ATC
An Asymmetric Topology Control Algorithm for Heterogeneous Wireless Sensor Networks

Mohsen Nickray and Ali Afzali-kusha
Low-Power High-Performance Nanosystems Laboratory
Department of Electrical and Computer Engineering, University of Tehran, Tehran, Iran

Keywords: Wireless sensor network, Power control, Asymmetric topology control.

Abstract: In this paper, we present an asymmetric topology control (ATC) algorithm for wireless sensor networks. In this algorithm, the sensor nodes incrementally adjust their transmission. The algorithm had three phases of Neighbor Discovery, Construct Topology, and Data Transmission. In the phase of Neighbor Discovery, the nodes exchanged their positions and maximum transmission powers. In phase II, each sensor node collaboratively adjusted its transmission range (power) while keeping the network connectivity the same as that of the case of transmitting with the maximum power. In phase III, all the nodes transmit data with the adjusted transmission power. To assess the efficiency of the proposed algorithm, its performance is compared to those of previously published works. Our algorithm not only preserve power and average node degree and average link length, it has privileges which enable it to work properly even in the absence of conventional error handling mechanisms.

1 INTRODUCTION

Recent advances in wireless and electronics technologies have led to emergence of wireless sensor networks (WSN) with large scale nodes. They are used in a wide spectrum of applications including industrial, military, and health monitoring applications. Most of these devices have limited battery lifetime where after depletion, it is extremely difficult (if not impossible) to replace the batteries. As a result, WSNs need efficient mechanisms which minimize the energy consumption while maintaining the network connectivity and improving the network capacity which is the simultaneous data transfer rate.

The topology control determines the required transmission power of each node to maintain the network connectivity while the energy consumption is minimized. Instead of transmitting with a maximum power, nodes in a wireless network collaboratively determine their transmission power and derive the network topology by forming proper neighbor relations under a specific topology control algorithm (Narayanaswamy et al., 2002). Using adjusted transmission power have several benefits (Rodoplu and Meng, 1999). They include minimizing the MAC layer contention, improving the spatial reuse and network capacity, and increasing the network life time by minimizing the power consumption.

WSNs are divided into two categories of homogeneous and heterogeneous. The homogeneous network contains devices with the same hardware capabilities such as computation, link, energy, and communication range while the heterogeneous network includes devices with different hardware capabilities. Note that the heterogeneity in the transmission range of the nodes leads to asymmetric links where the transmission and receive paths are not the same. Asymmetric links are normally unidirectional. Recently in the literature, heterogeneous WSNs have attracted more attention. In these networks, asymmetric protocols seem unavoidable for two reasons (Liu and Li, 2003). Firstly, if all the links in the original topology are symmetric, it is not possible to assume different transmission ranges among nodes. In this case, the two farthest neighbors in network determine the transmission power of all the nodes. Secondly, if asymmetric links are allowed to exist in the finalized topology, the derived minimum power topology may become more power-efficient since the transmission range for each node may be determined according to the situation of its neighbors.

Nickray M. and Afzali-kusha A. (2010).
ATC - An Asymmetric Topology Control Algorithm for Heterogeneous Wireless Sensor Networks.
In *Proceedings of the International Conference on Wireless Information Networks and Systems*, pages 75-81
DOI: 10.5220/0002984800750081

In this paper, we consider the heterogeneity of the communication range for the WSN nodes. We introduce a distributed algorithm which constructs a topology with asymmetric links. In the algorithm, the information is only exchanged between the nodes which are attributed as local neighbors. As the algorithm is localized, it could be applied to networks with large scales. The rest of the paper is organized as follows. In Section II, we briefly review some related works on the topology control and the differences of this work with them are discussed. The proposed algorithm is presented in Section III. In Section IV, the results are discussed. Finally, section V concludes the paper.

