Fiaz, M., Yousaf, R., Hanfi, M., Asif, W., Qureshi, H. K. & Rajarajan, M. (2014). Adding the reliability on tree based topology construction algorithms for wireless sensor networks. Wireless Personal Communications, 74(2), pp. 989-1004. doi: 10.1007/s11277-013-1334-2

CITY UNIVERSITY LONDON
EST 1894

City Research Online

Adding the Reliability on Tree Based Topology Construction Algorithms for Wireless Sensor Networks

**Mahzeb Fiaz · Roomana Yousaf · Maryam Hanfi ·
Waqar Asif · Hassaan Khaliq Qureshi ·
Muttukrishnan Rajarajan**

Abstract Topology control is a technique used in wireless sensor networks to maximize energy efficiency and network lifetime. In previous literature, many tree based techniques have been proposed to save energy and increase the network lifetime. In tree based algorithms, the most promising solution is the formation of a network backbone, which serves on behalf of rest of the nodes in the network and therefore leading towards Connected Dominating Set (CDS) formulation. However, one imminent problem with all tree based solution is a compromise on network reliability. Therefore, to address reliability issues in tree based solutions, in this paper, we propose Poly3 which maintains cliques of size three in order to achieve network reliability on top of the CDS algorithm. This makes the network more robust to link removal. Our empirical and mathematical analysis reveals that Poly3 provides better reliability than algorithms of the same kind.

M. Fiaz · R. Yousaf · M. Hanfi · W. Asif (✉) · H. K. Qureshi
School of Electrical Engineering and Computer Science (SEECS), National University of Sciences
and Technology (NUST), Islamabad, Pakistan
e-mail: 09mscsewasif@seecs.edu.pk

M. Fiaz
e-mail: 08bicsembfiaz@seecs.edu.pk

R. Yousaf
e-mail: 08bicseryousaf@seecs.edu.pk

M. Hanfi
e-mail: 08bicsemhanfi@seecs.edu.pk

H. K. Qureshi
e-mail: hassaan.khaliq@seecs.edu.pk

W. Asif · M. Rajarajan
School of Engineering and Mathematical Sciences, City University, London, UK
e-mail: waqar.asif.1@city.ac.uk

M. Rajarajan
e-mail: r.muttukrishnan@city.ac.uk

🖄 Springer

1 Introduction

Wireless sensor networks (WSNs) generally comprises of a large number of intelligent, low cost and low power constrained sensor devices. Due to its energy and deployment constraint, a WSN is prone to faults and malfunctioning. In hostile environments, these faults may be due to natural disasters or human adversaries. Despite the fault prone characteristic of WSN i.e. mission critical nature of many WSN applications (health care, military and disaster recovery) requires that communication between sensor devices is reliable and dependable. In WSNs, communication between source and sink is generally dependent on the intermediate relaying sensor nodes. Therefore, the reliability of transmission is dependent on the routing technique and topology being deployed. This reliability should however be achieved while keeping in mind the energy constraint of WSN's. Due to this reason, reducing the cost of communication is an important way to save energy of sensor nodes and to prolong the lifetime of a sensor network. One such known strategy to save energy in WSNs is that of forming a Connected Dominating Set (CDS) based topology control (TC) scheme.

TC consists of two components: *topology construction* mechanism, which finds a set of backbone nodes to work on behalf of rest of the nodes while maintaining network connectivity and coverage, and *topology maintenance* mechanism, which changes the role of backbone nodes for uniform distribution of resources. Both these mechanisms work in an iterative manner until the network is depleted, thus together they increase the network life time when compared to a continuously running WSN without TC mechanism [15]. In CDS based TC schemes, only the backbone nodes are responsible for relaying messages over the network. The non-backbone nodes can thus turn off their transceiver and hence save energy. The backbone or a CDS size is a critical parameter, since it has been manipulated in many different ways. It has been seen that most researchers reduce the size of backbone, which they argue provides better reliability as the hop count gets reduced among backbone nodes. On the other hand, the reduction in the size of the backbone causes only few nodes to work on behalf of rest of the nodes thus forcing them to deplete their energy more quickly and hence reducing the network lifetime. Similarly, few studies focus on uniformly distributing the number of backbone nodes which allows them to uniformly consume energy and hence increase the network lifetime [3].

In our earlier work, we proposed Poly [8], which models the network as a connected graph and maintains polygenic redundancy around the sink node. Due to this reason, only few backbone nodes around the sink were able to achieve reliability while other backbone nodes were ignored. Hence, in Poly3 we add reliability on the tree based topology construction algorithms for WSNs. Poly3 achieves this with the help of cliques of size three—a tunable parameter—which is maintained throughout the network.

The proposed algorithm is mathematically and empirically analyzed with other solutions on various set of metrics while varying the density of the nodes. It has been observed that Poly3 provides 41 and 134 % better packet forwarding reliability for a packet forwarding probability of 0.3 and 0.7 respectively. Likewise, Poly3 has 61 % better link success reliability when compared with conventional CDS based algorithms. EECDS [14], CDS Rule K [13] and A3 [12] have 86, 92 and 71 % more energy overhead when compared to Poly3 under varying Grid HV topologies.

