Energy Balanced Topology for Sensor Networks

Kazi Muheymin Sakib
School of Computer Science and Information Technology
RMIT University
Melbourne, Victoria 3001
Email: sakib@cs.rmit.edu.au

Zahir Tari
School of Computer Science and Information Technology
RMIT University
Melbourne, Victoria 3001
Email: zahirt@cs.rmit.edu.au

Ibrahim Khalil
School of Computer Science and Information Technology
RMIT University
Melbourne, Victoria 3001
Email: ibrahimk@cs.rmit.edu.au

***Abstract*— Unattended sensor networks need to be self-configurable, scalable, energy balanced and intuitively having maximum possible lifetime. Clustering is one of the effective ways to achieve these objectives. In this research work, a novel virtual clustering technique is developed for the sensor networks. On top of that topology other networking issues e.g. routing, data aggregation can be implemented. The approach comprises a gateway selection algorithm with ensured network connectivity and coverage. The virtual clusters are formed around the gateways in a sensor field. The algorithm selects gateways among the deployed sensor nodes based on their residual energy. Each node in the network assesses its connectivity towards the gateway nodes, if the connection is vulnurable it starts a new gateway selection procedure. The technique is demand initiated, that is, if it is helpful towards the topology, a node becomes a gateway. Again, if the contribution becomes insignificant, a gateway can be turned to a normal node. Beside that, we also redefined the node characterization and node lifecycle, which also contributes significantly to the network lifetime. We apply our approach to a simulation environment and the simulation results justify our assertion.**

I. Introduction

During recent years, research in wireless sensor networks has become more and more active. At the same time, network protocols developed for sensor networks are of great importance to meet specific design goals. Design of sensor protocols for networks is affected by several factors. Sensor networks are often unattended so self-configurable algorithms need to be developed. A sensor network has limited memory, limited bandwidth, and limited computational capabilities [1]. So characterization of sensor nodes and better management of node life-cycle is needed. Despite of its energy constraint nature a sensor network is required to work for long time. Therefore, techniques needed to keep the network alive for as much time as possible. Finaly the techniques should be scalable.

We suggest that system designers can address these challenging issues in a demand initiated node organization. Sensor Nodes should contribute to the network backbone only it is needed to do so. Otherwise nodes can be inactive to save its precious energy for future usage. In a distributed dynamic system where the energy is a constraint, techniques should be developed where the recurrent transmission of dynamic state information is avoided. Moreover, to extend the network life an energy balancing mechanism should also be implemented.

In this paper we proposed an adaptive topology formation procedure for sensor networks, where the parameters of a particular area of interest will provoke sensor nodes to participate in the multihop network topology. The environment parameters can be obtained by assessing the connecting paths, residual energy and the query requests. The aim of this research is on developing a self-configurable energy balanced topology for the sensor networks which will live longer. This will be achieved by organizing sensor nodes in a virtual cluster, similar to ASCENT [2], and by balancing energy as well as tasks over the network, similiar to the distributed load balancing approach [3]. In particular, we will address following questions.

1) How do we efficiently manage the sensor nodes subject to balance the energy and tasks?
2) How do we organize the sensor nodes subject to self-configuring and lifetime maximizing?

Hierarchichal organization of sensor nodes is used in [4], however it may not always be scalable or self-configurable. Clustering, a special type of hierarchichal organization, is particularly useful for application that require scalability. Clustering is a well-known method in wireless communication. In a cluster, nodes send their data to the cluster head and the cluster head forwards the data to other cluster heads to get closer to the destination node [5]. Clustering enables bandwidth reuse, better resource utilization and power control. Thus, it maximizes the network capacity [6].

However, conventional clusters rely on a fixed infrastructure, more precisely, on a fixed area. Conventional clustering algorithms require all of the participating nodes to advertise cluster-dependent information repeatedly [7]. Some existing technique even require special types of nodes like energy-limitless sensors [8], as cluster heads. We argue that, neither the position of a sensor nor the area of interest for sensing can be fixed. Sensors are more promiscuous in nature; moreover new nodes can be added or discarded randomly. The topology should adapt to these important things. These issues cannot be handled with fixed architecture like conventional clustering. A dynamic clustering topology, which is adaptive and has self-organizing capabilities, is needed.

The main contribution of this paper is to develop a virtual clustering technique instead of its conventional counterpart which is described in Section III. Instead of fixed architecture, here we developed a dynamic, mutually overlapped clustering machanism. Members of a cluster are attached to the gateway instead of a cluster head, and the connection is not rigid. Any node can be attached to multiple gateways. Thus any node can be member of multiple clusters. We argue that the responsibilities of each sensor node might be changed over time. Based on current demand, nodes should be selected for specific task. To identify the tasks, we first define the node lifecycle. Based on the residual energy we classify the nodes into two types, SEN, nodes having sufficient energy to carry others information as well as sense own task and NEN, nodes having only necessary energy to sense its own task. This kind of classification helps us to make the network energy balanced. Finally at Section VI we show our simulation methods and results, which leads to draw the conclusion in Section V. Related works that either influenced or closely related to our work are described in the following section.