2 PREVIOUS WORKS

The concepts of relay region and enclosure for the purpose of power control were presented in (Rodoplu and Meng, 1999). The relay region is defined based on the following property. If the node *i* consumes less power when it chooses to relay through the node *r* instead of transmitting directly to node *j*, then the node *j* is in the relay region of node *r*. The enclosure of the node *i* is then defined as the union of the complement of the relay regions of all the nodes that the node *i* can reach by using its maximal transmission power. Although the proposed technique generates an energy efficient topology, it has a high messaging overhead (Rodoplu and Meng, 1999).

In (Li et al., 2003), a bidirectional topology based on Minimum Spanning Tree is introduced. The network connectivity is preserved in this topology where the degree of each node is bounded by six. A bounded degree is desirable because a small node degree reduces the MAC level contention and interference (Li et al., 2003). In (Li et al., 2001), CBTC(α) which is a two-phase algorithm is proposed. In this algorithm, each node finds the minimum power *p* such that transmitting with it guarantees that it can reach at least one node in every cone of degree of α. It was analytically shown that if $\alpha < 5\pi/6$, the network connectivity will be preserved.

A three phase algorithm for the topology control is introduced in (Liu and Li, 2003). In the first phase, each node broadcasts an initialization message where the nodes in its vicinity reply with a message containing their locations and maximum powers. Based on the information, each node establishes its vicinity graph. In the second phase, the minimum power vicinity tree is derived from the vicinity graph using the execution of the shortest path algorithm. In the third phase, each node calculates its transmission power and required transmission power of their vicinities by running the shortest path algorithm, and informs the neighbors using Power Request (PRQ) Messages. Each node, when receives a PRQ message from a neighbor, compares the power requirement from the neighbor node with its current power setting. If a neighbor requires a stronger transmission power, the node increases its power accordingly. The minimum-power topology guarantees the same reachability between any two nodes compared with the maximum topology where the nodes use their maximum transmission powers. The important shortcoming of the algorithm is its vulnerability to the packet loss in the third phase. PRQ losses lead to irreparable problems. Packet losses may occur in WSNs for the reasons explained here. Normally, WSNs are set up in adverse environmental conditions, like wind and rain, where the communication can be disrupted. In the configuration steps of sensor networks, where there is no topology control algorithm, all the nodes will transmit using the maximum power, and hence, packet losses are more probable.

Based on the above discussion, an asymmetric algorithm resistant to the packet loss is desired. In this paper, we introduce an asymmetric topology control algorithm which overcomes the shortcoming of the algorithm presented in (Liu and Li, 2003). The proposed algorithm works properly when the packet loss occurs but at a lower efficiency. In this paper the efficiency is assumed as a function of the average node degree and the average link length. The efficiency of the algorithm degrades inversely proportional to the packet loss rate.

3 PROPOSED ASYMMETRIC TOPOLOGY CONTROL ALGORITHM

The ATC algorithm which is proposed in this work is shown in Fig. 1. It is a distributed and localized algorithm which efficiently assigns the power level of each sensor node. The goal of the algorithm is to find a minimum transmission power of P_i such that the network connectivity is preserved. Algorithm has three phases which include Neighbor Discovery, Construct Topology, and Data Transmission.

A. Phase I

In the first phase, the node n_i broadcasts a discovery Hello message. The message contains (x_i, y_i) and

P_i^{max} which are the location and maximum power of the node using the maximum power of the node which is P_i^{max}. Having received this broadcast message, each neighbor j replies to it by a message which contains its location and maximum power using its maximum power of P_j^{max}. The node n_i detects the set of its localized neighbor denoted by $\mathrm{N}(n_i)$ based on the received reply messages. The node stores the neighbor unique ID nodes which is determined in the link layer, in a maximum range table (MRT) along with their other information including their positions and maximum ranges based on the received current power levels. The latter will be adjusted in the second phase.

$$\mathrm{N}(n_i) = \{n_j \mid n_i \, received\ a\ reply\ message\ from\ n_j\}$$

As mentioned before, the reply messages are transmitted with P_j^{max}. Even with this power level, if the node n_i is not in the range of one of its neighbors, it needs a multi hop path to reach n_i. This may occur due to the fact that we have a heterogeneous network with different transmission ranges. Various mechanisms like re-broadcasting by relay nodes, sending the message via network layer packet routing protocols, or using sub-routing layer services could be used (Liu and Li, 2003).