The rest of this paper is organized as follows. Section 2 summarizes the related work in this area. Section 3 explains the system description and definitions. Reliability analysis is

presented in Sect. 4. Simulation results are presented in Sect. 5. We summarize the salient findings of this paper in Sect. 6.

2 Related Work

In this section, we describe some prominent CDS based approaches which are later used to evaluate reliability of CDS based techniques. While one way of TC is by controlling the transmission range of nodes, backbone based solutions exercise TC by turning off unnecessary nodes while preserving network connectivity and communication coverage. In [2,11] distributed algorithms for constructing CDSs in unit disk graphs (UDGs) were first proposed. These algorithms consist of two phases to form a CDS. First they form a spanning tree and use it to find maximal independent sets (MIS), in which all nodes are colored black. In second phase, some new blue colored nodes are added to connect the black nodes to form a CDS. Likewise Yuanyuan et al. in [14] proposed Energy Efficient CDS (EECDS) algorithm which follows a two phase TC scheme in order to form a CDS based coordinated reconstruction mechanism to prolong network lifetime and balance energy consumption. Similarly Wu et al. in [13] proposed a two phase TC scheme that uses marking and pruning rules for exchanging neighbors list among a set of nodes. In CDS Rule K [13] a node remains marked as long as there is at least a pair of unconnected nodes in its neighbors; it is unmarked when it finds that all its neighbors are covered with high priority. All the above studies focus on increasing the network lifetime by forming a reduced topology but, they do not analyze the impact of a reduced topology on network reliability.

Network reliability is assured by Lanny Sitanayah et al. in [9] by adding extra relay nodes in a *single tiered* network. In a *single tiered* network all nodes forward packets directly to each other instead of relaying it through the backbone node. Similarly Han et al. in [5] provide reliability in full fault tolerant and partial fault tolerant environment for heterogeneous WSNs. They ensure reliability by adding extra relay nodes with an assumption that, relay nodes use the same transmission radii while sensor nodes have different transmission radii. Both these algorithms inherently add overhead by adding extra relay nodes in a network. This results in extra node energy and reduces network lifetime. Hence an algorithm is required that, while keeping in consideration the energy constraint, ensures reliability for the complete network among every set of backbone nodes.

3 The Poly3 algorithm

The Poly3 algorithm provides a solution to the network wide reliability problem in mission critical WSNs. Poly3 assumes no prior information about the position or orientation of the nodes, so the geometrical view of the topology is not visible to the nodes. In the subsequent subsections, we explain the topology construction phase of Poly3.

3.1 Topology Construction

Topology construction phase in Poly3 is subdivided in to three phases. In the first phase, a backbone based on CDS formulation is created. In the second phase, nodes send their neighbors list to the sink node, which they maintain during the first phase. In the last phase, which is a tunable phase, cliques of size three are retained. It is worth noting that during this phase few nodes also become active—if there are any—during the retention of the
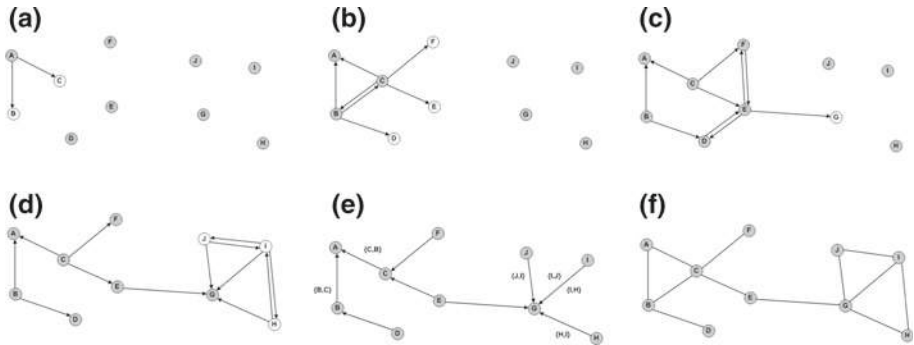
Fig. 1 The Poly3 algorithm. **a** Sink node, A broadcasts a *hello* message-received by nodes B and C and sets a timeout to receive *hello* message in response from its children, Nodes B and C recognize sender node A as their parent. **b** B and C further broadcast *hello* message with parent *ID* set to A. Covered nodes B and C also recognize one another as neighbors. When a node recognizes its children, it waits for finish discovery message. Therefore, A is now waiting for Finish message from B and C. **c** Next level nodes again broadcast a *hello* message after changing their parent *IDs* to their respective parent. Lets say E chooses C as its parent. Moreover F, E and D recognize their neighbors through *hello* message exchange. **d** Node I, J and H broadcast a *hello* message with parent *ID* set to G. Timeout expires for node D, F, H, I and J in which they do not receive any *hello* message with their *ID* set as parent *ID* and hence consider them selves as leaf nodes. **e** After the complete discovery of the whole network, the leaf nodes send a *finish discovery* message to its parent node and immediate neighbor after the timeout. These sets of nodes are called message paths. **f** Final topology, tree cycles of size three, denoting redundant paths

three clique set. The topology construction phase of Poly3 is similar to Poly [8] except for the last step in which instead of a single polygon, multiple cliques of size three are retained.

