The Worst-Case Capacity of Wireless Sensor Networks

Thomas Moscibroda
Microsoft Research
Redmond WA 98052
moscitho@microsoft.com

ABSTRACT

The key application scenario of wireless sensor networks is *data gathering*: sensor nodes transmit data, possibly in a multi-hop fashion, to an information sink. The performance of sensor networks is thus characterized by the *rate* at which information can be aggregated to the sink. In this paper, we derive the first scaling laws describing the achievable rate in *worst-case*, i.e. arbitrarily deployed, sensor networks. We show that in the *physical model* of wireless communication and for a large number of practically important functions, a sustainable rate of $\Omega(1/\log^2 n)$ can be achieved in *every network*, even when nodes are positioned in a worst-case manner. In contrast, we show that the best possible rate in the *protocol model* is $\Theta(1/n)$, which establishes an exponential gap between these two standard models of wireless communication. Furthermore, our *worst-case capacity* result almost matches the rate of $\Theta(1/\log n)$ that can be achieved in randomly deployed networks. The high rate is made possible by employing non-linear power assignment at nodes and by exploiting SINR-effects. Finally, our algorithm also improves the best known bounds on the *scheduling complexity* in wireless networks.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless Communication*

General Terms

Algorithms, Theory

Keywords

capacity, scheduling complexity, data gathering

1. INTRODUCTION

Most if not all application scenarios of wireless sensor networks—both currently deployed and envisioned in the future—broadly follow a generic *data gathering and aggregation* paradigm: By sensing a geographic area or monitoring physical objects, sensor nodes produce relevant information that has to be transmitted to an information sink for further processing. The primary purpose of sensor networks is therefore to provide users access to the information gathered by the spatially distributed sensors, rather than enabling end-to-end communication between all pairs of nodes as in other large-scale networks such as the Internet or wireless mesh networks. The key technique that enables efficient usage of typically resource-limited wireless sensor networks is *in-network information processing*. Sensor nodes cooperate to process and aggregate information as it is transmitted to the sink, ideally providing the sink with enough information to compute the aggregate functions of interest, while minimizing communication overhead.

The performance of a sensor network can therefore be characterized by the *rate* at which data can be aggregated and transmitted to the information sink. In particular, the theoretical measure that captures the possibilities and limitations of information processing in sensor networks is the many-to-one *data aggregation capacity*, or its inverse, the maximum sustainable rate (bit/s) at which each sensor node can continuously transmit data to the sink.

Given the fundamental importance of this “computational throughput capacity” in sensor networks [9], it is not surprising that there already exists a rich literature that deals with scaling laws for the achievable data aggregation rate in wireless sensor networks under various models and for different functions, e.g. [8, 1, 19, 11]. In this paper, we take an entirely new approach to the data aggregation problem in sensor networks in particular, and to capacity problems in wireless networks in general. We do so by complementing and extending the current literature in two directions.

First and foremost, we initiate the study of the *worst-case capacity* of sensor networks. Starting from the seminal work of Gupta and Kumar [12], scaling laws on the capacity in wireless networks have been almost exclusively derived making assumptions regarding the placement of the nodes in the network. The standard assumption is that nodes are either located on a grid-like regular structure, or else are randomly and uniformly distributed in the plane following a certain density function.

In contrast, we advocate studying scaling laws that capture the achievable rate in *worst-case*, i.e., arbitrarily deployed sensor networks. This novel notion of *worst-case capacity* concerns the question of how much information can each node transmit to the source, *regardless* of the network's topology. The motivation for studying arbitrary node distributions stems from the following practical consideration. Many sensor networks feature heterogenous node densities

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
IPSN'07, April 25-27, 2007, Cambridge, Massachusetts, USA.
Copyright 2007 ACM 978-1-59593-638-7/07/0004 ...$5.00.

and there are numerous application scenarios—for instance networks deployed indoors, or along a road—where node placement contains or resembles typical "worst-case" structures such as chains. Studying the worst-case capacity of wireless networks therefore offers an exciting alternative to the idealized capacity studies conducted so far.

One criticism that worst-case analysis is frequently confronted with is that it tends to focus on oversimplified models for wireless communication in order to keep the analysis concise enough for stringent proofs. And indeed, algorithmic work on data gathering and the many-to-one communication problem has been based on simplified protocol or graph-based models, e.g. [15, 10, 3]. Similarly, with the exception of the work by Barton and Zheng [1], capacity studies in sensor networks have been using the protocol model [8, 19]. Studying simplified models such as the protocol model has generally been accepted as a reasonable first step by both practitioners and theoreticians, because—that was the premise—results obtained in these models do not divert too dramatically from more realistic models and, ultimately, from reality.