B. Phase II

In the Construct Topology phase, each node n_i should decide about its final transmission power denoted by P_i. The power is determined by a distributed process stated in phase II of Algorithm1 shown in Fig. 1. Any change in the node power level will be informed by broadcasting an update message with the maximum power of P_i^{max}. Each node updates its MRT table using the update packets sent by its neighbors. To determine the current transmission power of the node i, the algorithm starts by initializing the power to the amount needed to reach the nearest neighbor in $N(n_i)$. Then, each node incrementally adjusts the power such that it achieves the same neighbor set as the maximum range topology.

In phase II of the algorithm, in order to construct the final topology of $\acute{G}_{n_i}$, we need $(|\mathrm{N}(n_i)| - 1)$ iterations. The final topology will contain the least possible number of edges which is equal to $|V_i| - 1$. In each iteration, the algorithm wait for new update messages before selecting a new edge. The wait time is controlled by a timer denoted by T_i. If during the timer interval of T_i, a broadcast message is received from a neighbor in $N(n_i)$, then the MRT table and G_{n_i} are updated. In each iteration, the algorithm should add one link. At the end of wait time, based on the new MRT values, $G(n_i)$-$\acute{G}(n_i)$ is evaluated.

Phase I:
broadcast(n_i,(x_i, y_i), P_i^{max})
broadcast message is received from all neighbors and create $MRTable$

Phase II:
$P_i = P_i^{init}$
$T(n_i) = (n_i, n_j)$
broadcast(n_i,P_i, P_i^{max})
while ($|\mathrm{T}(n_i)| < |V_i| - 1$)
{
    update $G_{n_i}(MRTable)$
    set timer T_i
    If (broadcast message is received from a neighbor before timer expires)
    {
        update MRTable
        update $G_{n_i}(MRTable)$
    }
    if (timer T_i expires)
    {
    if ($|G(n_i) - \acute{G}(n_i)| == \emptyset \ and \ |\acute{T}(n_i)| < |V_i| - 1$)
    {
        calculate Δp_i
        $P_i = P_i + \Delta p_i$
        update MRT
        update $G_{n_i}(MRTable)$
        broadcast(n_i, P_i)
    }
    }
    e_i= edge with minimum weight from $G_{n_i}(V_i, \mathrm{T}(n_i))$
    $\acute{T}(n_i) = \acute{T}(n_i) + e_i$
}

Phase III
Transmit(n_i,Data, , P_i)

Figure 1: Algorithm 1 clarify the body of the ATC algorithm.

Table 1: ATC algorithm notations.

symbol	Definition
P_i	Current power level of node n_i
$G(n_i)$	Topology for the node n_i resulted by Algorithm2 including vertices ($V_i = N(n_i)$) and edges ($T(n_i)$) resulted from $MRT(n_i)$
$\acute{G}(n_i)$	$\acute{G}_{n_i} = (V_i, \acute{T}(n_i))$ Topology for the node n_i resulted from ATC algorithm including vertices ($V_i = N(n_i)$) and finalized edges $\acute{T}(n_i)$
outward edges	the edge from a closer neighbor to farther neighbor edges
backward edges	the edge from a farther neighbor to closer neighbor
$\overrightarrow{(n_i, n_j)}$	A directional edge from n_i (Source node) to n_j (Destination node).

$$G(n_i) - \acute{G}_{n_i} = \{G(V,E) \mid V = V_i, E = T(n_i) - \acute{T}(n_i)\}$$