The backbone construction phase in Poly3 is started by a randomly selected initiator node. The selection is dependent on the criteria that the node with the largest ID persists in case more nodes initiate the construction process. To start the process, the initiator node broadcasts Hello message in its communication area. The Hello message lists the parent ID of sending node. For sink node it is empty since the sink node is assumed to be parent of all nodes.

For the elaboration of the algorithm, we take a sample network shown in Fig. 1. It is supposed that node A in Fig. 1 initiates the backbone construction process by broadcasting the Hello message. It also sets a time out period for receiving the replies from any of its children. The broadcasted Hello message is received by node B and node C, which lie in the communication range of node A. After the reception of node A, both node B and node C rebroadcast the same message while only changing their parent ID, which is now set to node A. The messages by node B and node C are also received by node A which helps identifying node A that it has been chosen as a parent node. Once this process is completed, nodes B and C are considered as covered and thus causing node A to become active. Node A now waits for finish discovery message from its children. At this stage, for the sake of clarity, we assume that contention mechanism is available in case if messages by different nodes are received at the same instance of time. On the other hand, when uncovered nodes receive the message, they set the sender as their parent and repeat the same process. It is also worth noting that the reception of Hello message starts the process of maintaining a neighbors list until the whole network is covered.

These nodes now rebroadcast the Hello message with A as their parent node and set their own timers for receiving responses from their children nodes. This rebroadcasted Hello

message is also received by parent node A, which in turn identifies sender as its children nodes. Once identified these nodes are considered as covered, the parent node switches to an active state and starts waiting for finish discovery message from its children. When an uncovered node receives Hello message it sets the sender as its neighbor. In this way, the CDS creation nodes find their neighbors and the process is repeated until the whole network is covered.

As shown in Fig. 1, the message sent by node B and node C is received by their neighbors, which repeats the same process until Hello message arrives at leaf nodes i.e. nodes H, I and J. These nodes also repeat the same process but the expiry of the time out allows them to send a Finish Discovery message back to their respective parent nodes. In Finish Discovery message, nodes send the list of their neighbors to the parent node, thus starting the second phase of the algorithm. The parent node receiving the Finish Discovery message repeats the same process in the backward direction until the message is received at the sink node, which is the parent of all nodes. In this way, the Finish Discovery message converges towards the sink node.

After the reception of the finish discovery message at the sink node, the third phase of the algorithm starts. In this phase, the sink node compares the message paths visible from the neighbors list for the construction of clique set of size three for reducing the message complexity. The comparison is based on the fact that the nodes common between message paths lead towards a clique of size three i.e. message path G, I and J and message path G, J and I have two nodes in common. In addition, both paths comprise of the same parent node and therefore allowing forming a clique of size three. Once the clique set is chosen, the sink node creates the final topology by broadcasting the Create Topology message. Paths which have nodes C and B in common and both the paths are initiating from node A hence they form a polygon of size three as shown in Fig. 1. As mentioned earlier, that the reliability is a tunable parameter, therefore for the sake of this purpose we created a bound on the third phase of the algorithm. During this phase, if desired, one can set the number of three cliques that are required in the final topology. By doing this, the complexity gets increased but at the sake of required level of reliability and energy efficiency which is now dependent on the number of active clique set which contains the list of nodes that are part of the polygons.

When a node receives create topology message from sink node it checks if its name is in the list of active nodes, if it is in the list it sets its state to active. At the end of this process, each node is in either active or sleep mode. The set of active nodes act as a communication backbone for the network. The complete algorithm flow is shown in Algorithm 1. We now provide the network reliability analysis of the algorithm in the next section.

3.2 Poly3: Complexity Analysis

All the existing and proposed algorithms follow the same pattern of starting with the discovery of a CDS. The complexity of finding the CDS is same for the three existing algorithms, whereas, the complexity of Poly3 is less due to the sharing of explicit broadcast messages for CDS discovery by the parent node. In order to provide reliability, Poly3 searches for subsets of cycles among the message paths.

Assume that the sink node has received N message paths from all the child nodes in the network. In the worst case scenario, in order to find the complexity, we need to compare N^2 message paths of length L to find at least two common nodes in each iteration. Consequently, the worst case complexity of the searching algorithm will be $O((NL)^2)$. As we do not handle

Data: Connected Network
Result: Constructing a reduced network topology of clique of size three
Phase One:
for *all nodes in the network send a Hello message* **do**
    if *node is a the sink node* **then**
        set ID as empty
    else
        if *a node is the last leaf node* **then**
            Send finish discovery to parent node
        end
    end
    else
        Forward the Hello message by setting the ID to parent ID
    end
    end
end
Phase Two:
for *all nodes in the network send a Finish Discovery message* **do**
    if *node is not the sink node* **then**
        send finish discovery to its parent node with list of neighbors appended **else**
        Break
    end
    end
end
Phase Three:
while *Sink node compares message paths visible from the neighbor list* **do**
    if *two paths have two nodes in common* **then**
        Sink sends a message to those nodes to form a clique of size three
    end
end

Algorithm 1: Poly 3, Algorithm flow

all the message paths and are concerned with only a subset, this reduces the complexity significantly. If d is the divisor for the message paths into the number of subsets considered, then the complexity will be $O((NL/D)^2)$. If we exempt the reliability constraint considered than the complexity of Poly3 will be lower than the complexity of the existing algorithms. Therefore, the additional complexity of Poly3 is the result of the trade-off between reliability and energy efficiency.