Surprisingly, this assumption turns out to be fundamentally wrong when it comes to the worst-case capacity of wireless sensor networks. Specifically, we prove that for a large number of practically important functions, the achievable rate in the protocol model is $\Theta(1/n)$, whereas a rate of $\Omega(1/\log^2 n)$ can be achieved in *every sensor network* in the physical SINR model. Hence, there is an exponential gap between the maximally sustainable data rate that each sensor node can transmit to the information sink between these two standard models of wireless communication.

Although the paper thus generalizes the study of capacity in sensor networks in two dimensions—arbitrary *worst-case network topologies* and the *physical SINR model*—, it almost matches the best currently known rates that hold in uniformly distributed networks under the protocol model. In particular, for symmetric functions such as the max or avg, a sustainable rate of $\Omega(1/\log^2 n)$ is achievable in every network, even if its nodes are positioned in a worst-case manner and without using block coding, or any other involved coding technique. In comparison, it follows from a result by Giridhar and Kumar that in the well-behaved setting of uniformly distributed nodes and in the protocol model, the maximum achievable rate is $\Theta(1/\log n)$ without using block coding. This implies that the *price of worst-case node placement* (the maximum ratio between the achievable rates in worst-case and in uniformly distributed networks) is merely a logarithmic factor in the physical model, whereas it is exponentially higher in the protocol model.

The key technique that allows us to break the capacity barrier imposed by the protocol model is to employ an involved, non-intuitive *power level assignment* at the nodes. In particular, we make use of the fact that the wireless medium can be "overloaded" (see Figure 1 in Section 2) in the sense that links of different order of length can be scheduled in parallel by scaling up the transmission power of short links [22].

The paper also contains the following additional results. First, the above rates can be improved using *block coding techniques* described by Giridhar and Kumar in [8]. Using these techniques in the phyisical model, the achievable rate improves to $\Theta(1/\log\log n)$ in worst-case networks. This is an astonishing result, because it matches exactly the optimal rate that can be achieved even in randomly deployed networks. That is, when using block coding in combination with our algorithm, the price of worst-case node placement becomes a constant. It can also be shown that our algorithm yields an improved scaling law result for the so-called *scheduling complexity of wireless networks* [21].

To the best of our knowledge, this paper presents the first scaling laws on the worst-case capacity and on the price of worst-case node placement in sensor networks in the physical model. Our results imply that if achieving a high data rate is a key concern, using an involved power control mechanism at nodes is indispensable. Also, the exponential gap between physical and protocol model in wireless networks renders the study of simplified protocol models questionable.

Tables 1 and 2 summarize our new results on worst-case capacity in sensor networks and compares them to the known capacity results in uniformly distributed networks [8]. The tables also show the fundamental *price of worst-case node placement* in wireless sensor networks.

	random, uniform deployment	worst-case deployment	"worst-case penalty"
Protocol	$\Theta(1/\log n)$ [8]	$\Theta(1/n)$	linear
Physical	$\Omega(1/\log n)$	$\Omega(1/\log^2 n)$	logarithmic

Table 1: Achievable rate without block-coding.

	random, uniform deployment	worst-case deployment	"worst-case penalty"
Protocol	$\Theta(1/\log\log n)$ [8]	$\Theta(1/\log n)$	logarithmic
Physical	$\Omega(1/\log\log n)$	$\Omega(1/\log\log n)$	constant

Table 2: Achievable rate with block-coding.

The remainder of the paper is organized as follows. Section 2 formalizes the data gathering problem in sensor networks and defines the different models. While Section 3 proves a negative bound regarding the protocol model, our main result—the algorithm achieving high rate even in worst-case sensor networks—is presented in Section 4. Section 5 shows how block coding techniques from [8] can be used to further improve the rate. The relationship between our capacity results and the scheduling complexity of wireless networks is briefly discussed in Section 6. Finally, Section 7 reviews related work.

2. PROBLEM STATEMENT AND MODELS

2.1 Network Model

We consider a network of n sensor nodes $X = \{x_1, \ldots, x_n\}$ located arbitrarily in the plane. Additionally, there is one designated *sink* node s, where the sensed data eventually has to be gathered. The Euclidean distance between two nodes i and j is denoted by d_{ij} and let d_{max} be the maximal distance between any two nodes in the network. No two nodes are exactly co-located, but mutual distances can be arbitrarily small. As pointed out in the introduction, we investigate the capacity of networks whose nodes may be placed in a worst-case manner. Formally, we assume the existence of an imaginary adversary that selects the node positions in such a way as to minimize the achievable rate.

