Asian Journal of Information Technology 7 (9): 408-415, 2008
ISSN: 1682-3915

Energy Balanced Self-Configuring Sensor Networks

Kazi M. Sakib, Zahir Tari and Ibrahim Khalil
School of Computer Science and Information Technology,
RMIT University Melbourne, Victoria 3001, Australia

Abstract: Like other distributed systems, unattended sensor networks need to be balanced, especially energy balanced. Lifetime of a sensor network can be maximized if a balanced network is formed. In this research, we characterize nodes based on their residual energy levels and distribute tasks according to a node's capability to balance energy over the network. We characterize a node as SEN, when that node is having sufficient energy to perform additional responsibilities other than its own sensing task. Otherwise, we characterize a node as NEN, when that node is having only a small amount of energy to perform its own tasks. Every node starts as SEN and at some point it becomes NEN based on a predefined threshold value. Thus, the nodes are protected from the early exhaustions. A virtual self-configuring clustering technique is also developed to rotate the tasks among nodes. We apply our approach to a simulation environment and the results justify our assertion.

Key words: Sensor network, SEN, NEN, ASCENT, self-configuring

INTRODUCTION

The energy consumption is involved in each of the sensing activities. In such a system, distributing processing and communication activities among the deployed nodes is highly required, so that no single sensor node will be exhausted unexpectedly. Sensor networks are often unattended and a sensor has limited memory, limited bandwidth and limited computational capabilities (Heo and Varshney, 2005). Despite of its energy constraint nature a sensor network is expected to live long.

We suggest that system designers can address this challenging issue in a demand initiated node organization. Sensor nodes should contribute to the network backbone only if 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 energy balancing mechanisms should also be incorporated.

In this study, Mhatre *et al*. (2005) proposed two types of node deployment. Type 0 nodes are simple sensing nodes. Type 1 nodes are the cluster heads that perform long-range data transmissions, data aggregation and routing within the clusters. We argue that if nodes are pre-determined, perhaps from the factory, a precise node deployment is essential. The position of nodes need to be precise, otherwise some of the nodes will be disconnected, which in turn, will limit the network coverage. Ma and Aylor (2004) identifies 3 types of sensor nodes based on the resources such as SRC (Small Resource Capacity), MRC (Medium Resource Capacity) and LRC (Large Resource Capacity). LRC nodes are assumed directly connected with the main power supply, which implies, this kind of nodes can only be existed in an in-home sensor network. However, in a practical sensor field, the residual energy of a sensor node is decreasing.

Hierarchical organization of sensor nodes is used in Baek *et al*. (2004) and Cheng *et al*. (2003), however, hierarchy may not always be scalable, instead clustering, a special type of hierarchical organization, is particularly useful for applications that require scalability (Edgar and Callaway, 2003). 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 (Younis and Fahmi, 2004). Clustering enables bandwidth reuse, better resource utilization and power control (Heinzelman *et al*., 1999). 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 (Taek *et al*., 2003). Some existing techniques even require special types of nodes like energy-limitless sensors (Ma and Aylor, 2004), as cluster heads. Instead of conventional clustering presented a passive clustering technique (Taek *et al*., 2003). In

Corresponding Author: Kazi Sakib, School of Computer Science and Information Technolog, RMIT University Melbourne, Victoria 3001, Australia

passive clustering the clusters can be overlapped. However, every time the topology changes there will be at least two types of election procedures for electing the cluster head and identifying common overlapped nodes. In ASCENT (Cerpa and Estrin, 2004), selects the active nodes to reinforce the topology. Whenever, they select a node as active, the node stays awake all of its lifetime and performs multi-hop routing. We argue that the area of interest for sensing may change randomly, as a result, an active node may no longer be needed as active.

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 (Cerpa and Estrin, 2004) and by balancing energy over the network, similar to the distributed load balancing approach (Sinha, 1997). Energy scarcity makes a sensor network unique compared to the other distributed networks. To deal with this unique nature, networking decisions should be made based on residual energies. However, without characterizing the sensor nodes based on their energy levels, it is impossible to take any energy-based decision. In this research, we classify nodes into 2 types, SEN, nodes having sufficient energy to carry others information, besides it own sensing tasks and NEN, nodes having only necessary energy to sense its own tasks. By doing this, we can protect a node from early exhaustion, which in turn increases the network lifetime.