II. Related Work

In spite of being a relatively new field of study, researchers have begun discussing not only the uses and the challenges facing sensor networks [9], but also been developing preliminary ideas as to how these networks should be formed [7], [2], [10] as well as managing energy appropriately [11]. There have been some adaptive MAC layer protocols that influenced us a lot. Ye et al. developed a MAC layer protocol for sensor networks called S-MAC [12], Heinzelman et al. developed an application specific protocol architecture: LEACH [13], time division multiple-access (TDMA) MAC for low-energy operation is found in [14]. These works though share similar design principles with the technique proposed in this paper, but they focused on low-level synchronization, while we focused on topology formation.

Self-configuration is another branch of our study in sensor networks. Melodia et al. [15], Mondinelli et al. [16] and Sohrabi et al. [17] have made significant progress in self-configuration of sensor nodes. For configuration they mostly used the radio channel synchronization while we introduce a start state where every new node has to go through a thorough neighbor discovering process.

Mobile ad hoc network routing schemes, such as AODV [18], DSR [19], power-aware routing in mobile ad hoc networks, PAMAS [20], energy efficient routing schemes in wireless sensor networks, GAF [21] and Directed diffusion [22] adaptively configure the routing paths. In these protocols the optimal routes are chosen based on the energy at each node along the route or they find the best possible path based on their considered parameters. However, they do not adopt the basic network topology. Chang et al. developed a maximum lifetime routing protocol [23], where they find the shortest cost path using the link costs that reflect the communication energy consumption and the residual energy levels. This work shares the same notion of adaptation of the basic topology for efficient message delivery, but they used the neighbor information update mechanism. While our work creates the topology rather than routing paths considering the residual energy of its own. Upon which any kind of optimal routing algorithm can be implemented.

For organizing sensor networks Younis et. al. presented the hybrid clustering approach, HEED [5], Ma et. al. proposed a hub-spoke technique [8] for in-home sensor networks, Baek et al. developed a hierarchical approch [4]. We also influenced by Cheng et al [24]. They calculated the NP-Completeness of a minimum energy topology which is strongly connected. At the same time, in our work we argue that conventional clustering would not be an energy efficient solution for the sensor networks. Instead of conventional clustering Kwon et al.: presented a passive clustering technique [7]. In passive clustering the clusters are allowed to be overlapped. Thus the cluster heads have some common nodes, so that, instead of direct communication they can communicate with each other through those common nodes. By doing this, the need for special types of nodes for cluster heads is eliminated. However this flexibility also leads to substantilly higher overhead. Every time the topology changes there will be at least two types of selection or election procedures: one election procedure for cluster head and several others for identifying and selecting common nodes agreed by the neighboring cluster heads. In ASCENT [2], Cerpa et al. selects the active nodes to reinforce the topology. Whenever they select a node as an active node it stays awake all its' lifetime and performs multihop routing. It is already been mentioned that the area of interest for sensing may changes randomly, an active node may no longer be needed active. Even there may have more than the sufficient number of active nodes to carry the data. If the technique is not adaptive it will waste the precious energy, which in turn reduces the sensor network lifetime.

Energy scarcity make the sensor networks unique compared to the other distributed networks. To deal with this unique nature network decisions should be made based on residual energy. However, without characterising the sensor nodes based on their energy levles, it is impossible to take any energy based decision. Surprisingly, we did not find many works classifies sensor nodes.

In their paper [10], Vivek et al. proposed two types of node deployment. Type 0 nodes can only sense in its vicinity and has short-range communication ability. Type 1 nodes are the cluster heads that perform long-range transmissions, data aggregation and routing within the clusters etc. We argue that, if nodes are pre-characterized, perhaps from the factory, a precise node deployment is essential. The position of Type 1 nodes need to be exact, otherwise some of the type 0 nodes will be left outside the range of a type 1 node. This limits the network coverage. Yong et al. [8] identifies three types of sensor nodes. They denote sensors according to their resources as SRC - Small Resource Capacity, MRC - Medium Resource Capacity and LRC - Large Resource Capacity. They assumed that, LRC nodes are directly connected with the main power supply. So these nodes are actually energy-limitless sensors. This kind of nodes can only be found in an in-home sensor network. Again the node characterization is fixed

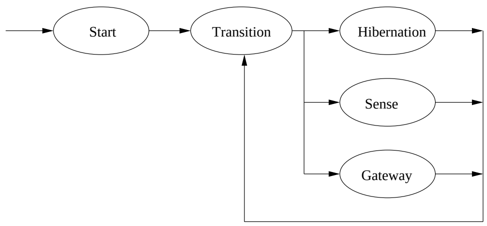

Fig. 1. Sensor Node Life-cycle.

and predetermined. However, in a practical sensor field, the residual energy of sensor nodes is decreasing continuously.

