A Cooperative Approach for Topology Control in Wireless Sensor Networks: Experimental and Simulation Analysis

Paolo Costa*, Matteo Cesana†, Stefano Brambilla†, Luca Casartelli†, Luca Pizziniaco†

*Department of Computer Science
Vrije Universiteit
Amsterdam, The Netherlands
costa@cs.vu.nl

†Dipartimento Elettronica ed Informazione
Politecnico di Milano
Milano, Italy
{cesana,brambilla,casartelli,pizziniaco}
@elet.polimi.it

Abstract

The choice of the transmission power levels adopted in Wireless Sensor Networks (WSNs) is critical to determine the performance of the network itself in terms of energy efficiency, connectivity and spatial reuse, since it has direct impact on the physical network topology.

In this paper, a cooperative, lightweight and fully distributed approach is introduced to adaptively tune the transmission power of sensors in order to match local connectivity constraints. To accurately evaluate the topology control solution, a small-scale testbed based on MicaZ sensor nodes is deployed in indoor and outdoor scenarios. Practical measures on local connectivity, multi-hop connectivity, convergence time and emitted power are used to compare the proposed approach against previously proposed ones. Moreover, a simulation analysis complements the experimental one in large-scale WSN scenarios, where a testbed implementation becomes unfeasible.

1 Introduction

Wireless Sensor Networks (WSNs) are increasingly emerging as a viable solution to support several types of applications ranging from environmental and building monitoring to object tracking and exploration of remote areas through mobile robots [3]. The wireless connectivity and the compact sensors' size make WSNs suitable even in harsh environments, where human support and control are limited.

However, to fully unleash the potential of the WSN technology, large effort must be put forth by researchers and practitioners to devise energy-aware solutions preserving battery power. In particular, network topology has huge impact on efficiency: at the MAC layer, the more connected is the network the higher is the collision probability, whereas the routing layer requires high connectivity degrees to set up effective routes. Hence, the design of effective distributed topology control protocols for WSN is a crucial issue which might determine the success of sensor network technology itself. Given the peculiarity of the WSN domain, topology control protocols must be fully or partially distributed and highly flexible and adaptive to cope with high network variability due to node mobility, wireless link quality fluctuation, or activity cycling.

In this paper we are interested in topology control as a way to determining the sensors'degree K, i.e. the number of neighbors directly connected to a given sensor through a bidirectional wireless link. Existing work [4, 17] proves that an optimal value of K does exist and should be maintained during the entire life of the system to ensure global network connectivity properties [12].

Traditional approaches for ad hoc and sensor networks, e.g., [8, 10, 13, 15, 18], let each node arbitrarily increase its transmission power until K neighbors are heard, possibly resorting to the maximum power whenever the threshold is not met. Other solutions [9, 12], instead, define the local connectivity target as a interval of feasible degrees, i.e., requiring that the degree K of each node fulfill the condition $K_{min} \leq K \leq K_{max}$. In [7] the authors introduce a mechanism based on explicit notification of non-connected nodes whereas in [5] is addressed the problem of designing fault tolerant topologies in WSNs for data collection by providing redundant multi-hop links towards the collecting sink.

A major drawback of the aforementioned pieces of work as well as the vast majority of solutions available in literature is that their evaluation relies either on simulation or mathematical analysis, only. Hence, it is difficult to assess the real feasibility of these solutions to real network scenar-

978-1-4244-2100-8/08/$25.00 ©2008 IEEE

ios, where many of the assumptions made during the analysis may not hold true any longer. Even though much work has been done on the implementation of real-life testbed for WSNs, to the best of our knowledge, very few papers have appeared on experimental studies of topology (and power) control solutions for sensor networks, with the notably exceptions provided by [6] and [10]. In the former, a distributed power control scheme is proposed and evaluated in real-life networks, with the purpose to maintain the quality of wireless links among sensors above a given threshold. The latter studies on real WSNs implementations the problems of synchronization and routing topology construction for in-building applications.

In this paper, we designed a novel protocol for topology control in WSNs which leverages the concepts of cooperation among sensors through the periodical exchange of neighborhood list. To demonstrate the suitability of our approach to real scenarios, we developed a comprehensive small-scale experimental testbed of indoor and outdoor WSNs based on the popular MicaZ motes. We exploited the testbed to compare the performance of our topology control solution against those of other approaches. Then, we validated the testbed results against simulation, highlighting the impact of common assumptions and, finally, we complemented our analysis through simulations in large-scale WSNs.

To summarize, the contribution of the current paper is therefore twofold: first, we illustrate a cooperative approach to provide effective topology control in WSNs by means of controlling the transmission power (Section 2). Second, we report on our extensive evaluation, both in a small scale real testbed (Section 3) and in a simulated large-scale scenario (Section 4), illustrating the pitfall of available solutions and demonstrating the suitability of the proposed approach for the scenario we target. Finally, Section 5 ends the paper with brief concluding remarks.

2 Protocol Description

In this section, we illustrate our approach for distributed topology control in WSNs, by means of a reference example. We start off by describing the basic elements and then we refine the description to account for the peculiarity of WSNs.