4 Network Reliability

Network Reliability has been defined in two ways namely packet delivery reliability and link redundancy. The former is dependent on path length among active set of nodes and the second is dependent on number of extra links used throughout the network. In this section, we compare Poly3 on both set of performance metrics, however, before that; we explain different notations associated with the metrics.

– Packet Forwarding Probability: Packet forwarding probability (P_f) is defined as, the probability that a packet will be successfully delivered to the next hop in the path length between the source and the destination node. The packet forwarding probability is the product between the probability of not having a collision at the MAC layer P_c and the probability that a packet is not lost due to channel errors (P_e), and is given by:

$$P_f = P_c P_e \tag{1}$$

– Average Path Length: Average path length is defined as the mean of the shortest path lengths between all pair of vertices and it represents how quick information transfer can be done in a network. It is given by:

$$l = \frac{1}{n(n-1)} \left[\sum_{i,j} d(v_i, v_j,) \right] \tag{2}$$

Here n represents the number of nodes/vertices in a network and d is the distance between nodes i and j for all pair of active nodes in the network.

We now explain the performance of Poly3 on the two set of metrics in the next subsections.

4.1 Packet Delivery Reliability

Most of the tree based solutions ensure that every pair of nodes is connected to each other through at most one path. Hence, a packet send from node A to node D can have only one path in order to reach the destination. As a result, the tree based algorithms are not viable since the network is prone to failure and can be decomposed into two or more disjoint components. On the other hand, reducing the path length with varying node density among set of backbone nodes provides a solution to increase the packet delivery reliability. Therefore, the packet delivery reliability is given by:

$$R(P_f, l) = P_f^l \tag{3}$$

The possibility that the packet will be received by the destination node is dependent on the path length among set of backbone nodes, since they are responsible for relaying information towards the sink node. In order to address the packet delivery reliability, Poly3 forms a clique set of size three which provides polygenic redundancy while also helping reducing the path length among set of nodes in the network. It is due to the reason that high degree backbone nodes get connected with other bunch of nodes, thus reducing the overall path length.

Figure 2 shows the packet delivery reliability of Poly3 and Tree based CDS algorithms. In order to see the impact of increasing path length, the results are computed by varying the node density up to 400 nodes. In addition, different WSN applications can have different packet forwarding probability due to the vagaries of communication, therefore, the results are
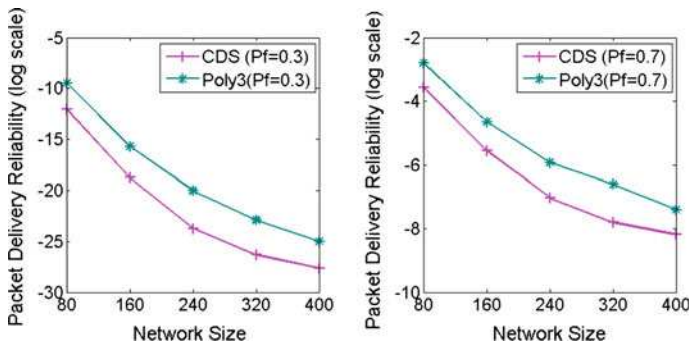

Fig. 2 Packet delivery reliability for $P_f = 0.3$ and $P_f = 0.7$

computed for $P_f = 0.3$ and $P_f = 0.7$. Results were computed under the Network Analyzer tool available as a plugin in Cytoscape [10]. It is evident that increase in node density increase the path length, however, Poly3 provides better Packet delivery reliability due to the fact that backbone nodes are connected in the form of many cliques set.

4.2 Link Success Reliability

The redundancy in graph theory is defined as the expected number of spanning tree that are functional [3]. Due to this reason, removal of a single edge in a spanning tree leads to a graph of many disjoint components. Therefore, all individual edges play a key role for the successful delivery of data. The tree based algorithms based on CDS forms a backbone, which is also a tree, therefore, the lack of non-identical spanning tree under dynamically changing conditions exposes the algorithms for mission critical applications. On the other hand, failure of a link triggers the topology maintenance algorithm again and again and hence putting constraint on energy stringent WSN devices. It is therefore very important that the topology is constructed in a robust way, which to certain extent also helps in achieving energy efficiency under topology maintenance.

Reliability, which is associated with redundancy, is defined as the probability that there is at least a functional spanning tree or a connected network under random link failures. Therefore, the more the redundancy in the network, the more is the reliability in the network. However, for WSNs, the level of redundancy is dependent on the energy efficiency that is required in most of the application scenarios. To demonstrate the performance of Poly3, we take a sample network shown in Fig. 3. For computing the reliability, we used Linear Algebra package available in Maple [7]. To start with, we suppose that $B = (b_{i.j})_{n.n}$ denotes the adjacency matrix of graph G, then

$$b_{i.j} = \begin{cases} 1 & \text{if vertices } v_i \text{ and } v_j \text{ are adjacent,} \\ 0 & \text{otherwise.} \end{cases}$$