III. Proposed Topology Formation Technique Description

Our technique deploys nodes according to their abilities and responsibilities. The technique selects gateways, normal sensing nodes or hibernating nodes with the aim of creating a balanced, lifetime maximized network. Based on the current demand nodes can move from one state to another. To do so, we first define the life-cycle of a sensor node then characterise the nodes based on their residual energy, finally implement a gateway selection algorithm to form the virtual clusters.

A. Sensor node life-cycle

Sensor nodes are resource constrained, specially in terms of energy. To manage the sensor nodes efficiently, a well defined node life cycle is needed.

By observing the activities of sensor nodes, a specific pattern is found over their lifetime. Each node starts with initialization and then based on the current available metrics such as energy level, communication environments and query requests it enters into the execution phase.

According to their activities, we divide the sensor node lives into three categories -

- Initialization phase
- Decision making phase and
- Execution phase

The phases are shown in the Figure 1. The initialization phase only occurs once when a sensor node starts itself. In our diagram the Start state represents the initialization phase. During their life span, sensor nodes have to make various decisions, for example in which execution state it should be. This is the decision making phase represented by the 'Transition' state in the diagram. Apart from initialization and decision-making, each node is either in the non-active state or in the active state. In the non-active state hibernating nodes do nothing other than listening. Active states can be divided into two types - sense and gateway. In the sense state nodes sense its own territory based on the prescheduled request or query from the sink (i.e the destination node). On the other hand, in the gateway state, nodes perform data communication, data aggregation in addition to its own sensing.

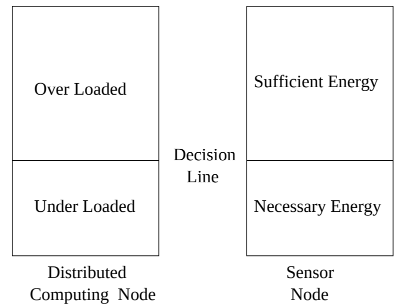

Fig. 2. Distributed and Sensor Node characterization.

B. Node classification

Without characterizing the nodes on their energy levels, it is impossible to create an energy balanced network. At any given time, the residual energy level of each node may not be the same. This is understandable because besides sensing, some of the nodes have additional responsibilities, like transmitting other node's information and data aggregation. To make the network live long, these additional responsibilities should be redistributed among other nodes periodically. Periodical task redistribution will ensure an energy-balanced environment in the network. Distribution of responsibilities must be fair to ensure that a node having minimum energy to sense its region and transmit the sensed data to its nearby gateway node, should not be considered for any additional burdens.

Here we assumed that all the nodes are architecturally equal, that is, all other resources - processing power, memory capacity are equal in all the sensor nodes. It is only the energy, which is variable. It is actually decreasing over time. This unique nature of sensor nodes leads us to characterize them based on their residual energy. In traditional distributed systems, processors are characterized by their process loads [3]. For example, processors having loads below a certain threshold are called under-loaded processors as shown in Figure 2. While others with a load above the predefined threshold are called over-loaded processor. Similar concept is adopted here. Sensor nodes will be divided into two groups at a given time -

- Sufficient Energy Node (SEN)
- Necessary Energy Node (NEN)

The current (residual) energy level will be normalized to the maximum battery capacity and scaled to 100 [25]. Based on the normalized residual energy; a minimum level of energy is drawn, which will be at least needed to sense the region and to transmit the sensed data to nearby gateway station. Nodes having residual energy above that level are called Sufficient Energy Node (SEN), these nodes have sufficient energy to take additional responsibilities. Nodes having residual energy less than or equal to that level are called Necessary Energy Node (NEN), these nodes only have energy to perform their own sensing tasks. Now the nodes are characterized into two types, we propose, a node only need to inform others when it changes from SEN to NEN. This technique reduces communication overhead substantially.

1) Determining node types (SEN/NEN): For simplicity we assumed that -

- Comparing with the data transmission or data reception, energy consumption in sensing is negligible [26].
- A node only transmits whenever a data packet is ready, and packet generation is directly proportional to sensing.
- We assume either nodes receive requests for sensing from sink nodes or there is a prescheduled query task for every node. Based on the query, nodes sense and generate data packets. The query of sensing may be random, non-uniform, so we can assume that, each sensing node has packet streams with Poission distribution.

Then, energy consumption or transmission cost over time t of a non-gateway node is -

$$E_{i(NG)}(t) = E_{tx} \times \lambda_i \times t \tag{1}$$

Where, E_{tx} - is the energy cost for transmitting a single packet. λ_i - is the packet generation rate at node *i*.

A gateway node receives data packets from its neighbors then transmits those to the next gateway. Since all the neighbors have independent packet streams with Poission distribution, the packet stream for a gateway node is still a Poission process. Let, gateway node *j* itself generates packets at a rate λ_j packets and has n_j neighbors. Then the packet arrival rate at gateway *j* is -

$$\lambda_j = \lambda_j + \sum_{k=1}^{n_j} \lambda_k \tag{2}$$

Where, λ_k is the packet generation rate at the *kth* neighbor.