2.1 Topology Control Basics

The protocol we present is composed of two distinct phases: the *Neighbor Discovery* and the *Topology Update*. Both are performed periodically to react to arbitrary changes in topology as induced by node failures. Since the protocol is designed to be extremely lightweight and focuses on large-scale and dynamic environments, we do not require any form of synchronization among nodes that would require additional overhead. Hence, we do not make any assumption on how discovery and update phases are scheduled on different nodes. We just impose that the interval between two subsequent update phases is fixed and it is chosen in a way to guarantee that all beacons from potential neighbors have been received. Randomization may be exploited to determine the instant to broadcast beacon (between two update phases) in order to reduce the likelihood that beacons from different nodes collide.

In the protocol description, we will adopt the assumption that the transmission range of a given communication can be estimated by each node knowing the transmission power level, the reception threshold power and the propagation model. This allows us to use only for description purposes the concept of transmission range. We then comment on practical implementation issues in Section 3.

Neighbors Discovery As mentioned in Section 1, traditional approaches [11] for topology control in ad hoc networks resort to maximum power transmissions during the discovery phase to detect all the nodes potentially reachable. Once this information is acquired, each node tunes its transmission range (or power) to achieve the desired neighbor degree. We argue that this solution is detrimental in that it wastes precious energy resource and it may also lead to non-optimal solutions since transmitting at maximum power is likely to create interference among node transmissions thus preventing some nodes to correctly receive packets from other nodes. This issue is particularly critical in WSNs, where transmitting at maximum power might dissolve the benefits in terms of energy saving coming from the topology control scheme adopted.

Our protocol, instead, takes a different approach: each node starts transmitting at low power and incrementally increases until K neighbors are contacted. To this end, each node periodically broadcasts a $beacon$ message containing its ID, the list of its current neighbors[1] $\mathcal{N}$ and the transmission range ρ (or the transmission power level) used. For instance, considering a generic node s, its beacon message β_s will have the following structure $\langle s, \mathcal{N}_s, \rho_s \rangle$.

When a node r overhears β_s, it saves the ID of the sender s together with its relative distance δ_{s-r}. This information can be estimated at the receiver by considering the ratio between the transmitting power at the sender site and the power computed at the receiver site, relying on the relation between the power attenuation and the distance. For wireless links, a signal transmitted with power P_t over a link with distance d gets attenuated and is received with power

[1]Two nodes are considered neighbors if and only if their relative distance is less than their transmission ranges.

$$P_r \propto \frac{P_t}{d^\alpha} \text{ with } \alpha \geq 2$$

where α is a constant that depends on the propagation medium[2], as illustrated in Section 3.

If δ_{s-r} is smaller than the transmission range used by r, ρ_r, s is included in $\mathcal{N}_r$. Otherwise, s cannot be considered neighbor because the link is not symmetric (s cannot hear beacons from r because $\rho_r < \delta_{s-r}$).

Topology Update During the topology update phase, each node computes how many neighbors it has collected during the discovery phase. If they are less than K, it increases the transmission range by a factor ρ_{inc} defined as protocol parameter.

Otherwise, if the number of neighbors is equal or greater than K, the transmission power is regulated to cover at most the distance of the K^{th} neighbor[3]. This way, a node is free to adaptively tune its range to cover exactly K neighbors.

2.2 Cooperation in the Topology Control Protocol

The protocol just described is indeed successful in maintaining the desired neighbor degree on each node and it enables saving a large amount of energy if compared with protocols without topology control [9]. Nevertheless, it still shows some drawbacks that may negatively impact the overall performance. This undesired behavior is depicted in Figure 1(a). There are three nodes, namely A, B and C, which have already reached the desired local connectivity ($K = 2$ in this case). However, there exists a further node D which instead needs to find K neighbors. Unfortunately, regardless how long its transmission range is, it is unable to connect to any node, since all other nodes have already K neighbors and are therefore unwilling to extend their range to include D. According to the protocol just described, D would end up in transmitting at maximum power, thus consuming high power and creating significant interference to other communications.

A common solution found in literature [12] consists in specifying low and high bounds for node degree ($K_{min} = K$ and $K_{max} > K$). Throughout the paper, we will refer to this solution as to MINMAX approach. MINMAX approach actually solves the problem but at the expense of an increased average number of neighbors (and, consequently, of the transmitting power used). Indeed, this mechanism does not distinguish among nodes that do need a connection (*critical nodes* in our terminology) and nodes which do not have this requirement. This is clarified in Figure 1(b), in which both A and B decide to increase their range to connect with D, whereas only one additional neighbor (beside C) is needed by D.

In [7], the authors propose a cooperative approach to overcome this issue. Whenever a node is below the desired local connectivity, it explicitly signals it to surrounding nodes through a special `help` packet. It then uses a `satisfy` packet to notify neighbors whenever a node is no longer critical. We will call this mechanism EXPLICIT . Even if a node has more bi-directional links than needed, it may increase its transmission power to help critical neighbors.