The main contribution of this study, apart from the node characterization, is to develop a virtual clustering technique. Instead of fixed architecture, here we developed a dynamic, mutually overlapped clustering mechanism where tasks are rotated among the neighbors. To identify the tasks, we first define the node lifecycle.

SENSOR NODE LIFECYCLE

Sensor nodes are resource constrained, especially in terms of energy. To manage the sensor nodes efficiently, a well-defined node lifecycle is needed, which will help us to understand and identify the tasks involved. By observing the activities of a sensor node, it is found that each node starts its life 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 categorize a sensor life as follows (Fig. 1):

- Initialization phase.
- Decision making phase.
- Execution phase.

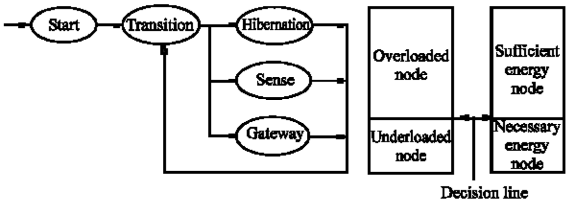

Fig. 1: Sensor node life-cycle with distributed and sensor node characterization

The initialization phase only occurs once when a sensor node starts. In the Fig. 1, 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 Fig. 1.

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.

NODE CLASSIFICATION

Without characterizing the nodes, based on residual 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, for example transmitting other node's information or data aggregation. To extend the network life, these additional responsibilities should be redistributed periodically among other nodes. This 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 nodes, should not be considered for any additional tasks.

Here all the nodes are assumed architecturally equal that is all other resources-processing power, memory capacity are equal in all the sensor nodes. It is only the energy, which varies, more precisely; it is decreasing over time. This unique nature of sensor nodes leads to characterize nodes based on their residual energy. In traditional distributed systems, processors are characterized by their process loads (Sinha, 1997). For example, processors having loads below a certain threshold are called under-loaded processors as shown in Fig. 1. While others with a load above the predefined threshold are called overloaded 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 residual energy level is assumed normalized to the maximum battery capacity and scaled to 100 (Hong *et al*., 2002). Based on the normalized residual energy; a least level of energy is drawn, which will be needed to sense the region and to transmit the sensed data to nearby gateway stations. 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.

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

- Comparing with the data transmission or data reception, energy consumption in sensing is negligible (Min *et al*., 2002).
- A node only transmits whenever a data packet is ready and packet generation is directly proportional to sensing.
- We assume either node 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 poisson 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} = The energy cost for transmitting a single packet.

λ_i = The packet generation rate at node i.

A gateway receives data packets from its neighbors and transmits to the next gateway. Since, all the neighbors have independent packet streams, the packet stream for a gateway node is still a Poison process. Let, gateway j generates packets at a rate λ_j 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:

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

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., $E_{tx} = Er_x = C$.

Total energy consumption at gateway j, which is total energy spent to receive packets from the neighbors and transmit X_j (t) packets becomes as follows:

$$E_i\,(G)(t) = E_{rx}\times\sum_{k=1}^{n_j}\lambda_k\times t + E_{tx}\bar{X}_j\,(t) \tag{4}$$

$$E_i\,(G)(t) = C(\sum_{k=1}^{n_j}\lambda_k\times t + \bar{X}_j\,(t)) \tag{5}$$

Now, if node i, spends t_1 time as a non-gateway node and t_2 time as a gateway, total energy consumption of node i over time t (where: $t = t_1 + t_2$) can be found by Eq. 1 and 5 as:

$$E_i(t) = E_{i(NG)}(t1) + E_{i(G)}(t2) \tag{6}$$

If total amount of initial energy in a sensor node, is E and the energy decision level is Eth (Energy Threshold) then:

$$\begin{array}{ll} & \delta > Eth\ type = SEN \\ \delta = (E - E_{i(t)})/E = & \delta < Eth\ type = NEN \\ & \delta = 0\ type = Exhausted \end{array} \tag{7}$$

PROPOSED TOPOLOGY FORMATION TECHNIQUE

The proposed technique deploys nodes according to their capabilities. The technique selects gateways, normal sensing nodes or hibernating nodes with the aim of creating a balanced, lifetime maximized network. However, based on the current demand, nodes can move from one state to another.

