the nearest available neighbors and decides the next hop. The Surge application was used for data aggregation, where every nodes periodically takes light sensor readings and sends them over the network to the BS. The simulator provides an environment which includes realistic properties of a network, like interferences and collisions, asymmetric links, changes in the link quality, nodes death and failure etc.

Evaluation of the communication cost, as well as the estimation of the remaining energy, were done based on the power information about Berkeley Mica2 mote [23] and using the credit point system, proposed by [24]. In this system, every node is assigned some number of points that reflect its residual energy. Each packet reception or transmission reduces points from the node, based on the packet size and the transmission power level.

Every plot was taken as an average of 27 different runs. In all the experiments, 250 nodes including a single BS were run. The simulated time was 20000 seconds, to observe the network in a stable state until it collapses when the major of the nodes die.

Every node starts with a random residual energy, ranges from 250 to 500 thousand points. The power level of a regular node was -20 dBm and the power level of a CH was -13 dBm. A timer of 540 seconds was set for periodic cluster selection triggered by each CH or by each node in BP and HEED, accordingly, and a timer of 11 seconds was used between the rounds of the message passing. Both the power levels and the timers are the default parameters used by HEED in TinyOS. We adapted the transmission rate and the aggregation rate to the network size, so the transmission rate by the application was increased to 6144 milliseconds. Every CHs that receives the packets aggregates them and transmit them every 3 minutes. The other parameters are taken to be the defaults defined in TOSSIM.

5.2 Network Performance

We first study the network performance of the two algorithms, in terms of data packets received by the BS. Each node constantly transmits data points to its CH which aggregates all the points into a single packet and forwards them toward the BS.

As one can see in Figure 2, clustering with BP achieves more than 40% higher throughput than HEED, where the data points received by the BS are significantly greater. This higher throughput is expressed by both data collection rate and time.

The trend of the data rate during the network lifetime is shown in Figure 3. In Figure 3(a), there is an increase in the data rate over time, both because the

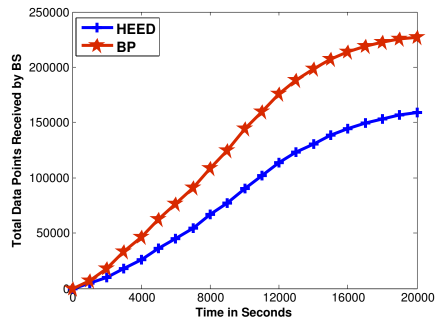
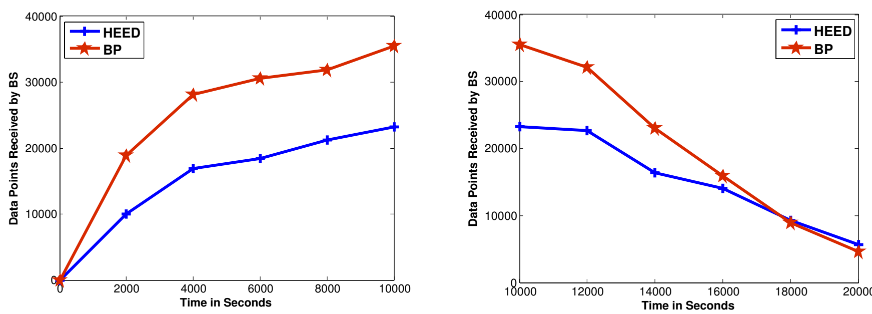


Fig. 2. Data collection time



(a) Data collection rate (b) Deterioration of data collection rate

Fig. 3. Data collection rate during the network lifetime

network becomes more stable and also because nodes start to die, so the network experiences less interferences. The number of live nodes in the system decreases to about 150 nodes at time 10000, but the network is still well connected and only the nodes' redundancy is removed. From this time, the nodes die quickly, so the connectivity of the network and its coverage rapidly decrease. Since the data rate of BP is larger than HEED, the deterioration is steeper.

The advantage of BP can be explained by several network parameters, which are all a result of the fact that BP selects CH better. The non-optimized routing of HEED can be shown by the average hop count of HEED, as presented in Figure 4 which is larger than BP. This means that the number of transmissions in the network may increase, so the number of interferences and the dropped packets increase as well.