Beside the additional overhead required, this solution suffers from oscillation behavior. Indeed, in the example in Figure 1, A and B would first increase their range to reach D since it has sent a `help` packet. However, later D would send a `satisfy` packet and hence A and B would reduce their power, since they have more than $K = 2$ neighbors. This would lead to an oscillating behavior moving from situations depicted in Figure 1(a) and Figure 1(b). This issue was also confirmed by our experimental and simulation analysis, as detailed in Section 3.3

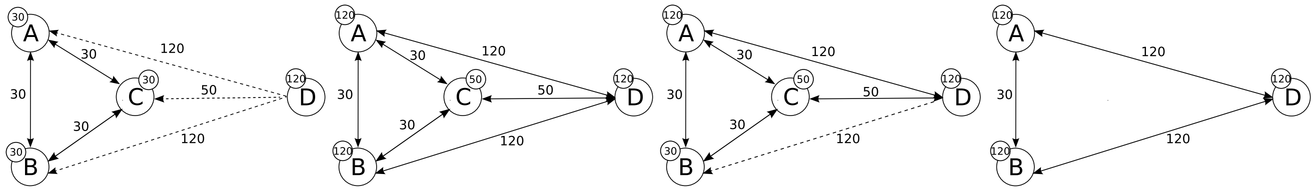
To avoid this behavior, our protocol leverages the neighbor list provided by each node in its beacons. During the discovery phase, when a node r receives a beacon from s, it computes the size of s's neighborhood ($\mathcal{N}_s$) and if it is lower than K, s is marked as *critical*. In the update phase, critical nodes are included as neighbors and transmission range is modified accordingly. Since cooperation is based on the content of the neighborhood lists, we will refer to our solution as to LIST BASED .

As pointed out above, it may occur that two or more nodes decide to accept a critical node as neighbor when, instead, for instance only one would suffice. Nevertheless, this does not represent an issue: during the next discovery phase the critical node will receive beacons from these nodes and will decide which one is more convenient (i.e. the closest). This way, in the subsequent discovery phase the critical node will broadcast its new neighbor list, containing the neighbors ordered from the closest to the farthest. Consequently, all other nodes can realize that they are not needed and can reduce their range. To avoid oscillatory behavior, we mark a node as critical for two subsequent discovery phases. In such a way, the critical node can receive the beacon from the cooperating node and can perform its choice (if there are more cooperating nodes than needed) in the next phase.

In our example, both A and B would decide to extend their range to reach D. However in the next phase D will beacon its neighbor list ($\mathcal{N}_D =< C, A, B >$), enabling other nodes to detect whether they could reduce their range. In this case, only B is allowed to decrease its power, as it

[2] α is typically around 2 in free space and around 4 for indoor environments.

[3] Here, of course, we do not account for the error introduced by estimation protocol. In a real deployment, we would add ε to the K^{th} node distance to tolerate it.



(a) (b) (c) (d)

Figure 1. Operation of the topology control algorithm: a simple example ($K = 2$). Values on edges represent distances while the number on each node shows the node's transmission range.

is the third D's neighbor whereas the critical threshold is $K = 2$. C and A, instead, cannot reduce their transmission ranges since they are essential to provide connectivity to D.

Our topology control protocol is also able to cope seamlessly with node failures. Indeed, during the discovery phase each node receives beacons from its neighbors and in the subsequent update phase it adjusts its range according to the data collected in the previous phase. With reference to our example, suppose that C crashes (e.g., because it runs out of battery) yielding to the situation sketched in Figure 1(d). B and D would realize that their number of neighbors is below the threshold $K = 2$. Hence, now B decides, cooperatively, to accept D because the latter is critical ($\mathcal{N}_D =< A >$) since it needs more neighbors. Interestingly, this operation is also beneficial for B as it has too few neighbors as well.

3 Experimental Evaluation

In order to test our protocol on the field, we set up experimental testbeds both in outdoor and indoor scenarios. In the following we highlight the implementation issues we have encountered, and we comment on the performance measures we have gathered through the testbed.

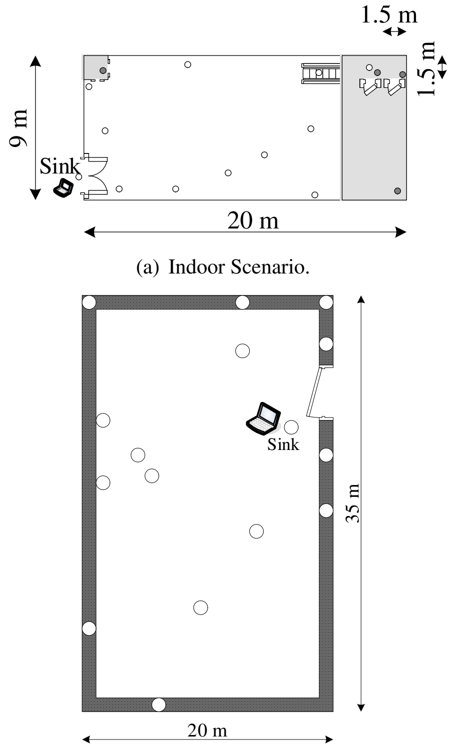
3.1 Testbed Setting

Each experiment adopts 16 XBow MicaZ [2] sensor nodes running topology control functions. MicaZ nodes are equipped with ChipCon CC2420 radio transceiver [1] which allows to choose among 8 transmission power levels in the interval [-25dBm, 0dBm] as specified in Table 1.

In the indoor experiments, the MicaZ nodes have been positioned in a warehouse building (20m x 9m) as described in Figure 2(a): 11 sensors are positioned at the ground floor, 4 at the mezzanine and 1 on the connecting stairs. Moreover, one of the ground-floor nodes is inside a separate room. All the nodes directly lay over materials and machines of the warehouse at different heights.

