Interference Arises at the Receiver

Martin Fussen, Roger Wattenhofer, Aaron Zollinger
{mafussen@student, wattenhofer@tik.ee, zollinger@tik.ee}.ethz.ch
Computer Engineering and Networks Laboratory
ETH Zurich, Switzerland

***Abstract*— Energy consumption in general and interference in particular being among the most critical issues in wireless networks, this paper introduces an explicit definition of interference, based on the number of other nodes by which a given network node can be disturbed. With this definition we show that there exist instances of sensor networks in which no topology control algorithm—aiming at interference reduction by having nodes restrict their transmission power levels—can construct a valid data gathering network with interference less than logarithmic in the number of network nodes n. In a second part of the paper we introduce the Nearest Component Connector (NCC) algorithm, which asymptotically matches this lower bound, guaranteeing to build a valid topology with interference in $O(\log n)$ in any given sensor network. Finally the paper compares NCC to other previously proposed data gathering structures in average-case networks.**

I. Introduction

Among the most critical resources in wireless networks with autonomous nodes is energy. One of the foremost approaches to reducing energy consumption consists in minimizing interference between the network nodes and consequently in reducing the number of message collisions and hence required retransmissions. The concept of *topology control* confines interference by having the network nodes reduce their transmission power levels and drop long-range connections in a coordinated way. At the same time transmission power reduction has to occur in a controlled manner in order to preserve connectivity of the network.[1]

Most of the previous work maintains to solve the interference issue in wireless networks implicitly by constructing sparse topologies or topologies with constant-bounded node degrees. Such an implicit notion of interference can however lead to topology control algorithms that fail to reduce interference since message transmission can affect nodes even if they are not direct neighbors of the sending node in the resulting topology graph [1]. Besides demonstrating this weakness of implicit interference models, [1] introduces an explicit definition of interference, based on the number of nodes potentially disturbed by communication over a link.

In contrast we assume in this paper a receiver-centric perspective. Particularly, we formulate an interference definition at the heart of which lies the question by how many other nodes a given network node can be disturbed. Compared to the sender-centric interference definition proposed in [1], the definition of interference presented in this paper reflects intuition more closely in the sense that interference is considered at the receiver, where message collisions prevent proper reception. Informally, our interference definition can also be considered to correspond to the effort required to avoid collisions, be it by means of time division multiplexing—assigning transmission time slots such that no two messages collide at a receiving node—, by means of frequency division multiplexing—having messages sent in different assigned frequency bands—, or by means of code division multiplexing—where small interference allows for reduced coding overhead.

In this paper we consider interference in sensor networks. A sensor network consists of sensors deployed in a given region with the task of sensing a certain physical value (such as temperature, humidity, brightness, or motion). The sensors are equipped with radio devices and—in the popular monitoring scenario model—periodically transfer the sensed data to a designated data sink node. To allow all data to be gathered at the sink, a topology control algorithm therefore constructs a *sink tree*, a directed tree with all arcs (directed edges)—modeling unidirectional communication links—pointing towards the sink node. In the context of interference reduction, the task of the topology control algorithm is to find such a sink tree with least possible interference. Thereby we account for the fact that in the monitoring scenario communication from the sink to the sensors occurs rarely and can therefore be neglected with respect to interference.

Assuming a worst-case perspective we show in the paper that there are network instances in which any topology control algorithm will construct a resulting network with interference at least $\log n - 1$. We furthermore propose the *Nearest Component Connector (NCC)* algorithm, which provably produces at most $O(\log n)$ interference in any network in polynomial time. In this sense the NCC algorithm is asymptotically optimal. In a second part of the paper we compare the NCC algorithm with previously proposed structures in average networks. On the one hand we thereby show that—besides being asymptotically worst-case optimal—NCC also in the average case produces interference results comparable with previously proposed structures. On the other hand a minimum-spanning-tree-based structure not originally designed to reduce interference interestingly appears to outperform NCC in average-case networks, while it cannot guarantee to produce low interference in worst-case examples.

The paper is organized as follows: After discussing related work, we introduce our interference model and a formulation of the considered interference minimization problem in Section III. While Section IV shows that there exist network instances on which any topology control algorithm will produce interference at least $\log n - 1$, the subsequent section presents

[1]Also clustering and the construction of dominating node sets is sometimes considered topology control. In this paper we however only study topology control based on transmission power reduction.

the NCC algorithm and proves that it matches this lower bound. Section VI discusses interference generated by NCC and other algorithms in average-case networks. Section VII concludes the paper.