Virtual clustering: A novel virtual clustering technique is proposed where other networking issues, e.g. routing, data aggregation can be implemented. In virtual clustering, there is no fixed 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 since overlapping is allowed. Virtual clustering is demand initiated, adaptive and self-configuring. We build the virtual clusters around the gateway nodes. The gateway selection algorithm chooses 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.

Whenever a gateway node is selected, it informs its neighbors through gateway confirmation message. Each node receives the gateway confirmation message, updates the gateway table. Though the gateway node does not maintain any 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 will form a loosely coupled virtual cluster with the surrounding neighbors, which is shown in the (Fig. 2). According to the definition of a gateway, there must have at least one gateway node in the neighbors. 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, the topology will be formed all over the sensor field.

Gateway selection algorithm: After receiving a gateway request message, a node first checks its 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, it is considered an eligible candidate to become a gateway. Each eligible node checks the number of gateways in its communication range. If the number of gateway nodes equals the upper bound of a threshold (Gth), it refrains from becoming a gateway node.

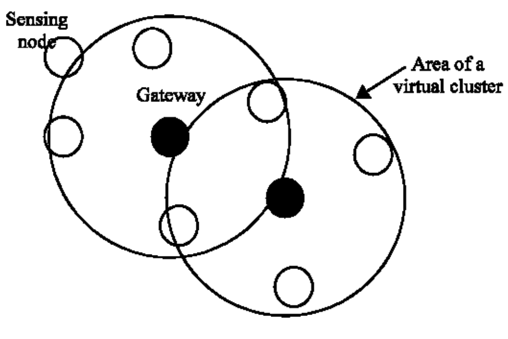

Fig. 2: Virtual clustering

If the node is a SEN, connected to at least another 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 sends its willingness 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 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 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 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.

Network connectivity and coverage: As gateway selections are demand initiated, there could be isolated clusters. However, our algorithm carefully considers that important issue. Here we mathematically proved that, the algorithm ensures the network connectivity and network coverage.

Algorithm 1: Gateway Selection Algorithm

```
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
```

Assume: Each node has a gateway information table.
Assume that:

- n_i is the set of neighboring nodes of gateway i.
- N is the set of all the sensor nodes in the network.
- f (n) is the gateway selection algorithm.

The algorithm starts from the initial gateway 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. From the initial gateway, the algorithm constructs a gateway sequence according to the following recursion rule.

If, from the initial point (i.e. sink), distances of a gateway I and its neighboring gateway i + 1, are h_i and $h_i + 1$, respectively, then:

$$|h_i +1 - h_i| \le r \tag{8}$$

$$=> h_i + 1 \le h_i + r \tag{9}$$

According to the definition of a gateway, there must be another gateway (GNo _= 0) within the communication range of that node. So that

$$f\,(n_i) = \{x_i|x_i \in n_i \text{ and } x_i \in n_i - 1\} \tag{10}$$

$$f\,(n_i + 1) = \{x_i + 1|x_i + 1 \in n_i + 1 \text{ and } x_i + 1 \in n_i\} \tag{11}$$

Which implies that

$$n_i \cap n_i + 1 \mathbin{\_=} \varphi \tag{12}$$

Now, if Eq. 9 and 12 holds and if there are no nodes which form partition in the network, then:

$$n_0 \cup\ |n_1 \cup |n_2 \cup |.........\cup |n_i = N \tag{13}$$

Equation 12 ensures that a virtual cluster is connected to at least one other cluster. That ensures the connectivity among the network. Equation 13 ensures that the algorithm covers the entire network.