The outdoor experiments have been conducted on top of a flat roof of the same warehouse building. The experiment area is 35 meters long and 20 meters wide. MicaZ are deployed as shown in Figure 2(b), where nodes on the border of the flat roof (including sink) lay on a 1-meter-high parapet, while other sensors are positioned directly on the ground. We decided to place the other sensors on the ground to emulate unfavorable propagation conditions, thus stressing the topology control solutions.

Since it is very impractical to manually download data sensor by sensor at the end of each experiment, we implemented an automatic procedure to collect at a sink node all the data stored during the experiment by all the other



(a) Indoor Scenario.

(b) Outdoor Scenario.

Figure 2. Network topologies of the testbed.

Power Level ID	Emitted Power [mW]
7	52.2
6	49.5
5	45.6
4	41.7
3	37.5
2	33.6
1	29.7
0	25.5

Table 1. CC2420 Transmission Power Levels.

	Code Size	RAM Footprint
MINMAX	11.2Kbyte	488byte
List based	11.9Kbyte	744byte
Explicit	11.4	488byte

Table 2. Binary size and RAM footprint of the topology control protocols.

sensors. To this end, one MicaZ node in each scenario acts as information sink and is directly connected through a MIB510 Serial Gateway to a PC running Linux distribution Debian with 2.6.18 kernel version. Each sensor collects and stores periodical samples of information during the experiment including the list of perceived neighbors (and the corresponding power levels) and the current transmissions power. Upon completing the experiment, each sensor searches for a path to the sink using the MintRoute routing protocol [16] and then sends to the sink all these information samples.

The sink sensor passes such information to the PC which runs a Java filter, returning the overall performance measures used to evaluate the topology control solutions. Moreover, the Java tool implements also a query mechanism based on a diffusion protocol to force the sink node to request missing information that may get lost during the collection phase.

3.2 TC Implementation Issues

We implemented our protocol and the other approaches described in Section 2 in TinyOS ver.1.x and deployed them onto MicaZ sensors. The corresponding binary sizes (including radio stack, UART, timers and led components) and memory footprints are reported in Table 2.

During our experiments, we have observed that the stability of any topology control solution is highly affected by the variability of the wireless link quality. In fact, if link quality varies very often, the perceived number of neighbors is scarcely stable, and the algorithm itself is driven to frequent changes in the transmission power. Therefore, it is of utmost importance to introduce techniques to stabilize the number of neighbors filtering out the fluctuations of the wireless channel. First, we need to define metrics to measure the "quality" of a given link. It is shown in [14] that the Received Signal Strength Intensity (RSSI) of wireless links among MicaZ sensors geared with ChipCon 2420 transceivers provides a consistent estimate of the Packet Reception Rate (PRR). Namely, the authors show that if the RSSI is above -87dBm, the PRR is above 85%. Below that threshold a gray zone does exist, where the PRR may be extremely variable.

We inserted a control on the RSSI of the received transmissions according to which the information contained in a received beacon message is considered in the topology control procedures only if the RSSI is above the -87dBm threshold. Moreover, to achieve long-life link stability, we have decoupled in time the topology update and the beaconing phase allowing a topology update every twelve beaconing intervals. The information contained in a beacon is stored and used in the topology control phase only if at least x (parameter) out of the 12 beacons replicas have been received correctly, i.e., with an RSSI above -87dBm. Beacons contain the transmitter ID, the list of neighboring sensors and the transmission power level, which allows each sensor to locally create a list of neighbors with the corresponding transmission power levels to be used to reach them. Information on the criticality of each neighbors is also stored.

The results obtained from the testbed are validated against TOSSIM simulations in the very same indoor and outdoor network scenarios. We adopted the same empirical approach proposed in [16] to model the link behavior in simulations. Each sensor in the testbed transmits 200 packets to any other sensor which measures the packet reception rate. Such procedure is repeated for all the 8 transmission power levels and for all the 16 sensors in the network. Thus, for each sensor, we obtain measurements at different receivers when using different transmission power levels. We are therefore able to associate to each directed pair of sensors and for each power level a packet reception probability, which is then used in the TOSSIM simulations. Table 3 reports the values of packet reception probabilities for the links of node 1 towards all the sensors in the indoor environment.

3.3 Performance Evaluation

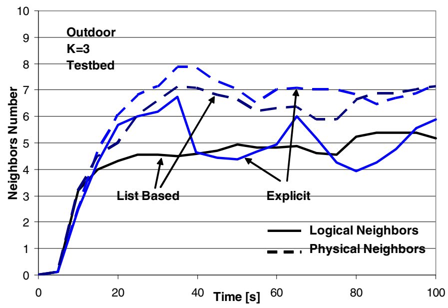
Every test is composed of two phases: in the first one, nodes are switched on from scratch and run the specific topology control algorithms for a period of 100s. After that, they move onto the second phase devoted to data collection and data elaboration as described in Section 3.1.