II. Related Work

The issue of energy efficiency in sensor networks [2], [3], [4]—particularly extending network lifetime—has been mainly studied in the context of optimal sensor placement and energy-efficient routing. Recently also the fact that certain types of sensed data allow for aggregation at sensor nodes [5] and the existence of redundancy in acquired information [6], [7]—for instance correlation between sensed data depending on the distance between sensors—has been considered.

The concept of topology control has been studied in the broader context of ad-hoc networks—wireless networks whose application is not confined or targeted to data acquisition and gathering, as is the case for sensor networks—for two decades [8], [9], [10], [11], [12], [13], [14], [15], [16]. Although topology control has sometimes been considered the task of generally constructing topologies with certain desired properties (such as network connectivity, planarity, sparseness or locality of construction), interference reduction has often been regarded as one of the main goals of topology control. However, most of the proposed topology control algorithms are stated to produce low interference implicitly by constructing sparse networks or networks with bounded node degree. As shown in [1], such implicit interference reduction can fail to effectively achieve its goal.

A notable exception to this is [17], which defines an explicit concept of interference between edges and shows—based on a time-step routing model—that there exist inevitable trade-offs between congestion, energy consumption and dilation. While this interference definition is based on the current network traffic, [1] proposes an explicit definition of interference that is independent of network traffic. This interference definition—adopted and further studied in [18]—is based on the question how many nodes are affected by communication over a given link. In contrast the interference model introduced in this paper considers interference at the intended receiver of a message since this is where message collisions actually have their negative effect.

III. Model and Notation

In this section we describe our model of a sensor network and formally define interference and interference minimization in the context of our model.

Our model of a sensor network is a directed graph $G(V,E)$ where nodes $\{v_1,\ldots,v_n\}$ placed in the plane represent the set of sensors including the sink. Communication links between sensors are modeled as arcs. We assume that the transmission power of each node can be adjusted. A higher transmission power allows a node to send messages over a longer distance. We assume that the covered area of a sending node v_i is a disk with v_i in its center. Furthermore we presume that a node can reach another node only if it is at most 1 distance unit away. In

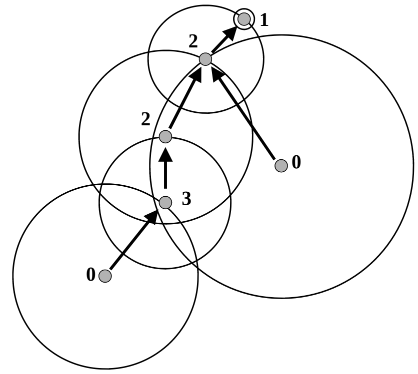

Fig. 1. A sink tree with 6 nodes. The uppermost node is the sink node. Each node is labeled with its interference value. The interference of the whole network is 3.

other words the graph consisting of all eligible arcs if all nodes set their transmission power to the maximum possible values corresponds to the so-called Unit Disk Graph constructed given the node set V, where an edge (or two symmetric arcs) exists if and only if the distance between two nodes is at most 1. Finally we only consider *connectable* graphs, which means that—with all transmission radii set to their maximum values—a path from any node to any other node in the network is constructible or—more technically—the Unit Disk Graph given the node set V consists of one connected component.

If we want to minimize interference in sensor networks, we have to look at topologies in which each node sends its data to at most one other node and a valid graph contains a path from every sensor to the sink, which results in a tree with the sink as its root and all arcs pointing towards the root. We call such a tree a *sink tree*. Figure 1 shows a sample sink tree with 6 nodes.

Definition 1: Given a set of nodes V and a sink s, a sink tree is a tree spanning V with all arcs pointing towards s.

We use an explicit model of interference. We explicitly count the number of nodes potentially disturbing the reception of a message. This definition best reflects the fact that interference is a problem occurring at the receiver. Minimizing the interference at each possible receiver (each node in the network) reduces the number of potential message collisions in the network and therefore lowers the amount of required retransmissions. This saves energy and allows for a longer lifetime of sensors equipped with batteries.

The interference value of a single node is the number of transmission circles by which the node is covered.

Definition 2: The interference value of a single node v is defined as

$$I(v) := |\{u | u \neq v \wedge v \in D(u, r_u)\}|$$

where $D(u, r_u)$ stands for the transmission circle with node u in its center and radius r_u.