Noting that $\acute{G}(n_i)$ is the number of edges before this iteration, if G_{n_i}-$\acute{G}(n_i) \neq \emptyset$, it means that we can add a new edge to $\acute{G}(n_i)$. For this condition, Algorithm 2, which is given in Fig. 2, selects the edge with the minimum distance. Otherwise ($G(n_i)$-$\acute{G}(n_i) = \emptyset$), it means that with the current power, we cannot add a new edge and the power should be increased. For this purpose, P_i is minimally incremented with Δp_i such that at least one new (n_i, n_x) edge denoted by e_i can be added to $\acute{T}(n_i)$. This edge is selected as a new member for $\acute{T}(n_i)$ and a update message is transmitted to the neighbors, to be informed about the last transmission power change.

Update$\boldsymbol{G_{n_i}}$
1 $V_i = \{n_j \mid d(n_i, n_j) \leq R_i^{max}\}$
2 Sort all vertices in V_{n_i} in increasing order of distance to n_i
3 for (*k=1* to $|V_{n_i}| - 1$)
4 for (*w=k+1* to $|V_i|$)
5 if ($r_k^{max} \geq \sqrt{(x_k - x_w)^2 + (y_k - y_w)^2}$
6 *and there is not any* (x, n_w) *in* $\Gamma(n_i)$ *that* $x \in V_{n_i}$)
7 $T(n_i) = T(n_i) + (n_k, n_w)$
8 for (*k*=$|V_i|$ to 2)
9 for (*w*=*k*-1 to 1)
10 if ($r_k^{max} \geq \sqrt{(x_k - x_w)^2 + (y_k - y_w)^2}$
11 *and there is a path(* $n_i \rightarrow n_k$*)*
12 *and there is not any* (x, n_w) *in* $\mathrm{T}(n_i)$ *that* $x \in V_{n_i}$
13 *and there is not* (n_w, n_k) *in* $\mathrm{T}(n_i)$ *)*
14 $T(n_i) = T(n_i) + (n_k, n_w)$

Figure 2: Algorithm 2 which constructs $\boldsymbol{G_{n_i}}$.

Algorithm2 (UpdateG_{n_i}) which is called by Algorithm1 is used to construct $G(n_i)$. This algorithm generates an asymmetric local graph $G_{n_i}(V_i, T(n_i))$ which contains all the feasible edges based on the transmission ranges of the current neighbors in the MRT table. Each edge in G_{n_i} has the potential to be selected as e_i in line 28 of Algorithm1. In lines 3-7 of Algorithm 2, the outward edges and in lines are 8-14 the selection of backward edges are checked. When the algorithm adds outward edge $\overrightarrow{(n_k, n_w)}$, line 5 checks n_w to see if it is in the transmission range of n_k. Also, line 6 makes sure that none of the edges in $T(n_i)$ has n_w as its destination node. In the addition of the backward edges, the edge $\overrightarrow{(n_k, n_w)}$ is added when n_w is in the range of n_k (line 10), n_k is accessible from n_i (line 11), none of the edges in $T(n_i)$ has common destination nodes with the (n_k, n_w) (line 12), and the edge with reverse direction is not a member of $\mathrm{T}(n_i)$ (line 13) ($(n_w, n_k) \notin \mathrm{T}(n_i)$). After $|N(n_i)| - 1$ iterations, $\acute{G}(n_i)$ which is an asymmetric topology containing $|V_i| - 1$ directional edge is constructed.

C. Phase III

Phase III of the algorithm is Data Transmission. While all the messages in phases I and II are transmitted with the maximum power level, in phase III all the messages will be transmitted with the transmission power of P_i determined in Phase II.