We collected the following performance metrics:

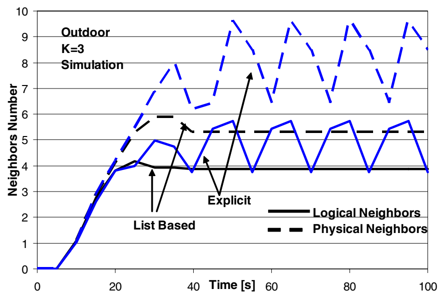
- *local connectivity*: we measure the local connectivity of any sensor in terms of *logical* neighbors, i.e. those neighbors connected through symmetric links, and *physical* neighbors, i.e. all the nodes reached by

		Transmitting Sensor													
		0	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
TX Power Level	**0**	0.9	0.9	0.18	0	0	0	0	0	0	0	0	0	0	0	0
	1	1	1	0.99	1	0.85	0.98	0.15	0.31	0	0	0	0	0	0	0
	2	1	1	1	1	1	1	0.75	1	0.17	0.75	0	0	0	0	0
	3	1	1	1	1	1	1	1	1	0.3	1	0.32	0	0	0	0
	4	1	1	1	1	1	1	1	1	0.35	1	1	0	0	0	0
	5	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
	6	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
	7	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0

Table 3. Empirical measures of PRR at sensor 1 from different transmitters, using different transmission power levels.

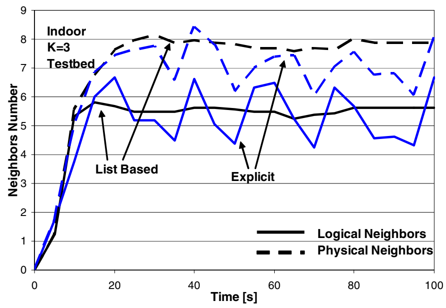


(a) Testbed.

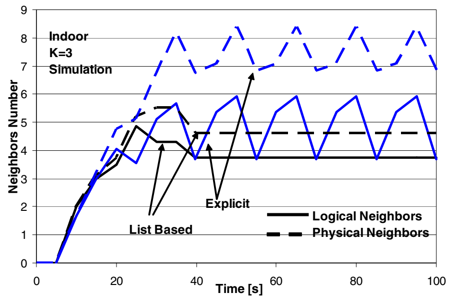


(b) TOSSIM Simulations.

Figure 3. Average number of physical and logical neighbors (Outdoor).



(a) Testbed.



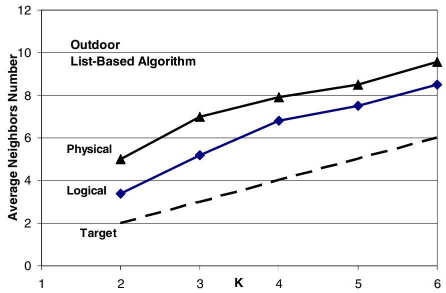
(b) TOSSIM Simulations.

Figure 4. Average number of physical and logical neighbors (Indoor).

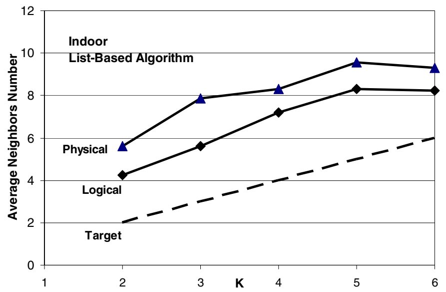
sensor's beacons (through symmetric and asymmetric links);

- *network connectivity*: we measure the network connectivity as the ratio between the number of vertexes of the largest connected sub-graph and the total number of sensors. When such ratio is 1 all the sensors constitute a fully connected graph;
- average *transmission power* and transmission power distribution.

The first parameter we analyze is the local connectivity provided by the cooperative LIST BASED and EXPLICIT topology control solutions. Figures 3 and 4 depict the average number of logical and physical neighbors in the outdoor and indoor network scenarios, respectively. Figure 3(b) and 4(b) give the simulation results obtained through TOSSIM in the very same testbed environments (outdoor and indoor). Both testbed and simulation results confirm the oscillatory behavior of the cooperative EXPLICIT approach, as expected and described in Section 2.2. We further observe that even if TOSSIM simulations provide the same behavior as testbed measurements, a difference in the absolute numbers holds, due to the non-ideal propaga-

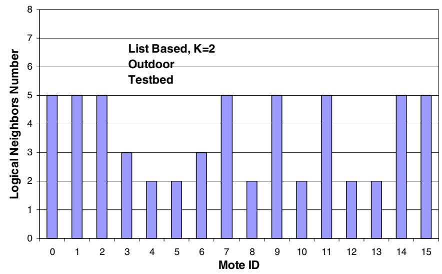


(a) Outdoor.

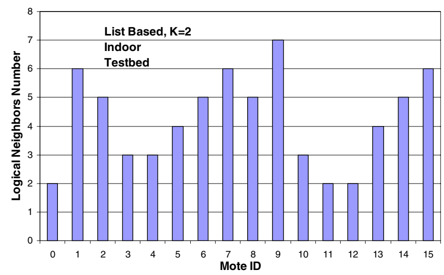


(b) Indoor.

Figure 5. Physical and logical neighbors against K (testbed).



(a) Outdoor.



(b) Indoor.

Figure 6. Number of logical neighbors per sensor in case K=2 (testbed).