As the interference of a whole network we use the maximum

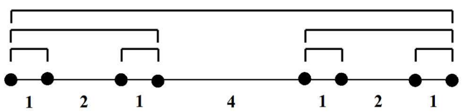

Fig. 2. A recursive arrangement of 16 nodes on a horizontal line. The labels indicate distances without normalization by the factor $1/(4 \cdot 3^{s-2})$.

of all interference values in the graph (see Figure 1).[2]

Definition 3: The interference of a Graph G(V,E) is defined as

$$I(G) := \max_{v \in V} I(v).$$

The problem we study in this paper consists in finding a sink tree with least possible interference for a given sensor network.

Definition 4: The Minimum Interference Sink Tree (MIST) problem is defined as the problem of finding a sink tree for a given node set with minimal interference.

In the remainder of the paper we consider topology control algorithms with the goal of solving the MIST problem.

IV. A LOWER BOUND

In this section we show that n nodes in a sensor network can be arranged in a way that no possible algorithm can construct a sink tree with interference less than $\log(n) - 1$. The existence of such examples constitutes a lower bound with respect to interference.

Theorem 4.1: There exist sensor networks with nodes arranged in a way that no algorithm can construct a sink tree with interference less than $\log(n) - 1$.

Proof: To prove this Theorem we present an arrangement of $n = 2^s$ nodes which cannot be connected to the given sink in the described way with interference less than $\log(n) - 1$.

The nodes are arranged on a horizontal line. Figure 2 shows the arrangement. The first $k = 4$ nodes v_1, v_2, v_3, and v_4, are positioned at coordinates 0, $1 \cdot \nu$, $3 \cdot \nu$, and $4 \cdot \nu$, where $\nu = 1/(4 \cdot 3^{s-2})$ is a normalization factor. Then a copy of the already positioned nodes is placed in distance $d = \overline{v_1 v_k}$ to the right of node v_k. This construction is recursively repeated until 2^s nodes are placed on the horizontal line. Note that due to normalization of the node positions every node can reach any other node in the network.

After the execution of any possible algorithm there must exist a directed path from each node in the set to the global sink. Let G_1 be the node group $\{v_1, \ldots, v_{2^{s-1}}\}$ and $G_2 := \{v_{2^{s-1}+1}, \ldots, v_{2^s}\}$. If we assume without loss of generality that the node group G_2 contains the global sink, the result of an algorithm has to contain an arc from G_1 to G_2. Because of the special arrangement of the nodes in our example, the gap between G_1 and G_2 has length equal to the (Euclidean) diameter of the two groups. The arc between G_1 and G_2 cannot be shorter than the gap and therefore interferes with all nodes but one in G_1. Figure 3 illustrates the idea of the proof.

If, in a next step, we look into G_1, there are 2^{s-1} nodes partitioned into two subgroups $G_{1.1}$ and $G_{1.2}$. Assuming, again without loss of generality, that the above arc from G_1 to G_2 originates in $G_{1.2}$, we can observe that there has to exist an arc leading out from $G_{1.1}$, which—bridging $G_{1.1}$'s adjacent gap—interferes with all nodes in $G_{1.1}$ (except for the node at which this arc originates). The existence of such an arc is a consequence to the required directed path from each node to the sink. The same argument recursively holds for all node levels in the arrangement.

This all together proves that the interference value of at least one node is incremented in all steps and because the node set is of size 2^s we get a maximum interference not smaller than $s - 1$ or $\log(n) - 1$. ∎

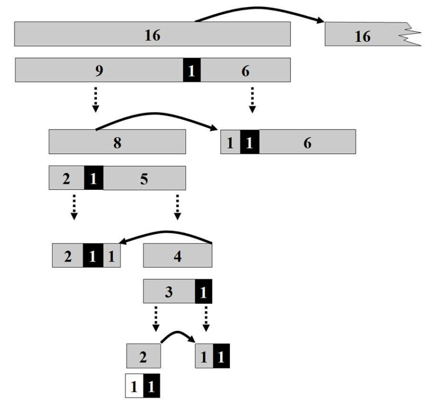

Fig. 3. Illustration of the proof of Theorem 4.1. The numbers stand for the number of nodes in each group. The dark fields represent the unaffected nodes in the last step and the hollow node (bottom) is the one whose interference value was incremented in each step.