The degrees of the vertices are represented by a diagonal matrix. If $D = (d_{i.j})_{n.n}$ denote the diagonal matrix of graph G, then

$$d_{i.j} = \begin{cases} deg(v_i), & \text{for } i = j, \\ 0 & i \neq j. \end{cases}$$

We used the matrix tree theorem [7] to find the number of non-identical spanning trees for the network shown in Fig. 3. According to the theorem, the spanning trees of graph G is the value of the cofactor of a matrix, i.e. $T = D - A$. Therefore, the matrix T for the assumed network equals

Fig. 3 A sample network
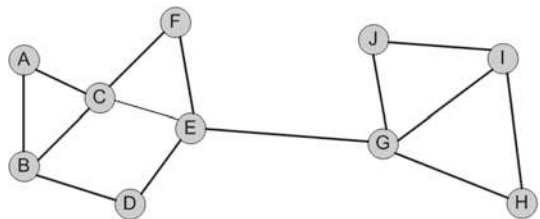
$$T = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 3 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 4 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 4 \end{bmatrix}$$

The cofactor of a matrix T equals 24, which means that there are 24 possible combinations of spanning trees. These combinations represent the total redundancy in the network. However, to demonstrate the reliability, the interest lies in measuring the probability that network remains connected under random link failure. To compute this, we must represent all spanning trees as a disjoint product as given below:

$$P(t_1 \vee t_2 \vee t_3 \vee \cdots \vee t_{24}) = P(t_1) + P(t_2\overline{t_1}) + P(t_3\overline{t_2t_1}) + \cdots + P\left(t_{24}\overline{t_{23}t_{22}\ldots t_1}\right),$$

where t is a spanning tree in the network.

If we suppose that all edges have the same reliability $P_1 = P_2 = \cdots = P_n = P$, then, the reliability of the network shown in Fig. 3 is given by: $24p^9 - 49p^{10} + 34p^{11} - 8p^{12}$. The adjacency matrix for all tree based CDS algorithms remain the same as all these algorithms maintain a single spanning tree.

Figure 4 compares Poly3 algorithm with CDS tree based algorithm. It also draws the link success reliability of the original sample network shown in Fig. 4. The results were computed by varying the probability up to 0.4 and by inserting them in the equation computed previously. The results show that Original Network provides better link success reliability because all the links are functional. However, taking up this sort of a network further would impact on the energy. On the other hand, Poly3 provides better reliability when compared with CDS algorithms. It is due to the fact that few redundant links are maintained, which allows having a more reliable network. Therefore, it is evident that Poly3 can work better for mission critical WSN applications when compared with tree based algorithms [13]. The performance of CDS based schemes increases with the increase in probability but is lesser than that provided by Poly3 because CDS based algorithms have all the nodes connected through a single path only. By maintaining cliques of size three we have increased reliability compared to CDS based algorithms and we have

Fig. 4 Link success reliability
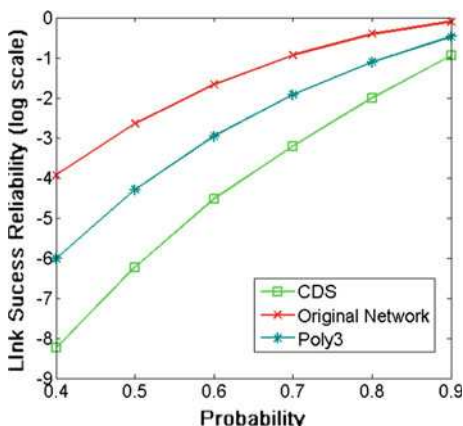
also achieved energy efficiency that is comparable to the existing topologies as shown in Sect. 5.

5 Simulation Results

For evaluation purpose, we used Attaraya Simulator [1] that have been specifically designed for WSN TC algorithms. The Atarraya underlying features provide many advantages which includes, different energy models for motes, communication models, energy and node location distribution resources that can adapted according to the requirement in the simulations. On the other hand, we evaluate the performance of the algorithms under two metrics namely message overhead and energy overhead. The former shows the number of exchanged messages while constructing the topology while latter shows the overhead associated or the energy consumed during the exchange of the messages. The CDS formation in the network forms a tree, which is unreliable. Therefore, in this paper, the inherent problem in the CDS based algorithms is addressed by adding the reliability on top of the CDS trees. Therefore, we selected three different CDS algorithms (CDS, EECDS, and A3) to validate our findings and compared with the proposed Poly3 algorithm.

For evaluation of the algorithms under discussion, we distributed the nodes in an area of 600m × 600m while varying the node density from 80 nodes upto 500 nodes. All the results were averaged over 100 simulations runs. We evaluated the algorithms under three mote energy models available in Atarraya namely mica energy model, simple mote energy model and Waspmote energy model. In addition, we also used the same communication model namely mica, simple mote and Waspmote communication model. These energy and communication models are named after the name of the motes used in the market. The Simple motes energy utilization under *Active*, *Idle* and *Sleep* mode is 8, 3.2 and 0.103 mA/h respectively. This varying energy consumption mechanism enables a mote to survive for a longer period in a network. In an *Active* state when a node needs to transmit or receive, the energy consumption increases to 12 and 7 mA/h respectively [4]. On the other hand, MICA motes are a product of Crossbow [4]. MICA motes consume 12 mA/h in a full processor operation mode, while they consume only 0.01 mA/h in *Sleep* mode. A MICA mote has only two states, this enables it to survive for a longer time than Simple motes. The transmission and reception operation consumes 10 and 7 mA/h respectively.