SIMULATION

We used OMNeT++ as the simulation tool. We assumed that the sink node is a stand-alone machine connected to the main power supply. Other than that all the sensor nodes have the same initial memory, same energy and same processing power. We also assumed that the nodes have the same communication range. Any node within the communication range can communicate with others bi-directionally. We have created our own energy model as described in this study. The sink node is positioned at 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 is directly proportional to the sensing query. The sensing query is also random and exponentially distributed. For each data communication there will be involved three types of energy consumption-sensing, 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.

Our goal is to implement our algorithm in the simulation environment to compare with others. This serves the important purpose of validating our theoretical assumptions. The metrics that have been chosen to analyze the performances are:

Network lifetime, which is measured by two ways, one is the time taken before the first node of the network dies (Chang and Tassiulas, 2004). Another one is the time taken to exhaust 50% of the nodes, similar to the metric used in Cerpa and Estrin (2004). To show energy balancing, the standard deviation of residual energies is used. Standard deviation is calculated when the residual energy of at least one of the nodes reaches to zero.

To transmit data, we used shortest path algorithm. The transmission layer implements the simple CSMA MAC. We used fixed parametric values for the Gateway threshold ≤3, decision line of energy levels = 20% Gateway formation waiting time is twice the roundtrip. We will justify these values later.

Comparative study: We compare the energy balancing and average lifetime of a 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 (Cerpa and Estrin, 2004). In Fig. 3, we plot the standard deviation of remaining energy levels when 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 standard deviation of remaining energy levels should be zero. The Fig. 3 shows that the deviation is about 50% in our technique, while others are around 90%.

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-

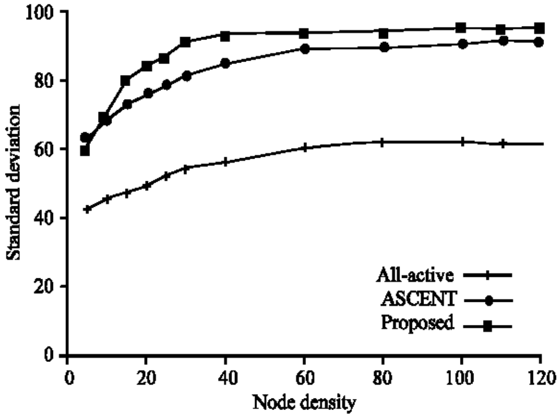

Fig. 3: Density vs. standard deviation of remaining energies: when first node dies

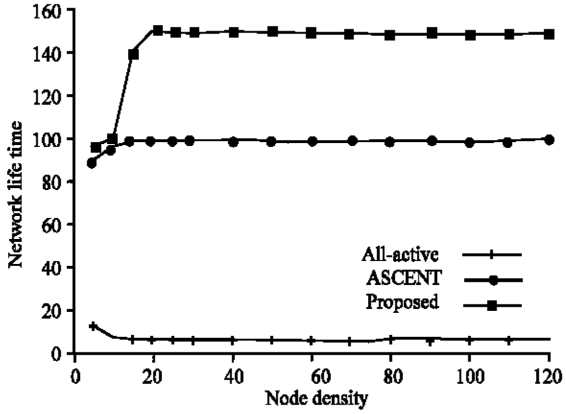

Fig. 4: Density vs. life time: time taken to die the first node

gateway nodes, the task distribution was more even and there were more nodes to carry the duty of an out going gateway. In the proposed technique, if any node crosses the decision line of the energy level, 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. These types of classifications were absent in other methods, which strongly effects their network lifetime. The result clearly shows that the proposed technique outperforms the others under moderate to high node densities. Another significance of this result is the scalability. It rapidly reaches around 150 time unit with 15 nodes and enters into a saturation region.

Figure 5 shows the time taken to die at least 50% 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. The Fig. 5 shows that the proposed technique is still better than others.