V. NCC ALGORITHM

In this section we present the Nearest Component Connector algorithm *(NCC)* as described in detail in Algorithms 1 and 2.

The general idea of this algorithm is to connect components to their nearest neighbors. This is done in several rounds and leads to a sink tree. A component can be a single node or a group of previously connected nodes. When the algorithm starts, each node in the given sensor network forms a component of its own. First the predefined global sink is treated exactly as a normal node. Whenever two or more components are connected in one round, they form a single component in the following round of NCC. If we have a look at an arbitrary component during the execution of the algorithm we observe that this component has exactly one node all other component members have a directed path to. This means that there is one node which gathers all sensed data of the component. We call this special node the *local sink* of its component.

Whenever a new arc is established during the execution of NCC, it goes from a local sink of a component C to the nearest node not in C. However, due to the fact that all nodes have

[2] It can be argued similarly that the interference of a whole network can be defined as the average of the node interference values. This definition is not considered in this paper. Note that with this alternative definition of interference, the problem of finding a valid data-gathering structure with minimum interference can be solved optimally constructing a Minimum Directed Spanning Tree with arc weights corresponding to the number of nodes covered by each edge.

Algorithm 1 Nearest Component Connector Algorithm NCC

Input: V: a set of nodes placed in the plane
$s_g \in V$: a predefined global sink

$G := (V, E := \emptyset)$
$lsinks := V$ // set of local sinks
while $|lsinks| > 1$ **do**
  for all $s \in lsinks$ **do**
    $E' := \emptyset$
    $C :=$ component containing s
    if s cannot reach any node outside C **then**
      $s' :=$ nearest node to s (hop metric) capable of reaching a node outside C
      $movesink(G, s, s')$
      $s := s'$
    end if
    $E' := E' \cup \{e\}$, where e is the arc from s to its nearest neighbor (Euclidean distance) outside C
  end for
  if $G' := (V, E \cup E')$ contains cycles **then**
    remove one of the arcs in each cycle from E'
  end if
  $G := G'$
  $lsinks :=$ sinks in G // sinks are nodes having no outgoing arc
end while
$s :=$ only remaining sink in $lsinks$
if $s \neq s_g$ **then**
  $movesink(G, s, s_g)$
end if

Output: G

Algorithm 2 Procedure $movesink(G, s_1, s_2)$

Input: Graph $G = (V, E)$
s_1: a local sink in G
s_2: a node in the same component as s_1

sp := shortest path from s_1 to s_2 according to the hop metric
remove all arcs originating at nodes on sp (including s_2) from E
add arcs on sp to E

maximum transmission range 1, it is possible that the current sink s of a component C cannot connect to any node outside C. In this case another node s' is designated to become the new sink of C, particularly the nearest node to s (with respect to the number of hops) capable of reaching any node outside C. This happens by removing all arcs originating at nodes on the shortest path $sp(s, s')$ from s to s' and subsequently adding the arcs along $sp(s, s')$ (cf. Algorithm 2). Note that every component contains at least one node capable of reaching another node outside its component since we only consider connectable networks.

If a round—connecting every sink to its nearest neighbor outside its component—produces a cycle, it is broken by removing one of its arcs at the end of the round. This guarantees the construction of a valid sink tree topology. After the last round of NCC, however, the root of the resulting tree is not necessarily the global sink. In this case the root of the resulting tree is moved to the global sink again by means of the $movesink$ procedure (Algorithm 2).

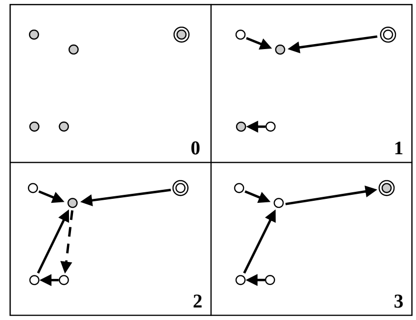

Fig. 4. A sample execution of NCC on a given set of 5 nodes. Situation 0 shows the given nodes and the predefined sink (top right node) In each of the following two rounds every local sink connects to the nearest node not in its own component. In round 2 a cycle is produced. It is broken at the end of the round by removing one of the involved arcs (dashed arrow). After the last round (Situation 3) the arc originating from the global sink is removed and an arc is added from the only remaining local sink to the predefined global sink. For clarity of representation the node distances are assumed to be sufficiently small such that execution of the $movesink$ procedure is not required.