The Waspmote's are a product of Libelium and are known due to their long transmission range. The Waspmote's uses a four state transition model for conserving energy namely: *On*, *Sleep*, *Deep Sleep* and *Hibernate* state. The *On*, *Sleep*, *Deep Sleep* and *Hibernate* state consumes 15mA/h, 55A/h, 55A/h and 0.06 A/h, respectively [6].

Energy distribution of these nodes follows a uniform process while the node location distribution follows a random process. As mentioned in Sect. 3, that the third stage of the algorithm is tunable, however, for simulations, only 33 % of the cliques set in the network were retained. The algorithms were evaluated under indoor communication and outdoor communication scenarios as explained in the next subsections.

5.1 Grid Topology

In this section, simulation results obtained from the four algorithms by deploying two different node location distribution grid topologies namely Grid HV and Grid HVD are compared. In Grid HV and HVD, nodes communicate with their horizontal, vertical and diagonal neighbors depending upon the topology deployed.
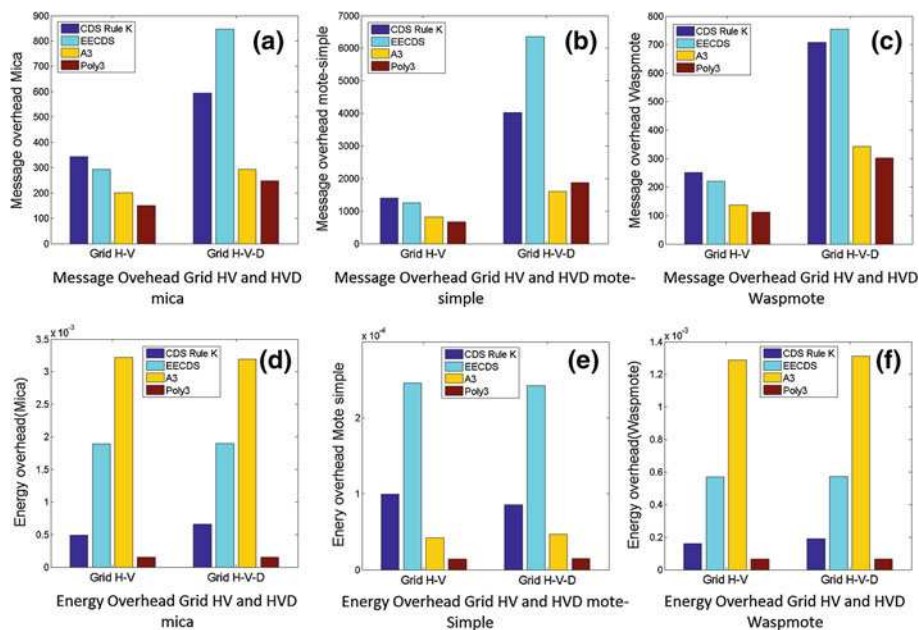
Fig. 5 Performance comparison under Grid HV and Grid HVD topologies. **a** Message overhead grid HV and HVD mica. **b** Message overhead grid HV and HVD mote-simple. **c** Message overhead grid HV and HVD Waspmote. **d** Energy overhead grid HV and HVD Mica. **e** Energy overhead grid HV and HVD mote-simple. **f** Energy overhead grid HV and HVD Waspmote

For validation of our results we tested Poly3, A3, EECDS and CDS Rule K, under grid indoor environment. We observed that the residual energy, message overhead and energy overhead trend of all these algorithms was uniform under all these communication models as shown in Fig. 5. CDS Rule K and EECDS have a greater message and energy overhead due to their two phase topology construction mechanism in both the algorithms. Whereas the message overhead of A3 algorithm is low due to its three way handshake, which allows the algorithm to form a reduced topology with reduced energy overhead. It is evident that Poly3 works better than the other algorithms due to the fact that it constructs the topology in a single phase and use message sizes not exceeding 25 bytes. On the other hand, the A3 algorithm uses long and short messages for construction, which demonstrates different performance under different energy models.

Poly3 has low energy overhead when compared to all the remaining algorithms but has a higher message overhead than A3 in few cases. A3 uses a signal strength based selection metric, allowing it to select distant nodes in the CDS. However this metric consumes more energy when it comes to forming a backbone in grid topologies due to the uniform node distant distribution. On the other hand, Poly3 uses a broadcast node selection mechanism, which efficiently selects nodes under varying network size hence yielding better remaining energy due to less overhead when compared to other tree based algorithms.