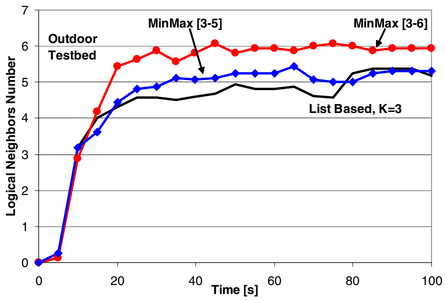
tion conditions of the testbed compared to the static empirical propagation model used in the simulations.

Notably, in both scenarios the list-based algorithm provides a number of logical neighbors slightly higher than the target parameter K. In fact, Figure 5 reports the measured number of logical and physical neighbors against the target value K for the LIST BASED topology control approach in the testbed. Such difference in excess is due to the "price of cooperation", that is, the fact that the cooperative approach forces a subset of nodes to increase their transmission power to help critical neighbors (see Section 2.2). Figure 6 zooms on this effect by reporting the number of logical neighbors per sensor in case the LIST BASED approach is used with K=2. As clear from the two figures, the aforementioned "price of cooperation" leads some sensors to have a number of logical neighbors which is higher than the target value (K=2) in both indoor and outdoor testbed scenarios.

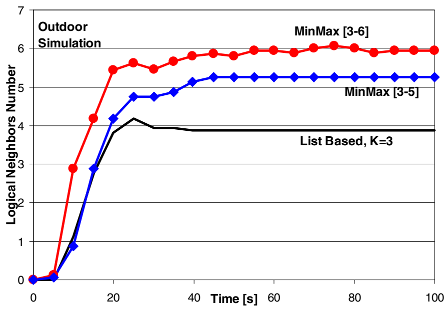
One might argue that non-cooperative approaches based on an interval $[K_{min}, K_{max}]$ of feasible degrees (hereafter referred to as MINMAX protocol) may make cooperation useless. To address this remark, we tested this strategy in the testbed and compared its performance with the LIST BASED cooperative approach. In our experiments, we have set $K_{min} = K = 3$ whereas we tested two values for K_{max} (5 and 6) in the MINMAX case.

Figures 7 and 8 compare the average number of logical neighbors in outdoor and indoor scenarios for the three cases: LIST BASED cooperative, MINMAX $K_{max} = 5$ and MINMAX $K_{max} = 6$. Results obtained through TOSSIM simulation are also reported for the sake of comparison. We observe that the MINMAX protocol leads the nodes to have an average number of neighbors often close to K_{max}. The reason stems from the fact that a node cannot distinguish between critical and non-critical nodes and, hence, it always accepts neighbors until the threshold K_{max} is met. Moreover, in some cases, MINMAX approach provides an average number of neighbors which is even slightly higher than K_{max} (e.g., MINMAX 3-5 in Figure 8(a)); this counterintuitive behavior has two causes: the specific testbed topology and the quantization of the transmission power levels. In fact, it may happen that one sensor tuning its power level to reach its K_{max}-th neighbor, reaches also other farther away neighbors. In other words, the granularity with which neighbors can be added to the neighbors'list may be coarse. This granularity effect is visible in Figure 6, where the maximum number of logical neighbors per sensors is clearly higher in the indoor testbed scenario (7 versus 5), which features an higher density of sensor nodes.

The overall effect is that nodes consume much more power because the more neighbors they have, the higher their transmission power is, as readily confirmed by Figure

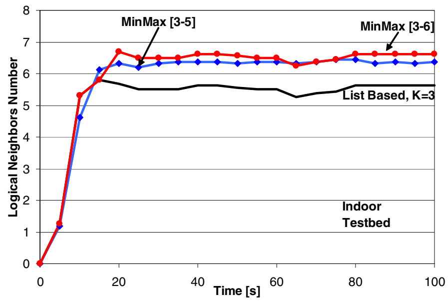


(a) Testbed.

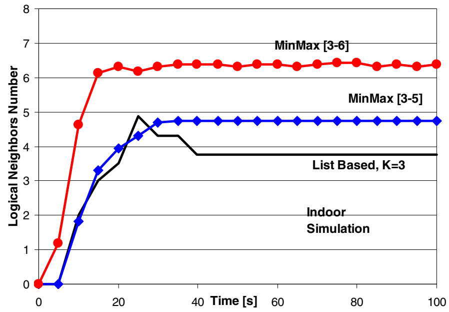


(b) TOSSIM Simulations.

Figure 7. Average number of logical neighbors (Outdoor).



(a) Testbed.



(b) TOSSIM Simulations.

Figure 8. Average number of logical neighbors (Indoor).

	Time to Connectivity [s]		
	List-based	MINMAX $[3-5]$	MINMAX $[3-6]$
Indoor	25	20	20
Outdoor	15	10	10

Table 4. Time to reach full multi-hop connectivity.

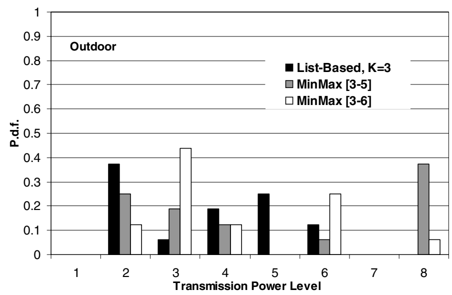
9 which reports the average transmission power over time and the *P.d.f.* of the transmission power levels at the end of the testbed experiments (t = 100s).