Figure 4 shows a sample execution of the NCC algorithm.

We will now prove that the presented NCC algorithm constructs a valid sink tree topology for a given sensor network consisting of n nodes with an interference value in $\mathrm{O}(\log n)$. We will also see that the execution of NCC takes polynomial time only.

Theorem 5.1: The NCC Algorithm constructs a sink tree on a given Graph $G = (V, E)$ with $|V| = n$ with an interference value in $\mathrm{O}(\log n)$ in polynomial time.

Proof: This proof has three parts. In the first one we show that NCC does not need more than $\log n$ rounds (while-loop iterations) to build the sink tree. In the second part we show that in each of these rounds the interference value of a node will not be incremented by more than a constant value. In part three we show that NCC terminates in time polynomial in the total number of nodes.

To show the first part we use the fact that in each round a local sink s either establishes an arc to the nearest node of another component or that another component establishes an arc to the component s is part of. The two or more connected components together form one component in the next round. Therefore the number of components in round i is at most half the number of the components in round $i-1$. This implies that after at most $\log n$ rounds only one component is left and the algorithm terminates.

In the second part of the proof we show that each round increases the interference value at most by a constant. First, the $movesink(G, s, s')$ procedure only increases the interference values of nodes in C, the component of s and s', since none of the nodes on the shortest path from s to s' can reach any node

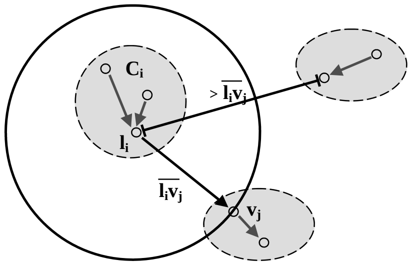

Fig. 5. An illustration of the second part of the proof of Theorem 5.1. An arc from the local sink l_i to v_j only interferes with nodes in l_i's component and v_j (and nodes in other components only if they are exactly at distance $\overline{l_i v_j}$).

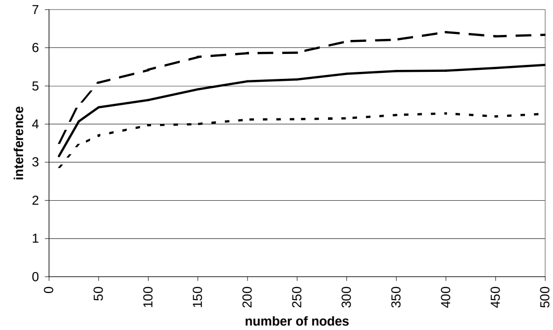

Fig. 6. Simulation results for the Shortest Path Tree algorithm (dashed), the Nearest Component Connector algorithm (solid) and the Minimum Spanning Tree algorithm (dotted) in a range from 10 to 500 nodes.

outside C; furthermore introduction of the arcs on the shortest path from s to s' increases the interference value by at most 3. (A node v suffering interference increase of at least 4 would contradict the fact that only arcs on a shortest path are added to the graph; the shortest path could be shortcut via v, exploiting that in the Unit Disk Graph model all arcs of length up to one unit are eligible.) Second, once the sink of a component C can reach another node outside C, we use the fact that each sink connects to the *nearest* node in a component different from its own, as illustrated in Figure 5. If a local sink l_i which is part of component C_i connects to a node v_j, its distance to all nodes not in C_i is at least $\overline{l_i v_j}$. So only nodes which are members of component C_i or nodes with the same distance from l_i as v_j are affected by the new arc. Furthermore a component establishes maximally one new arc in a single round and maximally 6 local sinks can establish an arc to the same node.[3] All this shows that the interference value of a node is maximally incremented by a constant in a single round of NCC.

Together with part one of the proof and the fact that the $movesink$ procedure is applied at most one more time, we see that the interference value of any node is incremented at most $\log(n) + 1$ times and each time by at most a constant value. This proves that the interference of the whole network is in $\mathrm{O}(\log n)$.