5.2 Influence of Node Density

To analyze the impact of message overhead and energy overhead while varying the node density, results were taken from a 80 node network up to 400 nodes.
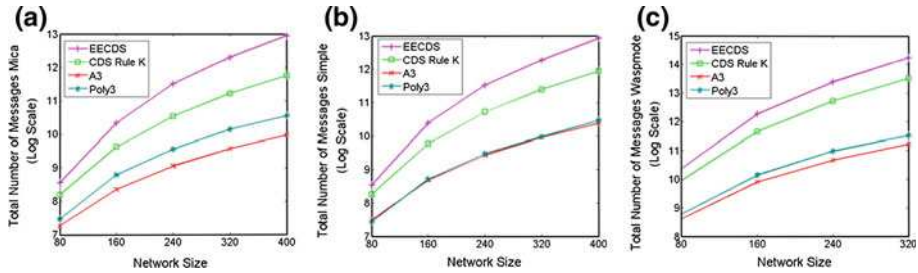
Fig. 6 Impact of varying node densities on message overhead for **a** Mica, **b** Simple and **c** Waspmote
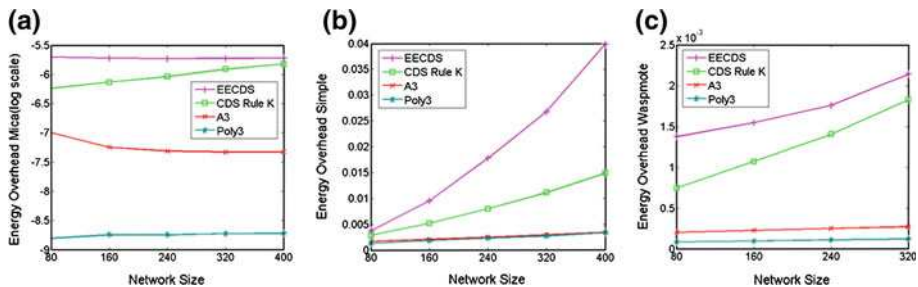

Fig. 7 Impact of varying node densities on energy overhead for **a** Mica, **b** Simple and **c** Waspmote

Figure 6 shows that, as the network size grows, the number of exchanged messages grow exponentially. The number of exchanged messages of EECDS and CDS Rule K is greater than Poly3 due to its two phase construction process. A3 has better message overhead due to it distant node selection metric. This helps fewer nodes to be selected for the CDS, hence leading to quick convergence of the algorithm. On the other hand Poly3 requires the children node to send back their node *IDs* to their parent node, leading to more energy overhead.

Figure 7 shows that as the network size increases the energy overhead of EECDS and CDS Rule K increases, due to the increase in packets exchange between nodes. Poly3 outperforms these algorithms because in it a child node does not explicitly send a message to the sink and the sink node does not process all the message paths for clique set construction. Instead the parent node hears the broadcast message send by the child node, resulting in low energy overhead. Under simple communication model A3 has similar performance than Poly3, but it has greater energy overhead under Mica and Waspmote. Thus, as a consequence of distant node selection, more energy is consumed with variation in node density.

In order to analyze the impact on the total connected covered sensing area, we varied the network size from 100 nodes to 500 nodes.

Figure 8 shows the effect of increase in network size for the total connected covered sensing area. It shows that Poly3 covers more sensing area compared to EECDS, A3 and CDS Rule K algorithms at the end of the topology construction. The EECDS, A3 and CDS Rule K forms a backbone based on reduction in the size of the CDS, hence compromising on the area under coverage. However, Poly3 achieves energy efficiency by minimizing the message complexity and tries to achieve reliability by forming the cliques set in the complete geographical area under coverage and thus providing better sensing area.
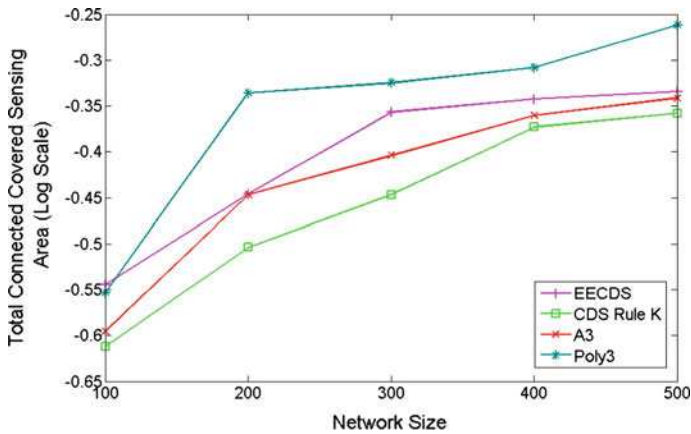
Fig. 8 Effect of increasing number of nodes on the total connected covered sensing area

6 Conclusions

In this paper, we proposed a TC algorithm—Poly3—which forms a CDS backbone by retaining cliques set of size three. It tries to ensure that the backbone nodes throughout the network are connected with each other in a reliable manner. The Poly3 also provides the reliability and energy efficiency as a tunable parameter, which can be adjusted accordingly. Our performance comparison with other tree based TC algorithms on different energy models reveal that Poly3 works better in terms of reliability and has less message overhead when compared with other algorithms.