The Expectation value of total packets at gateway *j* for duration *t* is -

$$\overline{X_j}(t) = \lambda_{j0} \times t + \sum_{k=1}^{n_j} \lambda_k \times t \tag{3}$$

For simplicity, let us assume that, energy to transmit (E_{tx}) and receive (E_{rx}) a single data packet is the same and is constant, i.e.,

$$E_{tx} = E_{rx} = C \tag{4}$$

Then, total energy consumption at Gateway *j* - total energy spent to receive all the packets from the neighbors and transmit $\overline{X_j}(t)$ packets -

$$E_{i(G)}(t) = E_{rx} \times \sum_{k=1}^{n_j} \lambda_k \times t + E_{tx}\overline{X_j}(t) \tag{5}$$

$$\Rightarrow E_{i(G)}(t) = C(\sum_{k=1}^{n_j} \lambda_k \times t + \overline{X_j}(t)) \tag{6}$$

Now if node *i*, spends t_1 time as a non-gateway node and t_2 time as a gateway node. Then total energy consumption of node *i* over time *t* (where, $t = t_1 + t_2$) can be found by equation 1 and equation 6, as -

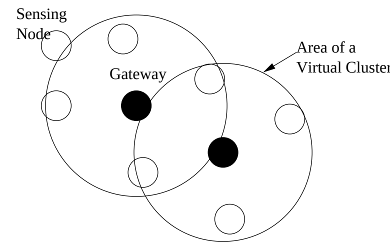

Fig. 3. Virtual Clustering.

$$E_i(t) = E_{i(NG)}(t_1) + E_{i(G)}(t_2) \tag{7}$$

If total amount of initial energy in a sensor node is *E* and the energy decision level is E_{th} [Energy Threshold] then,

$$\delta = \frac{E - E_i(t)}{E} = \begin{cases} \delta > E_{th} & Type = SEN \\ \delta < E_{th} & Type = NEN \\ \delta = 0 & Type = Exhausted \end{cases} \tag{8}$$

C. Virtual clustering

Here we propose a novel virtual clustering technique having energy efficient but lifetime maximizing topology. On top of that topology other networking issues, e.g. routing, data aggregation can be implemented. In virtual clustering, there is no fixed (geographical) area for any clusters. Cluster areas can overlap with each other's. Members of any virtual cluster can be associated with multiple gateways. Members of any virtual cluster can even initiate to form another virtual cluster as overlapping is allowed. That is virtual clustering is demand initiated, adaptive and self-configuring.

We build the virtual clusters around each of the gateway nodes. The gateway selection algorithm, described later, choose the appropriate gateways. To maintain the network connectivity each gateway is formed within the range of at least another gateway. It will also ensure that nodes within the cluster will have redundant connections to their gateways and have minimized packet loss probability as well. The region of a virtual cluster is the region of the gateway node, so that clustering is nothing but selecting the gateway nodes then associating the neighbors with the gateway. In the following sections gateway formation and member association procedure will be described in detail.

1) Virtual cluster formation: Whenever a gateway node is selected, it informs its neighboring nodes through gateway confirmation message. Each node receives the gateway confirmation message, updates the gateway table. Though the gateway node does not maintain any extra member list, the nodes received the gateway confirmation message become connected with that gateway. To be able to transmit sensed data, every node needs to be connected with at least one gateway. Thus the gateway node will form a loosely coupled virtual cluster with the surrounding neighboring nodes, which is shown in the Figure 3.

According to the definition of the gateway, there must have at least one gateway node in the neighboring nodes. We assumed that there are some nodes called initial gateways. Initial gateways never change their states. Sink nodes are considered as all time gateway nodes. Based on these initial gateway nodes, gradually the topology will be formed all over the sensor field. This assumption was necessary to form the topology.

D. Gateway selection Algorithm

After receiving a gateway request message, a node first checks the energy level and connection to other gateways. If the energy level of that node is above the energy threshold and the node is connected with at least another gateway node, the node is considered as an eligible candidate to become a gateway. Each eligible node then checks the number of gateway nodes in its communication range. If the number of gateway nodes equals the upper bound of a threshold (*Gth*), it refrains itself from becoming a gateway node.

If the node is a SEN, connected to at least one other gateway node and the number of connected gateways has not exceeded the gateway threshold *Gth*, it enters into the transition state to participate in the gateway formation procedure. The node then sends its willingness message to its neighbors and waits for a predefined amount of time (t_p) to hear from other nodes. If there are no other aspirant nodes from the neighbors, it becomes a gateway node, and informs the neighbors to update their respective gateway tables. If the node receives willingness messages from others, that is, there is more than one eligible node; the node will wait for a random amount of time (t_s). Within this period of time, if any confirmation of gateway message is received from any other nodes, it exits from the gateway procedure and reverts to its previous state. Otherwise, after waiting t_s time, it sends its confirmation of gateway message to its neighbors and enters into the gateway state. The gateway selection algorithm is given in the Algorithm 1.