The only remaining part of Theorem 5.1 we need to prove states that NCC terminates in polynomial time. Every node v_i is a sink in one iteration of the while loop. (Actually it can be a sink in more than one loop iteration if a cycle is broken by removing the arc originating at that node. The fact however that for every such removed arc at least one other arc is added to the tree in the same round entails only an additional factor 2.) A sink has to find its nearest neighbor in a foreign component. This can be implemented using a list of neighbors, sorted according to their distances for each node and a union-find structure to check if a node is in a foreign component. The n lists of sorted neighbors can be constructed in time $\mathrm{O}(n \cdot n \log(n))$. Maintaining the union-find structure and component membership lookups during the execution of NCC can also be done in time $\mathrm{O}\big(n^2 \log(n)\big)$, while the total cost for shortest path computation is in $\mathrm{O}\big(n \cdot n^2\big)$. These observations prove that NCC terminates in polynomial time. ■

[3]This is the so-called "kissing number." It is defined as the number of equivalent spheres that touch an equivalent sphere without intersections. The kissing number in the two-dimensional plane is 6. In three dimensions it is 12.

We present NCC in a centralized manner. This reflects the fact that in a sensor network we have an instance (the sink) commonly assumed to have much more computing power and energy than all other nodes (sensors). Therefore the sink can run NCC and distribute the topology information of the constructed sink tree in an initialization phase.

Nevertheless a distributed variant of NCC without the co-ordination of a central instance is feasible. This variant would require counters in each node which keep track of the number of component unions the node was involved in since the start of the algorithm. These counters then guarantee that only components in the same "round" can establish new arcs between each other. Also computation of shortest paths and the $movesink$ procedure are implementable in a distributed way using a variant of flooding and by sending according messages over the thereby found shortest path, respectively.

VI. Interference in Average-Case Networks

We have seen that the NCC algorithm has asymptotically optimal worst-case behavior in the sense that it produces interference not greater than $\mathrm{O}(\log n)$ for all possible node arrangements. In this section we will have a closer look at the average case behavior of our algorithm. We do this by simulation. The nodes in our simulations are distributed randomly and uniformly in a square field. Also the sink is chosen randomly. To see how our algorithm behaves in average-case networks we compare it to two other construction methods for sink trees which have been proposed previously as data gathering structures. These two methods are:

1) The Minimum Spanning Tree algorithm (MST) with weights equal to the Euclidean edge lengths, all edges pointing towards the global sink.
2) The Shortest Path Tree algorithm (SPT) with respect to the energy metric. (The SPT contains the shortest paths from all nodes to the sink.) In the energy metric the cost of an edge equals its Euclidean length raised to the power of two. All edges point towards the global sink.

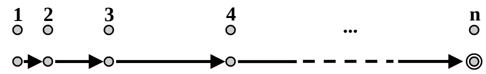

Fig. 7. Nodes arranged on a horizontal line with exponentially increasing distances and the rightmost node chosen as the global sink. Applied on this setting the MST algorithm produces interference $n-2$.

In order to allow for evaluation in different conditions all three algorithms constructed sink trees for networks with different node densities. We simulated networks from 10 to 500 nodes distributed in a square with its side length chosen such that all nodes are mutually visible (in order to emphasize the differences between the simulated algorithms without restricting the set of eligible arcs by introduction of the Unit Disk Graph model). Plotted in the diagram are the averaged values over 100 runs for each simulated node density.

Figure 6 shows that all three algorithms produce rising interference with increasing node densities. Closer observation yields that our NCC algorithm performs better than the Shortest Path Tree algorithm but worse than the Minimum Spanning Tree algorithm. This is quite intriguing as the very simple MST algorithm, which was not explicitly designed to reduce interference, seems to outperform our NCC algorithm in average-case networks. Note however that MST is not asymptotically worst-case optimal and can produce interference of $n-2$ for a sensor network consisting of n nodes. A sample of such an arrangement is shown in Figure 7.

VII. Conclusion

The approach we assume in this paper in order to study interference in wireless and particularly sensor networks differs from most of the previous work in two ways: First, we introduce an explicit definition of interference. Second, our definition of interference is receiver-centric and reflects the fact that message collisions prevent proper message reception only if they occur at the receiving node.

With this formalized notion of interference we show on the one hand that there exist instances of sensor networks with n nodes in which it is impossible to construct a sink tree—a valid data gathering structure—with interference less than $\log n - 1$. On the other hand we describe the NCC algorithm asymptotically matching this lower bound in that it provably builds a sink tree with interference at most $\mathrm{O}(\log n)$ on any given sensor network. In addition to these worst-case observations we also evaluate the NCC algorithm in average networks. Intriguingly the latter results show that—although the interference values produced by NCC fall roughly in the same range as those of other constructions—a simple minimum-spanning-tree-based structure keeps interference at a lower level than NCC in average-case networks.