One crucial criterium according to which communication models for wireless networks can be partitioned is the description of the circumstances under which a message is received by its intended recipient. In the so-called *protocol*

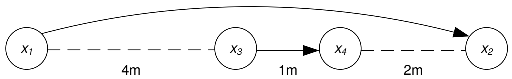

Figure 1: In the physical model, transmissions from x_1 to x_2 and from x_3 to x_4 can simultaneously be scheduled successfully, whereas in the protocol model, two time slots are required.

model [12], a node x_i can successfully transmit a packet to node x_j if $d_{ij} \leq r_i$, where r_i is x_i's transmission range, and if for every other simultaneously transmitting node x_k, $d_{kj} \geq (1+\Delta)r_k$. That is, x_j must be outside of every such node's interference range, which exceeds its transmission range by a factor of Δ.

The other standard model of wireless communication frequently adopted in networking community is the *physical model* [12]. In this model, the received power on the medium is assumed to decay with distance deterministically at an exponential rate with path-loss exponent $\alpha > 2$, which is a fixed constant between 2 and 6. Whether a message is received successfully at the intended receiver depends on the received signal strength, the ambient noise level, and the interference caused by simultaneously transmitting nodes. Formally, a message from x_i is successfully received by x_j, if the perceived *signal-to-noise-plus-interference ratio* (SINR) at x_j is above a certain threshold β, i.e., if

$$\frac{\frac{P_i}{d(x_i,x_j)^\alpha}}{N + \sum_{x_k \in X \setminus \{x_i\}} \frac{P_k}{d(x_j,x_k)^\alpha}} \geq \beta, \qquad (1)$$

where P_i denotes the transmission power selected by node x_i in the specific time slot. Since the main concern of this paper is to obtain scaling laws for the worst-case capacity as the number of nodes n grows, we assume α to be a fixed constant strictly larger than 2. It is well-known that for α equal to or very close to 2, no scheduling algorithm can perform well [13].

It is important to observe that these two standard models of wireless communication allow for fundamentally different communication patterns. Assume that in the simple four node example depicted in Figure 1 [22], node x_1 wants to transmit to x_2, and x_3 wants to transmit to x_4. In the protocol model (and in any other graph-based model), at most one of these two transmissions can successfully be scheduled in parallel, i.e., two time slots are required to schedule both transmissions. In the physical model, for $\alpha = 3$ and $\beta = 3$, however, both transmissions can easily be scheduled in a single time slot when setting the power levels appropriately. Specifically, assume that the transmission powers are $P_{x_1} = 1dBm$ and $P_{x_3} = -15dBm$, and let $\beta_{x_2}(x_1)$ and $\beta_{x_4}(x_3)$ denote the SINR values at receivers x_2 and x_4 from their intended senders x_1 and x_3, respectively. The following calculation [22] shows that even when both transmissions occur simultaneously, the receivers can decode their packets:

$$\beta_{x_2}(x_1) = \frac{1.26mW/(7m)^3}{10\mu W + (31.6\mu W/(3m)^3)} \approx 3.11$$

$$\beta_{x_4}(x_3) = \frac{31.6\mu W/(1m)^3}{10\mu W + (1.26mW/(5m)^3)} \approx 3.13$$

As we will see in Section 4, it is necessary to exploit these physical model properties of wireless communication in order to derive a high worst-case capacity in sensor networks.

2.2 Data Aggregation Problem

The maximum achievable rate in a packet-based collision model of wireless sensor networks has first been formalized and studied by Giridhar and Kumar in [8]. In this problem, each node x_i periodically senses its environment, and measures a value that belongs to some fixed finite set $\mathcal{X}$ (for instance a temperature up to a certain precision). It is the goal of a data gathering protocol to repeatedly compute the specific function $f_n : \mathcal{X}^n \rightarrow \mathcal{Y}_n$, and communicate its result to the sink s. Since sensor measurements are produced periodically, the function of interest must be computed repeatedly at the sink. Formally, the period of time during which every sensor node produces exactly one measurement is called a *measurement cycle*, and the sink must compute the value of f_n for every such measurement cycle.

In this work, we consider a practically important set of symmetric functions that we call "perfectly compressible". A function is perfectly compressible if all information concerning the same measurement cycle contained in two or more messages can be perfectly aggregated in a single new packet of equal size. Functions such as the mean, max, or various kinds of indicator functions belong to this category.