1) Network connectivity and coverage: As gateways selections are demand initiated, there could have a chance of isolated clusters. However, our algorithm carefully considers that immense issue. We mathematically proved that the algorithm ensures the network connctivity and network coverage in the followings.

Assume that,

- n_i is the set of neighboring nodes of gateway node *i* (representing virtual cluster *i* as well).
- *N* is the super set of all the sensor nodes in the network.
- $\oint(n)$ is the gateway selection algorithm, which selects a gateway from the set *n*.

The algorithm starts from the initial gateway nodes (sink nodes), and it selects gateways based on descent directions. If the communication range of a sensor node is *r*. average distance of the next gateway node will be r/2. That is, from the initial gateway, the algorithm constructs a gateway sequence according to the following recursion rule.

If the distance from the initial point (sink node) of gateway *i* and gateway *i+1* is h_i and h_{i+1} respectively. Then -

Assume:

Each node has a gateway information table to store at most Gth number of gateway info.

Let :

NType: type of node — GNo: no of gateways
Gth: gateway threshold — State: state of the node
Msg: Control message — Table: gateway info
t_p: predefined wait time — t_s: random wait time

When gateway request message received;
if *NType* $\neq$ *NEN AND Gno* $\neq 0$ *AND Gno* $\leq$ *Gth* **then**
  $State_{prev} \leftarrow State_{current}$;
  $State_{current} \leftarrow State_{transition}$;
  `Broadcast(` $Msg_{willingness}$ `);`
  `Wait(`t_p`);`
  if $Msg_{received}$ `==` $Msg_{willingness}$ **then**
    `Wait(`t_s`);`
    if $Msg_{received}$ `==` $Msg_{confirmation}$ **then**
      `Update(Table);`
      $State_{current} \leftarrow State_{prev}$;
    end
    else
      `Broadcast(`$Msg_{confirmation}$`);`
      $State_{current} \leftarrow State_{gateway}$;
    end
  end
  else
    `Broadcast(`$Msg_{confirmation}$`);`
    $State_{current} \leftarrow State_{gateway}$;
  end
end

Algorithm 1: Gateway Selcetion Algorithm

$$|h_{i+1} - h_i| \leq r \tag{9}$$

$$\Rightarrow h_{i+1} \leq h_i + r \tag{10}$$

According to the definition of a gateway, there must be another gateway (GNo $\neq 0$) within the communication range of that node. So that -

$\oint(n_i) = \{x_i | x_i \in n_i \ AND \ x_i \in n_{i-1}\}$
$\oint(n_{i+1}) = \{x_{i+1} | x_{i+1} \in n_{i+1} \ AND \ x_{i+1} \in n_i\}$

Which implies that -

$$n_i \ \cap \ n_{i+1} \neq \phi \tag{11}$$

Now, if equation 10 and equation 11 holds and there are no discrete sensor nodes in the network, then -

$$n_0 \ \cup \ n_1 \ \cup \ n_2 \ \cup \ .... \ \cup \ n_i = N \tag{12}$$

Equation 11 ensures that a virtual cluster is connected at least one other cluster. That ensures the connectivity among the network. Equation 12 ensures that the algorithm covers the entire network.

IV. Experimental Results

In this section, we report results from our preliminary performance evaluation of our proposed technique. We will describe our simulation environments, our methodology and the comparison results with other techniques.

A. Simulation Environment

We used OMNeT++ as the simulation tool. We assumed that the sink node is a stand alone machine connected to the main power supply. Otherthan that, all the sensor nodes have the same memory, same energy and same processing power. We also assumed that the nodes have the same communication range. Any node within the communication range of others can communicate with them bidirectionally. We have created our own energy model as described in the section III. The sink node is positioned in the top right corner of the simulated area, we then deploy the sensor nodes. Initially, we created a two hop network by putting the nodes as far as possible, that is, at the opposite end of the communication range. After that, we gradually increase the node density.

Number of generated data packets are directly proportional to the sensing query or sensing schedule. The sensing query is also random and exponentially distributed. The generated packets are to be routed to the sink node through the gateway nodes. For each data communication there will be involved three types of energy consumption - sense the data, transmit data to the gateway, then receive and retransmit that packet among the gateways. We assume that the energy consumed while there is no query negligible.

B. Methodology

In the designing of the topology formation technique, we assumed that an efficient management of energy is required for a long lasting sensor network. If network related decisions are taken based on the remaining energy levels of a sensor node, it would certainly increase the lifetime of the nodes as well as the network. Our next and most vital assumption was classifying the sensor nodes (i.e., NEN or SEN) based on residual energy levels with the aim of forming an energy balanced network. Our first goal is to implement our algorithm in the simulation environment to compare with others. This serves the important purpose of validating our theoritical assumptions.

Our next target is to study the sensitivity of the proposed technique to the choice of parameters, for example, decision line of residual energy levels, number of gateway threshold, gateway formation waiting time etc. The metrics that have been chosen to analyze the performance are : Network Lifetime - which is measured by two ways. The time taken before the first node of the network dies. A similar metric has been used in [23]. The other one is the time taken to exhaust 50 % of the nodes, similar to the metric used in [2]. To show energy balancing we used here the standard deviation of residual energies. Standard deviation is calculated when the residual energy of at least one of the nodes reaches to zero.