Besides local connectivity, it is worth evaluating the connectivity properties of the overall network topology. To this end, we have computed in post processing the percentage of sensors in the largest connected subgraph by automatically solving max flow problems on the data collected at the sink. We have observed that the network becomes fully connected in all the cases (indoor and outdoor) and for all the algorithms. The time-to-full-connectivity is reported in Table 4. The MINMAX approach allows to have a slightly lower time to connectivity, with the drawback of consuming more transmission power, as shown beforehand.

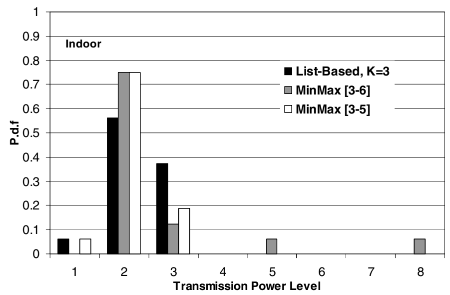
4 Simulation Analysis

The experimental results presented in the previous section have been obtained on real life testbeds featuring a small number of sensors. It is worth analyzing whether the performance characteristics of the topology control solutions highlighted so far still hold true for medium/large network scales. Since a real large-scale testbed was unfeasible, we resort to simulation in TOSSIM. The simulation results reported hereafter have been obtained on a square network topology (350m x 350m), where 200 sensors are randomly scattered. Each result shown hereafter has been obtained averaging over 100 realization of the sensors' distribution. The measured confidence index for all collected statistics is better than 5% in 98% of all cases.

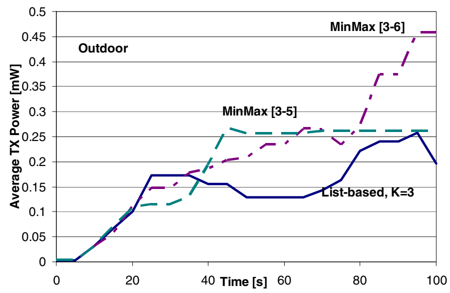
Moreover, we have resorted to an empirical approach to characterize the wireless links similar to the one described in Section 3. In details, we have measured the Packet Reception Probability (PRR) of a single outdoor wireless link of increasing length, when adopting different transmission power levels. The measured PRR has been used to characterize the packet reception procedure in the TOSSIM simulations, depending on the simulated distance between sender and receiver, and the transmission power.



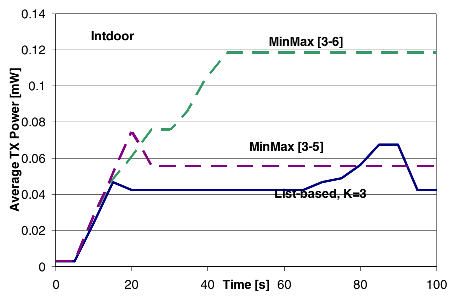
(a) P.d.f. (outdoor).



(b) P.d.f. (indoor).

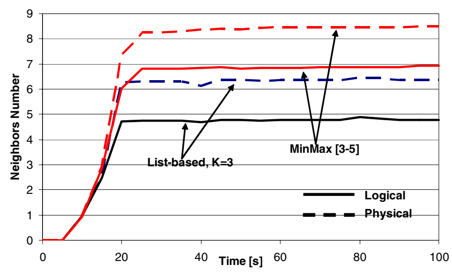


(c) Average (outdoor).

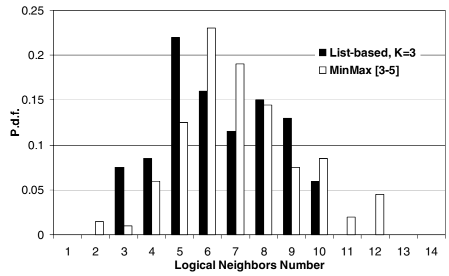


(d) Average (indoor).

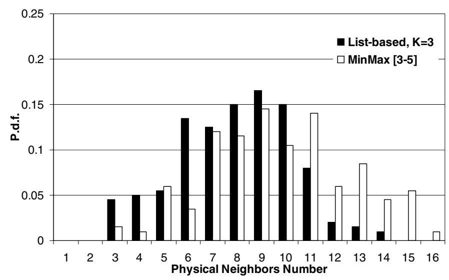
Figure 9. Transmission power (testbed).



(a) Average number of logical neighbors.



(b) P.d.f. Logical Neighbors.



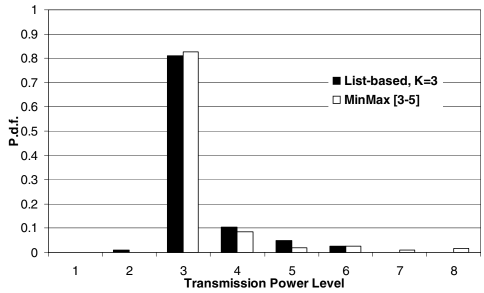
(c) P.d.f. Physical Neighbors.

Figure 10. Local connectivity (TOSSIM Simulation).