D. Example for The Proposed Algorithm

Fig. 3 illustrates the results of running of the proposed algorithm on node n_1. In the first phase of the algorithm, the node detects its neighbors and constructs the MRT table. Phase II devotes to the power adjustment. In phase II, firstly, r_1 is initialized by 1 as the distance between n_1 and its nearest neighbor and the edge (n_1, n_2) is added to $T(n_1)$. Figure 3 part (a) depicts topology for n1 after initial part of phase II. Figure 3 part (b) shows all the details happened in iteration 1. After the expiration of timer T_1, the MRT table is updated by the recent update messages showing the determined power of each node up to this iteration. For this example, according to received update messages ((n2,1)(n3,2)(n4,2)) the transmission range of n_2, n_3, and n_4 has updated to 1, 2, and 2, respectively. Note that at this stage, the update messages of nodes n5 and n6 have not received by node 1. Finally algorithm in this iteration select the edge with the minimum weight which is (n_4, n_7). In iteration 2 (Fig. 3 part (c)), edge (n_3, n_5) is selected. In iteration 3 (Fig. 3 part (d)), after the timer expiration, we have $G(n_i)$-$\acute{G}(n_i) = \emptyset$. Hence, the algorithm assumes $\Delta p_1 = \sqrt{10}$ and add (*n1, n4*) to (n_1) . In iteration 4, the MRT values lead the algorithm to select (n_7, n_6). Part (e) of figure 3 depicts details of the iteration. Finally, in the last iteration (figure 3 part (f)), n1 received (n2, 2) and update $G(n_i)$ and result will be addition of (n2, n3) to the $T(n_1)$. As figure shows the algorithm clarify all the transmitted packets by n1 for each iteration. n1 in phases I and II, totally sends 3 messages which all the messages are transmitted using maximum power.

E. Example for the Proposed Algorithm in Presence of Packet Miss

In this part we assume some problems have been suddenly happened in our network and the worst possible scenario has happened: All the packets from all the neighbors are damaged and we don't have any recovery mechanism then any one of our packets