Better deployment and network stability may be another reason for the advantage of BP over HEED. The estimated number of CHs in the system during each period of time is presented in Figure 5. Each period is about 540 seconds, with a single periodic clustering process. The figure shows the network state from the beginning with 250 nodes, until about 150 nodes are left, at which

Time (s)	HEED	BP
2000	5.11	3.04
4000	4.11	2.83
6000	3.61	2.73
8000	3.93	2.67
10000	3.88	2.55
12000	3.39	2.51
14000	3.81	2.44
16000	3.58	2.42
18000	3.46	2.38
20000	3.51	2.36

Fig. 4. Average hop count

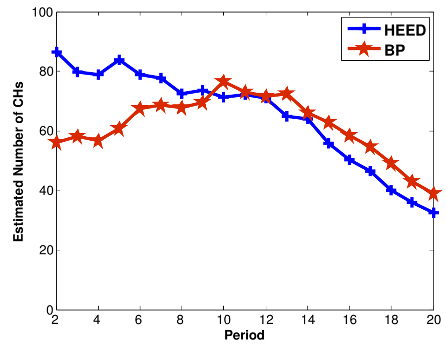


Fig. 5. Estimated number of CHs

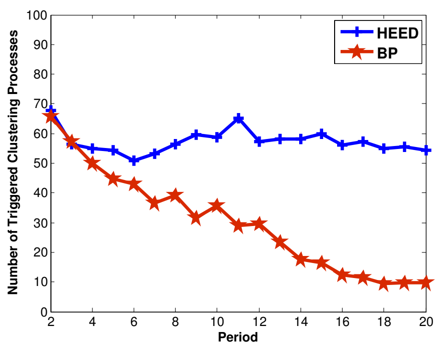


Fig. 6. Triggered clustering processes

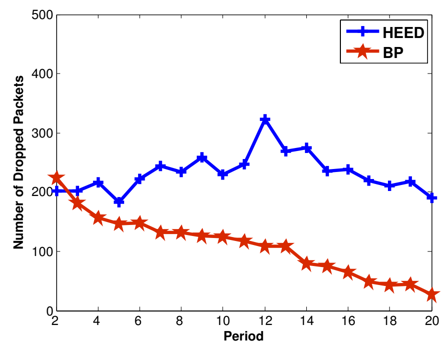


Fig. 7. Dropped packets

point (in periods 11-13) nodes start to die. At the beginning, BP has less CHs in the system which implies better aggregation, less transmission cost and interferences. Once nodes start to die, the number of CHs selected by BP in the system increases proportionally to the number of nodes that are alive and to the number of CHs which are selected by HEED. The intersection of BP and HEED in periods 11 and 12 is a result of the decrease in the number of CHs in HEED and the increased number of CHs, in proportion to the number of alive nodes, by BP. The increased number of CHs achieves better coverage and deployment and improves the network connectivity. The network with BP performs better even under conditions of topology changes, so as a result, less clustering processes are performed and less route failures exist, as it shown in Figure 6 and Figure 7.

The number of clustering processes that are triggered in HEED increases somewhat in period 11, which can be explained by the fact that nodes start to die, and consequently some of the nodes lose their CHs. Nonetheless, with the exception of that increase, during most of the duration, the number of clustering processes that are triggered is quite similar, even during the periods when there are much fewer nodes alive. This means that the network has proportionally more

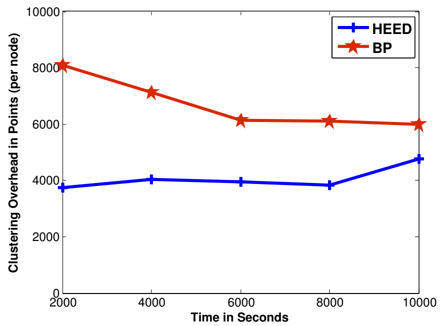


Fig. 8. Clustering process overhead

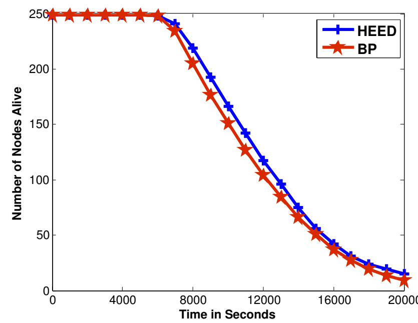


Fig. 9. Network lifetime

clustering processes and that it is not in a stable state. On the other hand, the number of clustering processes that are triggered in the BP scheme decreases over time, which shows better stability even when nodes die. The number of packets that are dropped because of no route, correlates to number of clustering processes that are triggered because of no CHs, and presents the same trend.