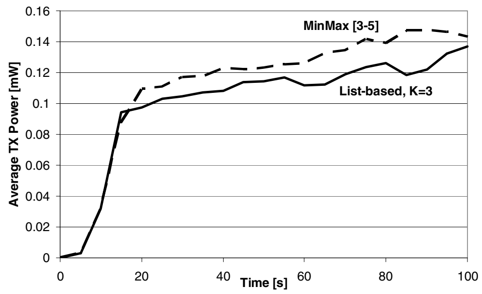
Again, we start off by analyzing the local connectivity provided by different topology control solutions. Figure 10 reports the the average number of logical and physical neighbors and their *P.d.f.* in the reference network scenario in case of the LIST BASED and MINMAX topology control algorithms. As already shown in the small-scale scenario (testbed and simulations), the MINMAX protocol provides higher average number of logical and physical neighbors with respect to the LIST BASED approach. As a consequence, the amount of consumed power is higher power as confirmed by Figures 11 which reports the average transmission power over time and the *P.d.f.* of the transmission power levels at the end of the simulation (t= 100s).

Moreover, in the MINMAX case, a small fraction of nodes still have less than $K_{min} = 3$ logical neighbors (see Figure 10(b)). This yields two significant consequences. Firstly, these nodes will transmit at maximum power[4] in the attempt to find other neighbors, thus dramatically increasing

[4]This explains why in Figure 11(a) there are a few nodes transmitting at the highest power level.



(a) P.d.f.



(b) Average.

Figure 11. Transmission Power (TOSSIM Simulation).

their power consumption and introducing high interference. Furthermore, the fact that not all the nodes match the local connectivity constraint leads to failures in the overall network connectivity too. Indeed, with the MINMAX approach and K_{max}=5 the network remains not connected (connectivity degree is 0.97), whereas the LIST BASED approach provides 100% connectivity in the simulation time.

5 Concluding Remarks

In this paper, we have described a lightweight and cooperative solution to the problem of controlling the local connectivity in wireless sensor networks. We set up and deployed real-life testbed of small-scale indoor and outdoor wireless sensor network to test the performance of the proposed solution against other common approaches.

Finally, we complemented the testbed analysis in small-scale environments with TOSSIM simulations both in small-scale and in large-scale wireless sensors networks. In all the tested scenarios, experimental and simulation results shows that our solution outperforms the other approaches, providing steady network connectivity while reducing the overall power consumption.

References

[1] http://www.chipcon.com.

[2] http://www.xbow.com.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Comm. Mag.*, 40(8):102–114, August 2002.

[4] D. M. Blough, M. Leoncini, G. Resta, and P. Santi. The k-neigh protocol for symmetric topology control in ad hoc networks. *In Proc. of ACM MOBIHOC*, pages 141–152, 2003.

[5] M. Cardei, S. Yang, and J. Wu. Algorithms for fault-tolerant topology in heterogeneous wireless sensor networks. *IEEE Trans. on Parallel and Distributed Systems*, March 2008.

[6] S. Conner, J. Chhabra, M. Yarvis, and L. Krishnamurthy. Experimental evaluation of synchronization and topology control for in-building sensor network applications. *In Proc. of ACM International Workshop on Wireless Sensor Networks and Applications*, pages 38–49, 2003.

[7] M. Gerharz, C. de Waal, P. Martini, and P. James. A co-operative nearest neighbours topology control algorithm for wireless ad hoc networks. *In Proc. of IEEE ICCCN*, pages 412–417, October 2003.

[8] L. Hu. Topology control for multihop packet radio networks. *IEEE Trans. on Comm.*, 41(10):102–114, October 1993.

[9] M. Kubisch, H. Karl, A. Wolisz, L. Zhong, and J. Rabaey. Distributed algorithms for transmission power control in wireless sensor networks. *In Proc. of IEEE WCNC*, 1:558–563, March 2003.

[10] S. Lin, J. Zhang, G. Zhou, L. Gu, J. Stankovic, and T. He. Atpc: adaptive transmission power control for wireless sensor networks. *In Proc. of the ACM Conference On Embedded Networked Sensor Systems*, pages 223–236, 2006.

[11] A. Muqattash and M. Krunz. Powmac: A single-channel power-control protocol for throughput enhancement in wireless ad hoc networks. *Jour. on Sel. Areas in Comm.*, 23(5):1067–1084, May 2005.

[12] R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. *In Proc. of IEEE INFOCOM*, 2:404–413, 2000.

[13] C. C. Shen, C. Srisathapornphat, R. Liu, Z. Huang, and C. Jaikaeo. Cltc: a cluster-based topology control framework for ad hoc networks. *IEEE Trans. on Mob. Comp.*, 3(1):18–32, February 2004.

[14] K. Srinivasan and P. Levis. Rssi is under-appreciated. *In Proc. of the Fourth Workshop on Embedded Networked Sensors (EmNets)*, May 2006.

[15] R. Wattenhofer, L. Li, P. Bahl, and Y. M. Wang. Distributed topology control for power efficient operation in multihop wireless ad hoc networks. *In Proc. of IEEE INFOCOM*, 3:1388–1397, April 2001.

[16] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. *In Proc. of ACM Sensys*, pages 14–27, 2003.

[17] F. Xue and P. R. Kumar. The number of neighbors needed for connectivity of wireless networks. *Wireless Networks*, 10(2):102–114, March 2004.

[18] C. Yu, K. Shin, and B. Lee. Power-stepped protocol: enhancing spatial utilization in a clustered mobile ad hoc network. *Jour. on Sel. Areas in Comm.*, 22(7):1322–1334, September 2004.