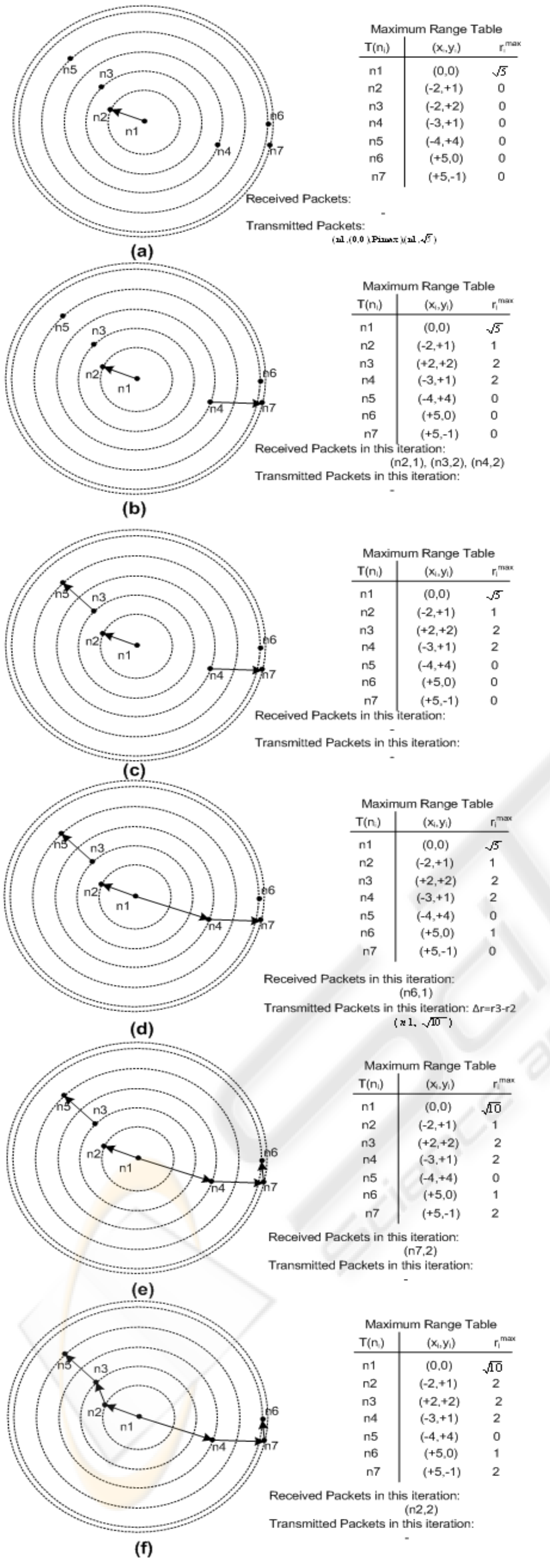


Figure 3: An example which illustrates running of ATC algorithm for node n_1. (a) phase I and initialization step of iteration 1 (b) Iteration 1 (c) Iteration 2 (d) Iteration 3 (e) Iteration 4 (f) Iteration 5 of algorithm.

is recoverable and any one of them is not retransmitted again. In a situation like this algorithm 1 works properly but the calculated maximum range will not be the optimum value. Figure 4 shows the result of algorithm 1 in the presence of 100% packet loss.it should be mentioned that the assumed packet loss is only for updates packets. In this situation the maximum transmition range in our example will change from

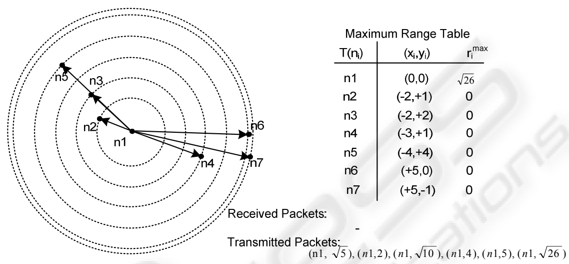


Figure 4: Result of the algorithm1 when node n_1 losses all the received packets.

4 RESULTS AND DISCUSSION

In this section, we discuss the results of applying the algorithm to a WSN and compare them with a similar algorithm. In addition, the time and message complexities of the proposed algorithm are discussed.

A. Simulation Results

The efficiency of the proposed algorithm is determined using some simulation results. We compare the algorithm with a proposed scheme against the algorithms given in (Liu and Li, 2003). The algorithm has the most similarity to our work. In this algorithm, Li and Liu present an asymmetric algorithm based on Prim's algorithm which is also a distributed and local technique (Liu and Li, 2003). Theoretically, (Liu and Li, 2003) calculates the best local answer in our case. The efficiency of the proposed algorithm is a function of the timer T_i and if we properly tune that, our approach could be as efficient as (Liu and Li, 2003). In this study, we use two schemes for the timer. In scheme I, we consider the timer proportional to ΔP_i while in scheme II the timer is assumed to be proportional to $P_i^{max} - P_i$.

Scheme I: Timer $\sim\Delta P_i$
Scheme II: Timer $\sim P_{imax} - P_i$

The results are extracted using simulations in MATLAB. The sensors, which are deployed in a 1000m × 1000m area, are uniformly distributed. The number of nodes is varied from 50 to 250. For every data point, the simulation is repeated 10 times.

Table I shows the average node degree for the topologies generated using the algorithms. Theoretically, the best average node degree, denoted by AND, is given by (Li et al., 2003)

$$\text{AND} = \frac{\sum_{i=1}^{n} Node\ i\ degree}{n} = \frac{2\times(n-1)}{n}$$

where n is the number of nodes. Note that

$$\lim_{n\to\infty} AND = 2$$

The results show that the (Liu and Li, 2003) and scheme II have the lowest AND. Fig. 4 compares the average length of links (ALL) for the proposed and (Liu and Li, 2003). As is evident from the figure, the length decreases when the node density increases. As the results show, the AND and ALL are about the same for both the (Liu and Li, 2003) and our proposed algorithm with scheme II. These parameters are slightly more for the proposed algorithm for sparse networks. For networks with large scale scheme II operates as efficient as (Liu and Li, 2003).