It is important to note that no retransmission is done in the simulation. When retransmission is performed, HEED is expected to perform much worse than BP, since retransmission means more interferences and more energy consumption.

5.3 Clustering Overhead

Although BP and HEED have both a constant and consistent number of rounds in the clustering process, BP suffers from more overhead during the clustering process. This is because the messages of BP are larger than HEED. BP messages, at the extreme, might reach up to 74 Bytes (17 cost entries with identification of total 4 Bytes plus header of about 6 Bytes), while HEED message have size of 29 Bytes at most. In fact, BP messages are usually not that long, and do not reach that limit, but still the messages are longer than HEED, so the transmission cost is higher.

Figure 8 shows that at the start of the simulation, the overhead of BP is about double the HEED overhead. Later, when the network becomes more stable, BP performs less re-clustering than HEED. HEED performs more because nodes die, so this difference significantly decreases.

5.4 Energy Characteristics

Network Lifetime BP achieves better network performance and reduces the transmission cost as well. However, the network lifetime, measured by the number of alive nodes of BP and HEED are quite similar, with a marginally (very small) advantage of HEED, as presented in Figure 9. This results from the fact that the total number of packets that are forwarded in the network is significantly greater in BP than HEED. This implies a higher total transmission cost.

Avg. hop count	Avg. number of nodes	Avg. initial energy (points)	Avg. lifetime (seconds)
1	13	374933.48	11673.37
1.5	14	394017.97	12727.69
2	57	388017.87	13047.62
2.5	81	388938.62	12546.50
3	76	335469.51	9587.23
3.5	8	292071.86	7800.46

Fig. 10. Energy information about the nodes in BP

BP pays for transmission of a single packet much less than HEED pays, as a result of the CHs' selection but over the network lifetime the overall transmission cost is similar.

When measuring the network lifetime as the time that the network is available for providing services, we can see in Figure 3(b) that BP succeeds in achieving better performance than HEED, until very close to the end. Only from time 18000, HEED has a slight advantage in the throughput, but this has no real meaning because there are about 20 nodes in the network and anyway the network does not operate appropriately. Therefore, from service availability point of view, BP has better overall network connectivity than HEED and thus better network lifetime.

Power and Load Balancing In multi-hop communication, the nodes closest to the BS usually tend to be burdened with a heavy relay traffic load and to die first. This is the hot-spot problem and many clustering algorithms suffer from it. To verify that this problem does not occur in BP, we analyze the energy characteristics of the nodes based on their distance from the BS.

A node with some physical distance from the BS can have different hop distances over time. For example, a node with distance 1.5 from the BS, can sometimes be connected directly to the BS and sometimes connected via a CH. The different hop count is mostly a result of link quality, which is affected by many network parameters.

We explore on the general concepts that arise from Figure 10 and not from the specific values, since the nodes start with a random initial energy, which definitely affects the network lifetime, even when power balancing takes place.

As shown, both, nodes that are very close to the BS, with distance 1-1.5 and more remote nodes, with distance 2-2.5 (that start with comparable initial energy) have similar lifetime. This means that the BP method succeeds in achieving power-balancing in the core of the network and it does not suffer from the hot-spot problem.

It is interesting to see that more remote nodes (distance 3-3.5) not only start with significantly less residual energy, but their lifetime is shorter. The reason for the initial low energy is that nodes with low residual energy usually would not be selected as CH, and this means that their average distance is larger because they are constantly connected to a CH one hop farther. The explanation for the short lifetime of those nodes, in general, is that they are located at the edge of the network. Nodes at the edge, usually have less neighbors and less chance for having CHs around, so they experience more topology changes and usually perform further frequent clustering processes, which result in more overhead as well. This overhead have a considerable effect on the nodes' lifetime.

6 Conclusions and Future Work

This paper presents a novel distributed inference scheme, based on BP, for efficient clustering in multi-hop WSN. This inference scheme selects CHs that minimize the overall transmission cost and at the same time balance the power among the nodes, for a longer network lifetime. Utilization of all available information, is more optimal than current solutions, and leads to a significant improvement in the network performance.

Using simulations, we show that the BP algorithm succeeds in improving the data transmission time and rate, so at the same network lifetime as the HEED scheme, the overall throughput of BP is increased by more than 40%. Moreover, clustering using BP mitigates the hot-spot problem by providing power and load balancing among the nodes.