References

1. Attaraya. http://www.csee.usf.edu/labrador/Atarraya.
2. Alzoubi, K., Wan, P., & Frieder, O. (2002). Distributed heuristics for connected dominating sets in wireless ad hoc networks. *Journal of Communications and Networks*, *4*(1), 22–29.
3. Clad, F., Gallais, A., & Mérindol, P. (2012). Energy-efficient data collection in wsn: A sink-oriented dynamic backbone. In *IEEE international conference on communications (ICC)*.
4. Crossbow: Xbow mib users manual. http://ebookbrowse.com/xbow-mpr-mib-series-users-manual-pdf-d386974151.
5. Han, X., Cao, X., Lloyd, E., & Shen, C. (2010). Fault-tolerant relay node placement in heterogeneous wireless sensor networks. *IEEE Transactions on Mobile Computing*, *9*(5), 643–656.
6. Libelium: Waspmote technical guide. http://www.libelium.com/development/waspmote/documentation.
7. Maple homepage. http://www.maplesoft.com.
8. Qureshi, H., Rizvi, S., Saleem, M., Khayam, S., Rakocevic, V., & Rajarajan, M. (2011). Poly: A reliable and energy efficient topology control protocol for wireless sensor networks. *Computer Communications*, *34*(10), 1235–1242.
9. Sitanayah, L., Brown, K. & Sreenan, C. (2011). Fault-tolerant relay deployment for k node-disjoint paths in wireless sensor networks. In: *Wireless Days (WD), 2011 IFIP* (pp. 1–6).
10. Smoot, M., Ono, K., Ruscheinski, J., Wang, P., & Ideker, T. (2011). Cytoscape 2.8: New features for data integration and network visualization. *Bioinformatics*, *27*(3), 431–432.
11. Wan, P., Alzoubi, K.& Frieder, O. (2002). Distributed construction of connected dominating set in wireless ad hoc networks. In: *Proceedings of the twenty-first annual joint conference of the IEEE computer and communications societies, INFOCOM 2002* (Vol. 3, pp. 1597–1604)
12. Wightman, P. & Labrador, M. (2008). A3: A topology construction algorithm for wireless sensor networks. In *Global Telecommunications Conference, IEEE GLOBECOM 2008* (pp. 1–6)

13. Wu, J., Cardei, M., Dai, F., & Yang, S. (2006). Extended dominating set and its applications in ad hoc networks using cooperative communication. *IEEE Transactions on Parallel and Distributed Systems*, *17*(8), 851–864.
14. Yuanyuan, Z., Jia, X. & Yanxiang, H. (2006) Energy efficient distributed connected dominating sets construction in wireless sensor networks. In: *Proceedings of the 2006 international conference on Wireless communications and mobile computing* (pp. 797–802). ACM
15. Zinonos, Z., Vassiliou, V., Ioannou, C. & Koutroullos, M. (2011) Dynamic topology control for wsns in critical environments. In: *4th IFIP international conference on new technologies, mobility and security (NTMS) 2011* (pp. 1–5)

Author Biographies

Mahzeb Fiaz has recently completed her Bachelors in Information and Communication Systems Engineering from National University of Sciences and Technology, Pakistan. Her areas of interest are Wireless Sensor Networks and Artificial Intelligence.

Roomana Yousaf did her bachelors in Information and Communication System Engineering from School of Electrical Engineering and Computer Science at the National University of Sciences and Technology, Pakistan. She completed her graduation in June, 2012. As a bachelors student at School of Electrical Engineering and Computer Science (SEECS) she did research on achieving reliability for mission critical Wireless Sensor Networks.

Maryam Hanfi did her bachelors in information and communication system engineering from School of Electrical Engineering and Computer Science at National University of Science and Technology in July 2012. Her area of interest is wireless communication and networks.

Waqar Asif received his M.S., degree from School of Electrical Engineering and Computer Science at the National University of Sciences and Technology, Pakistan. He is a Split Ph.D. student at City University London, UK and National University of Sciences and Technology. His current area of research is the impact of change in average path length on reliable infrastructure-less networks and biological networks.

Hassaan Khaliq Qureshi received his Ph.D. degree in Electrical Engineering from City University, London, England in 2011. Earlier, he received the M.Sc., degree in Electrical Engineering with a first class honors from Blekinge Institute of Technology, Sweden in 2006. He is currently working as an Assistant Professor in School of Electrical Engineering and Computer Science (SEECS), NUST. Before moving to Sweden, he worked in A.E.S.L (Caterpillar) Private Ltd. as a service engineer and then after completing masters, served as a lecturer in Bahahuddin Zakariya University, Pakistan. He then moved to Germany and remained Research Assistant in Technical University of Dresden and worked on European Union OPERA project. He serves as a reviewer for IEEE conferences, including Globecom, DCOSS, TENCON and for Springer's Wireless Personal communications, Wireless Networks and China Communications Journal. His main research interests include wireless channel modeling, Network optimization techniques, topology control and network reliability issues for infrastructure-less networks and graph theory.

Muttukrishnan Rajarajan received his B.Eng., and Ph.D. degrees from City University in 1994 and 1999 respectively. From 1999 he worked at City University as a Research Fellow. In August 2000 he left City University to join LogicaCMG as a Telecommunication consultant. He joined City as a lecturer in January 2002 where he is now a Senior Lecturer. In 2005 the online travel web services company GOQuo appointed Dr. Rajarajan as their Chief Technical Advisor. Dr. Rajarajan is a senior member of IEEE, a member of IET and a member of Technical Committees for ICOCN, PIERS 2008, eHealth 2008, Pervasive Health 2009 and has published more than 60 Journal and conference papers.