Table 2: Average node degree of the algorithms.

Algorithm	Proposed Algorithm		(Liu and Li, 2003)
	Scheme I	Scheme II	
Average Degree	3.5	3	3

In figure 5, we compare the average power needed to transmit 20 packets in the network between two corners of our network position (0, 0) and (1000, 1000) under free space transmission model. Comparison is between scheme I and II with LMST. The results are normalized to results of LMST method. As figure 5 shows the power result for scheme II is quite near to LMST which has minimum power among these classes of methods. Scheme II consume slightly more power than LMST.

In figure 6, the results for the AND and ALL as a function of update packet loss rate in phase II are plotted. This figure shows the tolerability of the algorithm against packet loss, while (Liu and Li, 2003) is quite sensitive to packet loss. The packet loss rate ranges from 0% to 100% and the results are for three network sizes of 50, 100, and 150 nodes. As the results for the average node degree reveal, the higher the packet loss rate is, the more ANDs we will have. The algorithm operates at its best efficiency when there is no packet loss. In this case,

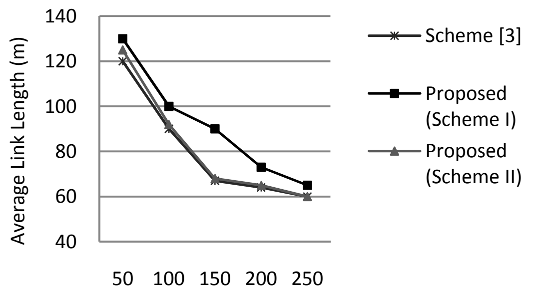


Figure 4: Average length of links for different algorithms as a function of the network size.

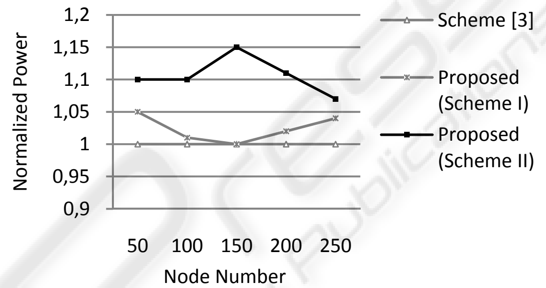


Figure 5: Power comparison of scheme I and scheme II with LMST approach.

there are ANDs of 3.2, 3.3, and 3.5 for the network sizes of 50, 75, 100, 125 and 150, respectively. The differences between the ANDs of the networks increase as the packet loss rate increases. The performance of the proposed algorithm in an ideal environment with no packet loss is almost as efficient as the (Liu and Li, 2003). When the packet loss increases, the efficiency of the algorithm degrades. In the case of a 100% packet loss rate, the algorithm performs like a primitive topology control algorithm which adjusts the transmission power by its farthest neighbor.

Figure 6(b) shows the variation of ALL when the algorithm is applied to network with different sizes. In situations where the low packet loss rate is low, the algorithm works like LMST. The greater the network size is, the smaller ALL will be. As the packet loss rate increases, the algorithm works more like the maximum topology control. In these cases, the greater the network size is, the larger will be.