Each experiment generates sense data with exponential distribution. The data is carried out to the sink using shortest path algorithm. The transmission layer implements the simple CSMA MAC, which is provided by the OMNeT++. We used fixed parametric value for the Gateway threshold (≤ 3), Decision line of Energy levels (=20 percent), Gateway formation waiting times (=0.002 seconds) for all the cases. We will justify these values later. For simplicity, the sensor nodes are considered immobile. We will experiment our work with the mobility framework later.

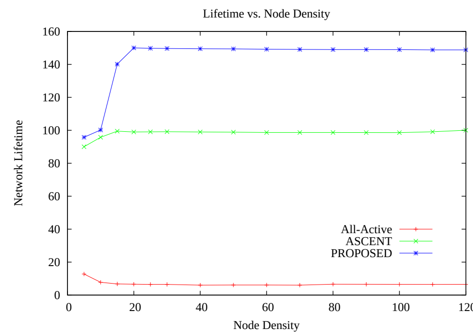

Fig. 4. Density vs. Lifetime : time taken to die the first node.

C. Comparative Study

Our first experiment compare the average lifetime of the sensor network where our proposed technique is applied, with the all-active case [the all-active case denotes the simple deployment of sensor nodes, where no topology formation algorithm was applied.], and ASCENT [2].

Figure 4 shows the time taken for the first node to die as a function of the node density. At a lower density our proposed technique performs similar to the ASCENT. This is because, when the density is too low, there will be excess gateway nodes to carry the data. The number of gateways is restricted by the parameter, gateway threshold. However, when we increased the node density, there were less gateway nodes comparing to the non-gateway nodes, so that, the task distribution was more even and there were more nodes to carry the duty of a out going gateway. In the proposed technique if any node crosses the Decision line of the energy levels, it is considered as NEN and they are forced to leave the gateway state. NEN nodes are alive and only can perform their own sensing. This kind of essenstial classifications were absent in other methods, which badly effects their network lifetime. The result clearly shows that the proposed technique outperforms the other two under moderate to highly densed condition.

Figure 5 shows the time taken to die at least 50 percent of the nodes while we varied the node density in our test bed. Though the gaps between the other techniques are reduced, however, we argue that, the network might be partitioned well before the time it takes to exhaust the 50% nodes. Still the figure shows the suprimacy of our proposed technique.

In figure 6 we plot the standard deviation of remaining energy levels while the energy level of the first node reaches to zero. This metric shows us how balanced the network is. For an ideal condition, where the tasks are exactly distributed, the

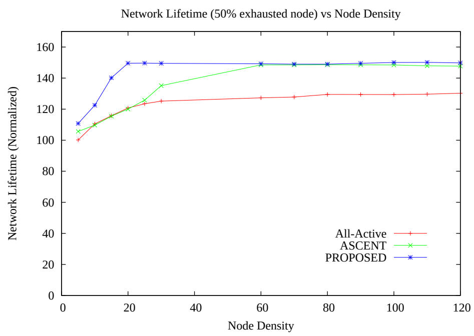

Fig. 5. Density vs. Lifetime : time taken to die 50 percent of the nodes.

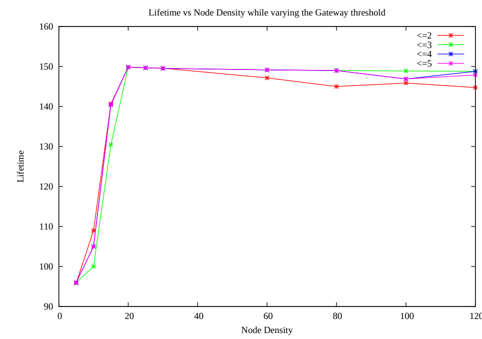

Fig. 8. Density vs. Network Lifetime : while the Gateway threshold varied.

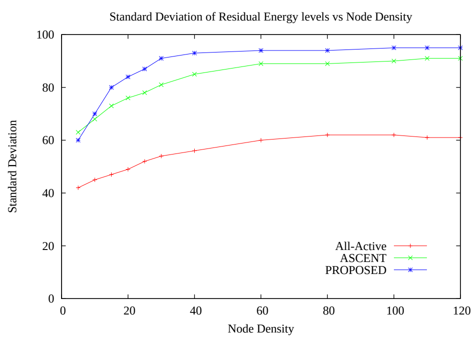

Fig. 6. Density vs. Standard Deviation of remaining energies : when first node dies.