Notice that none of the results in this paper is based on any kind of information-theoretic collaborative techniques such as network coding [16], interference cancelation techniques, or superposition coding [1]. The impact of *block coding* strategies [8], which are based on combining the function computations of consecutive or subsequent measurement cycles, are considered in Section 5. Sections 4 and 3 do not consider block coding.

As customary, we assume without loss of generality that time is slotted into synchronized slots of equal length [12, 8]. In each time slot t, each node x_i is assigned a power level $P_i(t)$, which is strictly positive if the sensor node tries to transmit a message to another node. A *power assignment* $\phi_t : X \mapsto \mathbb{R}_0^+$ determines the power level $\phi_t(x_i)$ of each node $x_i \in X$ in a certain time slot. If t is clear from the context, we use the notational short-cut $P_i = \phi_t(x_i)$.

A *schedule* $\mathcal{S} = (\phi^1, \ldots, \phi^{|\mathcal{S}|})$ is a sequence of $|S|$ power assignments, where ϕ^k denotes the power assignment in time slot k. That is, a schedule $\mathcal{S}$ determines the power level P_i for every node $x_i \in X$ for $|\mathcal{S}|$ consecutive time slots. A *strategy or scheme* $\mathcal{S}_n$ determines a sequence of power assignments $(\phi_1, \phi_2, \ldots)$ and computations at sensors, which, given any $\hat{X} \in \mathcal{X}^n$, results in the result $f(\hat{X})$ becoming known to the sink. Finally, let $T(\mathcal{S}_n)$ denote the worst-case time required by scheme $\mathcal{S}_n$ over all possible measurements $\hat{X} \in \mathcal{X}^n$ and over all possible placements of n nodes in the plane. The value $R(\mathcal{S}_n) := \frac{1}{T(\mathcal{S}_n)}$ is the *rate* of $\mathcal{S}_n$ and describes the *worst-case capacity* of the sensor network.

3. WORST CASE CAPACITY IN PROTOCOL MODEL

From a worst-case perspective, the *protocol model* is much simpler than the physical model discussed in Section 4. For this model, Giridhar and Kumar prove a variety of asymptotically tight results on the achievable rate in single-hop networks and networks that are deployed uniformly at random [8]. If transmission powers are fixed, a worst case network can clearly be as bad as a single-hop (collocated) network because all nodes can be very close to each other. In this section, we briefly sketch how to extend this result to worst-case networks even when transmission powers at nodes can be assigned optimally according to the given topology.

THEOREM 3.1. *In the protocol model with interference parameter Δ, the maximum rate for computing type sensitive and type threshold functions is $\Theta(\frac{1}{n})$ without block coding.*

PROOF. Consider nodes $x_1, \ldots, x_n$ located on a line with sink s at position 0, and x_i at position δ^{i-1}, for $\delta = 1 + \frac{1}{\Delta}$. Due to the exponential increase of the inter-node distances, scheduling any link from a node x_i interferes with every other link to its left in the network. Therefore, under the protocol model, this network behaves like a single-hop network, even if transmission powers are chosen optimally. Hence, the theorem follows from Theorem 3 in [8]. □

As shown in the next section, this result is worse by an exponential factor compared to the achievable rate in worst-case networks under the physical model, which drastically *separates* these two standard models for wireless network communication.

4. WORST CASE CAPACITY IN PHYSICAL MODEL

In this section, we establish our asymptotic lower bound on the worst-case capacity of sensor networks by designing an algorithmic method whose input is the set of sensor nodes and the aggregation function f, and whose output is a scheme $\mathcal{S}_n$ that achieves an asymptotic rate of at least $\Omega(\frac{1}{\log^2 n})$ in every network. We describe the method in a top-down way. First, we present a simple procedure that computes the data gathering tree $T(X)$ on which the scheduling scheme is based. Second, we give a high-level version of the function computation scheme that makes use of an abstract implementation of a *phase scheduler*, which manages to successfully and efficiently schedule transmissions on the physical layer. The actual implementation of the phase scheduler is then at the heart of the algorithm.

4.1 Data Gathering Algorithm - High Level

We begin by computing a hierarchical tree structure that consists at most $\log n$ so-called nearest neighbor trees [6]. Throughout the procedure, an active set $\mathcal{A}$ of nodes is maintained. In each iteration, the algorithm computes a *nearest neighbor forest* spanning $\mathcal{A}$, in which there is a directed link ℓ_{ij} from each active node $x_i \in \mathcal{A}$ to its *closest* active neighbor $x_j \in \mathcal{A}$. At the end of an iteration, only one node of each tree remains in $\mathcal{A}$ and continues executing the algorithm. The union of links thus created is called $T(X)$. Note that every $x_i \in X$ has *exactly one* outgoing link in $T(X)$.