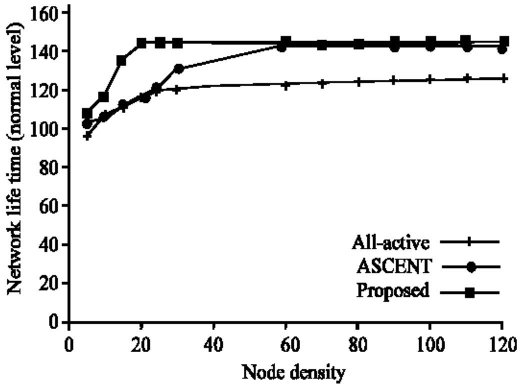

Fig. 5: Density vs. life time: time taken to die 50% of the nodes

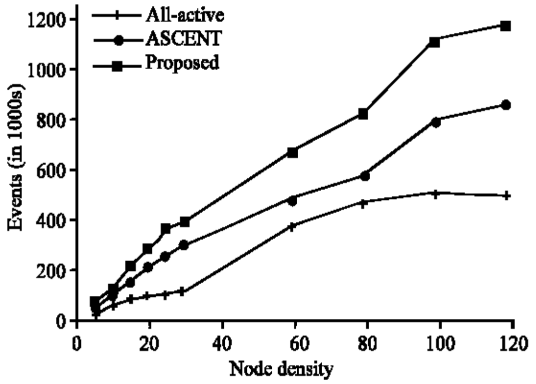

Fig. 6: Density vs. total number of generated events (in 1000s): upto exhaustion of first node

In Fig. 6, we show the number of events, i.e. the number of generated events upto the time when first node dies. Our proposed technique detects almost same number of events in a lesser dense condition comparing to other two techniques. However, when we increase the density it detects almost double than the ASCENT and triple than the All-active techniques detects the events. The result commensurate with our previous results (Fig. 4 and 5) that our technique increases the life time of generated events up to the time when first node dies. Our proposed technique detects almost same number of events in a lesser dense condition comparing to other two techniques. However, when we increase the density it detects almost double than the ASCENT and triple than the all active techniques detects the events. The result commensurate with our previous results (Fig. 4 and 5) that our technique increases the life time.

CONCLUSION

Sensor nodes are characterized as NEN or SEN (Fig. 1). By doing this, we able to reduce the overhead transmission by a large margin as nodes now need to inform their state once in a lifetime only when it reduces from SEN to NEN. Energy consumption is directly related to the task that is performed by a node. Clear understanding of node lifecycle (Fig. 1) helps us to distribute the tasks evenly. Which is important to make the network balanced.

Each of the gateway nodes forms an overlapping cluster as shown in Fig. 2. We call these clusters as virtual clusters. Because, the members and area of any particular cluster are not to the sink. Virtual clusters are dynamic and adaptive. Whenever a node faces a high packet loss, or a node moves out of communication range from its gateway, it can initiate the gateway formation procedure. As a result, a new virtual cluster is formed. The topology, that formed is self-organizing and scalable. This research does not consider the network partition. Network partition can happen due to dying of sensor nodes. Sensor nodes can die early because of over activity or external events like stomping, natural calamities. Detecting network partitions and handling that partition will be our future work. We also do not consider the redundant node coverage. However, we do believe scheduling for node coverage may enhance the network life.

ACKNOWLEDGMENT

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

REFERENCES

Baek, S.J., G.D. Veciana and X. Su, 2004. Minimizing energy consumption in large-scale sensor networks through distributed data compression and hierarchical aggregation. IEEE J. Selected Areas Commun. (J-SAC), 22 (6): 1130-1140. DOI: 10.1109/JSAC.2004.830934. http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/49/29265/01321225.pdf?arnumber=1321225.

Cerpa, A. and D. Estrin, 2004. Ascent: Adaptive self-configuring sensor networks topologies. IEEE Trans. Mob. Comput. (TMC), 3 (3): 272-285. DOI: 10.1109/TMC.2004.16. http://iee explore.ieee.org/Xplore/login.jsp?url=/iel5/7755/29217/01318596.pdf?tp=&isnumber=&arnumber=1318596.