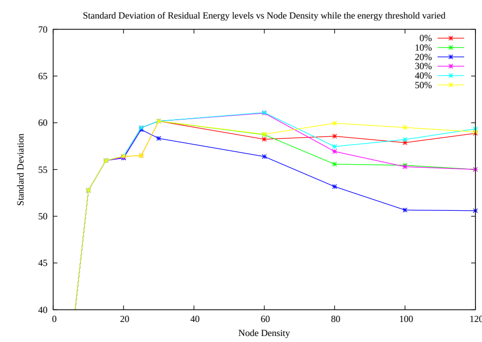

Fig. 9. Density vs. Standard Deviation of remaining energies : while the Energy threshold varied.

stadard deviation of remaining energy levels should be zero. Here the figure shows that the deviation is about 50 percent in our technique, while the others are around 90 percent.

To explain how our proposed technique reacts to different configuration parameters, we conducted sensitivity experiments on two parameters gateway threshold and energy threshold. Figure 7 and 8 show the standard deviation of remaining energies and network lifetime respectively, when we varied the gateway threshold from less than or equal to 2 to less than or equal to 5. Figure 7 shows that the lower the gateway threshold the larger the deviation, but after a certain threshold value the deviation almost stabilizes. That is,

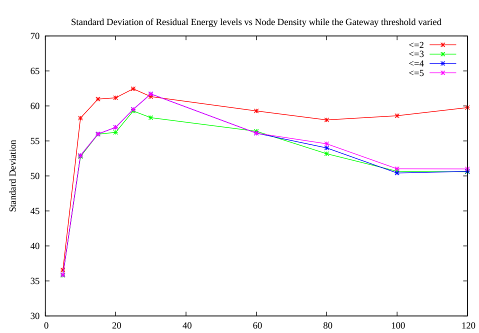

Fig. 7. Density vs. Standard Deviation of remaining energies : while the Gateway threshold varied .

for every network there is a saturation point for the gateway threshold. After that saturation point, increasing of gateway number doesn't create any effect to the standard deviation of the remaining energies. Similar scenario has also been noticed in Figure 8. The threshold value is proportional to lifetime upto a cetain point. After that it again stabilizes. Most importantly the saturation point of this two important network properties are almost same. In Figure 7, the best result is found when the threshold is less than or equal to 3, and in Figure 8, at threshold ≤ 3, we got closest to the best result. Gateway threshold is directly related to the maximum connectivity of a single node in the network. In our simulation, a single sensor node can have maximum five neighbors.

Energy threshold is used to decide whether the sensor node will be NEN or SEN, this type of characterization helps us to create a balaced network in terms of energy as well as tasks. Figure 9 shows the standard deviation of residual energies as a function of node density while we varied the energy threshold from 0% to 50% . We concluded that, at energy threshold 20 percent we got the least deviation for our simulated network.

V. Conclusion and Future Work

Each of the gateway nodes forms an overlapping clusters as shown in Figure 3. We call this type of clusters as virtual clusters as the members and area of any particular clusters are not mutually exclusive. The topology is also stateless as it is changing continuously and does not depend on the previous states. It only depends on current network demand. Each of these virtual clusters also contains at least another gateway. That means, within the one-hop distance of a gateway there is another gateway. So, most of the intermediate nodes may be connected with more than one gateway. This will ensure possibly redundant transmission paths.

Virtual clusters are dynamic and adaptive. Whenever a node faces a high packet loss, or a node moves out of communication range of its gateway, it can initiate the gateway formation procedure. As a result, a new virtual cluster can be formed. As the gateway procedure is demand-initiated, usually more gateways can be found in the high sensing active regions than the less active regions. This is an inherent procedure of reinforcement in the gateway formation procedure. This is a self-organizing topology. Each node in the network must go through the start state, where it will initialize itself by discovering its neighbors. After starting whenever a node enters the transition state it updates its neighboring information.

By characterizing sensor nodes as two types NEN or SEN (Figure 2), we able to reduce the overhead transmission convincingly. Now nodes only need to inform this state once in a lifetime when it reduces from SEN to NEN, if it is needed. Clear understanding of node lifecycle (Figure 1) helps to distribute the tasks evenly. Which is important to make the network balance.

This research does not consider the network partition. Network partition can happen due to dieing of sensor nodes. Sensor nodes can be died early because of over activity or because of some external events like stomping, natural calamities. Detecting network partitions and handling network partition can be our future work.

ACKNOWLEDGMENT

This work is supported by Australian Research Council (ARC), linkage scheme, no. LP0347217.

REFERENCES

[1] N. Heo and P. K. Varshney, "Energy-effi cient deployment of intelligent mobile sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 35, no. 1, pp. 78–92, Jan. 2005.

[2] A. Cerpa and D. Estrin, "Ascent: Adaptive self-confi guring sensor networks topologies," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pp. 272–285, July 2004.

[3] P. K. Sinha, *Distributed Operating Systems, Concepts and Design*. IEEE Press, 1997.

[4] S. J. Baek, G. D. Veciana, and X. Su, "Minimizing energy consumption in large-scale sensor networks through distributed data compression and hierarchical aggregation," *IEEE Journal on Selected Areas in Communication*, vol. 22, no. 6, pp. 1130–1140, Aug. 2004.