The BP framework that has been proposed is a feasible and realistic inference scheme, and can be effective for many other applications. The special attention to energy constrains and the fact that no assumptions were made regarding the network topology or size, differs this framework from other schemes for WSN that are based on BP, and makes it more practical and scalable to large networks with their dynamics.

Comparing the BP algorithm with an optimal clustering algorithm and applying methods in distributed inference to reduce the communication load, may be a useful area for future work.

References

1. Younis, O., Krunz, M., Ramasubramaian, S.: Node clustering in wireless sensor networks: Recent developments and deployment challanges. IEEE Network Magazine (2006)
2. Yu, J.Y., Chong, P.H.J.: A survey of clustering schemes for mobile ad hoc networks. IEEE Communications Surveys & Tutorials **7**(1) (2005)
3. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann (1988)
4. Crick, C., Pfeffer, A.: Loopy belief propagation as a basis for communication in sensor networks. In: Proceedings of the 19th Annual Conference on Uncertainty in Artificial Intelligence UAI'03. (2003)
5. Ihler, A.T., Fisher III, J.W., Moses, R.L., Willsky, A.S.: Nonparametric belief propagation for self-calibration in sensor networks. IEEE Journal of Selected Areas in Communication (2005)

6. Schiff, J., Antonelli, D., Dimakis, A.G., Chu, D., Wainwright, M.J.: Robust message-passing for statistical inference in sensor networks. In: Proceedings of the 6th International Conference on Information Processing in Sensor Networks IPSN'07. (2007)
7. Heinzelman, W.R., Chandrakasan, A.P., Balakrishnan, H.: An application-specific protocol architecture for wireless microsensor networks. IEEE Transactions on Wireless Communications **1**(4) (2002)
8. Younis, O., Fahmy, S.: HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. IEEE Transactions on Mobile Computing **3**(4) (2004)
9. Qin, M., Zimmermann, R.: VCA: an energy-efficient voting-based clustering algorithm for sensor networks. Journal of Universal Computer Science **13**(1) (2007)
10. Chatterjee, M., Das, S.K., Turgut, D.: WCA: a weighted clustering algorithm for mobile ad hoc networks. Cluster Computing **5**(2) (2002)
11. Li, C., Ye, M., Chen, G., Wu, J.: An energy-efficient unequal clustering mechanism for wireless sensor networks. In: Proceedings of the 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems MASS'05. (2005)
12. He, Y., Zhang, Y., Ji, Y., Shen, X.S.: A new energy efficient approach by separating data collection and data report in wireless sensor networks. In: Proceedings of the International Conference on Communications and Mobile Computing. (2006)
13. Kiri, Y., Sugano, M., Murata, M.: On characteristics of multi-hop communication in large-scale clustered sensor networks. IEICE Transactions on Communications **E90-B** (2007)
14. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. Science (2007)
15. Yedidia, J.S., Freeman, W.T., Weiss, Y.: Understanding belief propagation and its generalizations. Technical Report TR-2001-22, Mitsubishi Electric Research Laboratories (2002)
16. Jordan, M.I., Weiss, Y.: Probabilistic inference in graphical models. In: The Handbook of Brain Theory and Neural Networks. MIT Press (2002)
17. Wiberg, N.: Codes and Decoding on General Graphs. PhD thesis, Dept. of Electrical Engineering, Linköping, Sweden (1996)
18. Bickson, D., Dolev, D., Weiss, Y.: Modified belief propagation algorithm for energy saving in wireless and sensor networks. Technical report, The Hebrew University of Jerusalem (2005)
19. Banerjee, S., Misra, A.: Energy efficient reliable communication for multi-hop wireless networks. CM/Kluwer Journal of Wireless Networks (WINET) (2005)
20. Ault, A., Coyle, E., Zhong, X.: K-nearest-neighbor analysis of received signal strength distance estimation across environments. In: Proceedings of the 1st Workshop on Wireless Network Measurements WiNMee'05. (2005)
21. Woo, A., Tong, T., Culler, D.: Taming the underlying challenges of reliable multihop routing in sensor networks. In: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems SenSys'03. (2003)
22. TinyOS: http://www.tinyos.net/
23. Shnayder, V., Hempstead, M., Chen, B., Allen, G.W., Welsh, M.: Simulating the power consumption of large-scale sensor network applications. In: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems SenSys'04. (2004)
24. Younis, O., Fahmy, S.: An experimental study of routing and data aggregation in sensor networks. In: Proceedings of the 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems MASS'05. (2005)