B. Algorithm Complexity

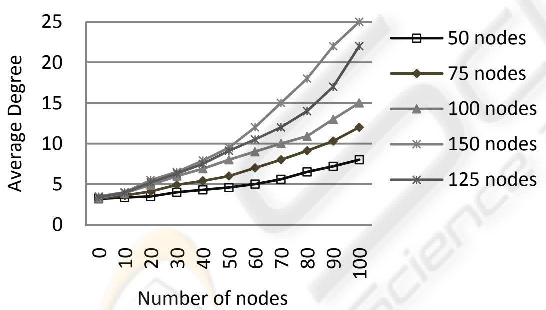
Time complexity: Let us denotes the number of neighbors of n_i as Δ which is equal to $|\mathrm{N}(n_i)|$. The number of algorithm iterations is $\Delta - 1$. The algorithm has $\Delta - 1$ rounds. As a result, the time complexity to construct G_{n_i} will be $\mathrm{O}(\Delta^2)$. Also, in each iteration, before the expiration of the timer T_i,

at most Δ broadcast messages is received. This makes the complexity of Algorithm1 equal to $O(\Delta^4)$.
Message Complexity: Assuming an ideal MAC protocol with no collisions and retransmission, the node n_i transmits at most $(\Delta - 1) + 1$ messsages. A hello message is transmitted at the beginning of the protocol in the phase of the neighbor discovery and at most $\Delta - 1$ messages for updating the determined levels of power (P_i). Since each sensor has at most Δ neighbors where each transmits $O(\Delta)$ messages, the number of messages received by the sensor n_i is $O(\Delta^2)$.

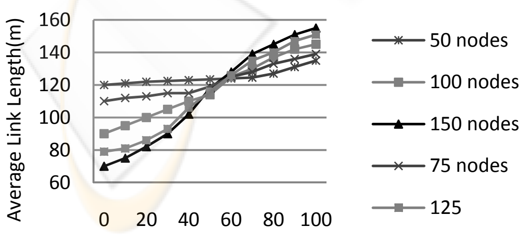
5 CONCLUSIONS

In this work, an Asymmetric Topology Control (ATC) algorithm was proposed. In this algorithm all the nodes simultaneously begin to transmission range assignment and during transmission power assignment, any modification in transmission range will broadcast for all the neighbors.

Results show that algorithm works in a good performance in comparison with similar algorithms while preserve average node degree and average link length. We also compare our algorithm to the LMST algorithm which is a minimum power algorithm. Results show our algorithm works as good as LMST approach.



(a)



(b)

Figure 6: Performance evaluation for the proposed algorithm versus the packet loss rate a) Average node degree b) Average link length.

REFERENCES

S. Narayanaswamy, V. Kawadia, R. Sreenivas, and P. Kumar, "Power control in ad hoc networks: Theory, architecture, algorithm and implementation of the COMPOW protocol," in *Proc. European Wireless Conference – Next Generation Wireless Networks: Technologies, Protocols, Services and Applications*, Florence, Italy, 2002, pp. 156–162.

V. Rodoplu, T. Meng, "Minimum energy mobile wireless network," *IEEE journal selected Area in Communications*, vol. 17, pp.1333-1344, 1999.

J. Liu and B. Li, "Distributed Topology Control in Wireless Sensor Networks with Asymmetric Links", in *Proc. IEEE Glabal Telecommunication Conference*, 2003, Vol. 3, pp. 1257- 1262.

N. Li, J. Hou, L. Sha, "Design and analysis of an mst-based topology control algorithm," in *Proc. of the IEEE Infocom 03,* San Francisco, CA, 2003, pp. 1702-1712.

L. Li, J. Y. Halpern, P. Bahl, and R. Wattenhofer, "Analysis of a cone-based distributed topology control algorithm for wireless multi-hop networks," in *Proc. ACM Symposium on Principles of Distributed Computing*, 2001, pp. 264-273.

S. Narayanaswamy, V. Kawadia, R. Sreenivas and P. Kumar, "Power control in ad hoc networks: Theory, architecture, algorithm and implementation of the COMPOW protocol," in *Proc. of European Wireless*, 2002.

M. Cardei, S. Yang, J. Wu, "Algorithms for Fault-Tolerant Topology in Heterogeneous Wireless Sensor Networks," *IEEE Trans. on Parallel and Distributed Systems*, Vol 19, Issue 4, April 2008.