[5] O. Younis and S. Fahmy, "Heed: a hybrid, energy-effi cient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, Oct. 2004.

[6] W. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *5th ACM/IEEE Mobicom Conference (MobiCom '99)*, Seattle, WA, USA, Aug. 1999.

[7] K. J. Taek, M. Gerla, V. Varma, M. Barton, and T. Hsing, "Effi cient fboding with passive clustering-an overhead-free selective forward mechanism for ad hoc/sensor networks," *IEEE*, vol. 91, no. 8, pp. 1210–1220, Aug. 2003.

[8] Y. Ma and J. H. Aylor, "System lifetime optimization for heterogeneous sensor networks with a hub-spoke topology," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pp. 286–294, July 2004.

[9] D. Ganesan, A. Cerpa, W. Ye, Y. Yu, J. Zhao, and D. Estrin, "Networking issues in wireless sensor network," *Journal of Parallel and Distributed Computing (JPDC), Special issue on Frontiers in Distributed Sensor Networks*, 2004.

[10] V. Mhatre, C. Rosenberg, D. Kofman, R. Mazumdar, and N. Shroff, "A minimum cost heterogeneous sensor network with a lifetime constraint," *IEEE Transactions on Mobile Computing*, vol. 4, no. 1, pp. 4–15, Jan. 2005.

[11] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, "Optimizing sensor networks in the energy-latency-density design space," *IEEE transactions on Mobile Computing*, vol. 1, no. 1, pp. 70–80, Jan. 2002.

[12] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *IEEE Transactions on Networking*, vol. 12, no. 3, pp. 493–506, June 2004.

[13] W. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application specifi c protocol architecture for wireless microsensor networks," *IEEE transactions on wireless communications*, vol. 1, no. 10, pp. 660–670, Oct. 2002.

[14] L. Clare, G. Pottie, and J. Agre, "Self-organizing distributed nensor networks," in *SPIE conference: Unattended Ground Sensor Technologies and Applications*, Orlando, Fl, USA, Apr. 1999, pp. 229–237.

[15] T. Melodia, D. Pompili, and I. Akyldiz, "On the interdependence of distributed topology control and geographical routing in ad hoc and sensor networks," *IEEE Journal on Selceted Areas in Communications*, vol. 23, no. 3, pp. 520–532, Mar. 2005.

[16] F. Mondinelli and Z. Kovacs-Vajna, "Self-localizing sensor network architectures," *IEEE Transactions on Instrumentation and Measurement*, vol. 53, no. 2, pp. 277–283, Apr. 2004.

[17] K. Sohrabi, W. Merrill, J. Elson, L. Girod, F. Newberg, and W. Kaiser, "Methods for scalable self-assembly of ad hoc wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 317–331, Oct. 2004.

[18] C. E. Perkins, E. M. Belding-Royer, and S. R. Das. (2003, Feb.) Ad hoc on-demand distance vector (aodv) routing. draft-ietf-manet-aodv-13.txt. [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-13.txt

[19] J. Broch, D. Johnson, and D. Maltz. (1999, Oct.) The dynamic source routing protocol for mobile ad hoc networks. draft-ietf-manet-dsr-03.txt. [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-03.txt

[20] S. Singh, M. Woo, and C. S. Raghabendra, "Power-aware routing in mobile ad hoc networks," in *ACM/IEEE Mobicom Conference (MobiCom '1998)*, Dallas, Texas, USA, 1998, pp. 181–190.

[21] Q. Gao, K. J. Blow, D. J. Holding, I. W. Marshall, and X. Peng, "Routing analysis and energy effi ciency in wireless sensor networks," in *IEEE 6th CAS Symp. on emerging technologies: Mobile and Wireless Communication*, Shanghai, Chaina, May 2004, pp. 533–536.

[22] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," in *4th ACM/IEEE Mobicom Conference (MobiCom '2000)*, Boston, MA, USA, Aug. 2000, pp. 56–67.

[23] J. Chang and L. Tassiulas, "Maximum lifetime routing in wireless sensor networks," *IEEE/ACM Transsactions on Networking*, vol. 12, no. 4, pp. 609–619, Aug. 2004.

[24] X. Cheng, B. Narahari, R. Simha, M. Cheng, and D. Liu, "Strong minimum energy topology in wireless sensor networks: Np-completeness and heuristics," *IEEE Transactions on Mobile Computing*, vol. 2, no. 3, pp. 248–256, July 2003.

[25] X. Hong, M. Gerla, and H. Wang, "Load-balanced, energy-aware communications for mars sensor-networks," in *IEEE Aerospace Conference*, Bigsky, MI, USA, Mar. 2002, pp. 3–1109 – 3–1115.

[26] R. Min, M. Bhardwai, N. Ickes, A. Wang, and A. Chandrakasan, "The hardware and the network: Total-system strategies for power aware wireless microsensors," in *IEEE CAS Workshop on Wireless Communications and Networking Power Efficient Wireless Ad Hoc Networks*, Pasadena, California, USA, Sept. 2002.