In this paper we focus on the interference aspect in sensor networks and neglect the fact that communication over long links is more energy consuming than over short links. Furthermore, our model does not capture the notion of signal-to-noise ratio, which characterizes the condition for proper signal reception in practical radio networks by stating that the signal to be received is required to be strong enough compared to the sensed noise or interference. We therefore consider our work to be a first step towards understanding the complex interplay between interference and energy efficiency in sensor networks.

Acknowledgements

We would like to thank Thomas Moscibroda and Pascal von Rickenbach for valuable discussions.

References

[1] M. Burkhart, P. von Rickenbach, R. Wattenhofer, and A. Zollinger, “Does Topology Control Reduce Interference?” in *Proc. of the 5^{th} ACM Int. Symposium on Mobile Ad-Hoc Networking and Computing (MOBIHOC)*, 2004.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless Sensor Networks: A Survey,” *Computer Networks*, vol. 38, pp. 393–422, 2002.

[3] C.-Y. Chong and S. P. Kumar, “Sensor Networks: Evolution, Opportunities, and Challenges,” *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, 2003.

[4] D. Estrin et al., *Embedded, Everywhere: A Research Agenda for Networked Systems of Embedded Computers*. Washington DC: The National Academies Press, 2001.

[5] A. Goel and D. Estrin, “Simultaneous Optimization of Concave Costs: Single Sink Aggregation or Single Source Buy-at-Bulk,” in *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2003.

[6] J. Chou, D. Petrovic, and K. Ramchandran, “A Distributed and Adaptive Signal Processing Approach to Reducing Energy Consumption in Sensor Networks,” in *Proc. of the 22^{nd} Annual Joint Conference of the IEEE Computer and Comunications Societies (INFOCOM)*, 2003.

[7] P. von Rickenbach and R. Wattenhofer, “Gathering Correlated Data in Sensor Networks,” in *Proc. of the ACM Joint Workshop on Foundations of Mobile Computing (DIALM-POMC)*, Philadelphia, Pennsylvania, USA, October 2004.

[8] T. Hou and V. Li, “Transmission Range Control in Multihop Packet Radio Networks,” *IEEE Trans. on Communications*, vol. 34, no. 1, pp. 38–44, 1986.

[9] L. Hu, “Topology Control for Multihop Packet Radio Networks,” *IEEE Trans. on Communications*, vol. 41, no. 10, 1993.

[10] N. Li, C.-J. Hou, and L. Sha, “Design and Analysis of an MST-Based Topology Control Algorithm,” in *Proc. of the 22^{nd} Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2003.

[11] X.-Y. Li, G. Calinescu, and P.-J. Wan, “Distributed Construction of Planar Spanner and Routing for Ad Hoc Wireless Networks,” in *Proc. of the 21^{st} Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2002.

[12] R. Ramanathan and R. Rosales-Hain, “Topology Control of Multihop Wireless Networks Using Transmit Power Adjustment,” in *Proc. of the 19^{th} Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2000.

[13] V. Rodoplu and T. H. Meng, “Minimum energy mobile wireless networks,” *IEEE J. Selected Areas in Communications*, vol. 17, no. 8, 1999.

[14] H. Takagi and L. Kleinrock, “Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals,” *IEEE Trans. on Communications*, vol. 32, no. 3, pp. 246–257, 1984.

[15] Y. Wang and X.-Y. Li, “Localized Construction of Bounded Degree Planar Spanner,” in *Proc. of the DIALM-POMC Joint Workshop on Foundations of Mobile Computing*, 2003.

[16] R. Wattenhofer, L. Li, P. Bahl, and Y.-M. Wang, “Distributed Topology Control for Power Effi cient Operation in Multihop Wireless Ad Hoc Networks,” in *Proc. of the 20^{th} Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2001.

[17] F. Meyer auf der Heide, C. Schindelhauer, K. Volbert, and M. Grünewald, “Energy, Congestion and Dilation in Radio Networks,” in *Proceedings of the 14th ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, 2002.

[18] K. Moaveni-Nejad, W.-Z. Song, W.-Z. Wang, Y. Wang, and X.-Y. Li, “Low-Interference Topology Control for Wireless Ad Hoc Networks,” in *Proc. of the 1^{st} Int. Workshop on Theoretical Aspects of Wireless Ad Hoc, Sensor and Peer-to-Peer Networks (TAWN)*, 2004.