Chang, J. and L. Tassiulas, 2004. Maximum lifetime routing in wireless sensor networks. IEEE/ACM Trans. Networking (TON), 12 (4): 609-619. DOI: 10.1109/TNET.2004.833122. http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/90/29402/01331424.pdf?arnumber=1331424.

Cheng, X., B. Narahari, R. Simha, M. Cheng and D. Liu, 2003. Strong minimum energy topology in wireless sensor networks: Np-completeness and heuristics. IEEE Trans. Mob. Comput., 2 (3): 248-256. DOI: 10.1109/TMC.2003.1233530. http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/7755/27636/01233530.pdf?arnumber=1233530.

Edgar, H. and J. Callaway, 2003. Wireless sensor networks: Architectures and protocols. Auerbach Publications, Boca Raton, FL, USA, pp: 13-93. ISBN: 0-8493-1823-8.

Heinzelman, W., J. Kulik and H. Balakrishnan, 1999. Adaptive protocols for information dissemination in wireless sensor networks. in 5th ACM/IEEE Mobicom Conference (MobiCom), Seattle, WA, USA, pp: 174-185. DOI: 10.1145/313451.313529. http://portal.acm.org/citation.cfm?id=313529.

Heo, N. and P.K. Varshney, 2005. Energy-efficient deployment of intelligent mobile sensor networks. IEEE Trans. Syst., Man Cybernet. (SMC) part A: Sys. Humans, 35 (1): 78-92. DOI: 10. 1109/TSMCA.2004.838486. http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/3468/29969/01369347.pdf?arnumber=1369347.

Hong, X., M. Gerla and H. Wang, 2002. Load-balanced, energy-aware communications for mars sensor-networks. IEEE Aerospace Conference, Bigsky, MI, USA, pp: 1103-1109-1113-1115. DOI: 10.1109/AERO.2002.1035240. PMID: 7529094. http://ie eexplore.ieee.org/xpls/abs_all.jsp?arnumber=1035240.

Ma, Y. and J.H. Aylor, 2004. System lifetime optimization for heterogeneous sensor networks with a hub-spoke topology. IEEE Trans. Mob. Comput., 3 (3): 286-294. DOI: 10.1109/TMC.2004.27. INSPEC: 8242147. http://ieeexplore.ieee.org/xpl/freeabs_all. jsp?arnumber=1318597.

Mhatre, V., C. Rosenberg, D. Kofman, R. Mazumdar and N. Shroff, 2005. A minimum cost heterogeneous sensor network with a lifetime constraint. IEEE Trans. Mob. Comput., 4 (1): 4-15. DOI: 10.1109/TMC.2005.2(410)4. http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/7755/29961/01369175.pdf?tp=&isnumber=&arnumber=1369175.

Min, R., M. Bhardwai, N. Ickes, A. Wang and A. Chandrakasan, 2002. The hardware and the network: Total-system strategies for power aware wireless micro-sensors. In: IEEE CAS Workshop on Wireless Communications and Networking Power Efficient Wireless Ad Hoc Networks, Pasadena, California, USA. http://www-mtl.mit.edu/research groups/icsystems/pubs/conferences/2002/rmin_cas 2002.pdf.

Sinha, P.K., 1997. Distributed Operating Systems, Concepts and Design. 1st Edn. IEEE Press, pp: 347-380. ISBN: 81-203-1380-1.

Taek, K.J., M. Gerla, V. Varma, M. Barton and T. Hsing, 2003. Efficient flooding with passive clustering-an overhead-free selective forward mechanism for ad hoc/sensor networks. IEEE. (JPROC), 91 (8): 1210-1220. DOI: 10.1109/JPROC.2003.814920. http://ieeexplore.ieee.org/Xplore/login. jsp?url=/iel5/5/27402/01219472.pdf?arnumber=1219472.

Younis, O. and S. Fahmy, 2004. Heed: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. IEEE. Trans. Mob. Comput., 3 (4): 366-379. DOI: 10.1109/TMC.2004.41. http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/7755/29662/01347100.pdf?temp=x.