Algorithm 1 Data Gathering Tree $T(X)$

$\mathcal{A} := X;\ T(X) := \emptyset$;
while $|\mathcal{A}| > 1$ **do**
  for each $x_i \in \mathcal{A}$ **do**
    choose $x_j \in \mathcal{A} \setminus \{x_i\}$ minimizing $d(x_i, x_j)$;
    if $\ell_{ji} \notin T(X)$ **then** $T(X) := T(X) \cup \ell_{ij}$; **fi**
  end for
  for each $x_i \in \mathcal{A}$ with $\ell_{ij} \in T(X)$ **do** $\mathcal{A} := \mathcal{A} \setminus \{x_i\}$;
end while
Add to $T(X)$ a link from the last node $x_i \in \mathcal{A}$ to s;

We next describe how the links of $T(X)$ are scheduled to achieve a good rate. Let $D_{T(X)} \leq n$ denote the depth of tree $T(X)$ and define a variable h_i for each node x_i according to its hop-distance in $T(X)$ to the sink s: One-hop neighbors of s have $h_i := D_{T(X)} - 1$, two-hop neighbors have $h_i := D_{T(X)} - 2$, and so forth. The node with the highest hop-distance from the sink has $h_i := 0$. The variables h_i induce a *layering* of $T(X)$ with the first layer (the one being furthest away from s) being assigned the value 0.

Consider the k^{th} round of measurements taken by the nodes. All data for this measurement is forwarded towards the sink node in a hop-by-hop fashion and aggregated on the way in each node. Since the measurement data of nodes in different tree layers requires a different amount of time to reach the sink, the forwarding of measurement data at nodes close to the sink is delayed in such a way that data aggregation at internal nodes is always possible. Hence, the information is sent to the root in a *pipelined* fashion, advancing one hop in every $L(X)$ time slots, where $L(X)$ is the number of consecutive time slots required until every link in $T(X)$ can be scheduled successfully at least once. In our algorithm, $L(X)$ will be $c \log^2 n$ for some constant c.

Algorithm 2 Forwarding & Aggregation Scheme

Node x_i receives the data for the k^{th} measurement from each subtree in $T(X)$ by time $(h_i + k)L(X)$;
Node x_i aggregates the k^{th} measurement data from all its children;
Node x_i sends the aggregated message over link ℓ_{ij} to its parent x_j in $T(X)$ in time slot $(h_i+k)L(X)+t(\ell_{ij})$ with transmission power $P(\ell_{ij})$, where $t(\ell_{ij}) \in \{0, \ldots, L(X)\}$ is the intra-phase time slot allocated to link ℓ_{ij} by the **phase scheduler**.

The forwarding and data aggregation scheme described in Algorithm 2 is straight-forward. However, notice that it does not describe the actual physical scheduling procedure of the forwarding scheme, thus leaving open its most crucial aspect. In particular, Algorithm 2 uses an abstract "phase scheduler" that assigns an intra-phase time slot and transmission power level to each node in every phase of duration $L(X)$. The crucial ingredient to make the algorithm work therefore lies in the allocation of the intra-phase time slots $t(\ell_{ij})$ and power levels $P(\ell_{ij})$. In particular, we must make sure that the proclaimed transmission can actually be performed in a successful way in the physical model, even in worst-case networks. Before defining and analyzing this "phase scheduler", we state the following general lemma.

LEMMA 4.1. *Consider a network X and its data gathering tree $T(X)$. Assume the existence of a* phase scheduler *procedure that successfully schedules each link of $T(X)$ at least once in an interval of $L(X)$ consecutive time slots. The resulting data gathering scheme has a rate of $\Omega(1/L(X))$.*

PROOF. Since every link of $T(X)$ is scheduled at least once in every phase of length $L(X)$, the kth measurement from a node x_i moves (in a pipelined fashion) at least one hop closer to the sink in $L(X)$ time slots. Since we consider perfectly compressible functions, and by the definition of h_i, if follows that node x_i receives all aggregated information from its entire subtree by the time $(h_i+k)L(X)$. It can then aggregate its own kth measurement and send the new message to its parent in $T(X)$ in at least one time slot in the interval $(h_i + k)L(X) + 1, \ldots, (h_i + k + 1)L(X)$. By induction, the root receives the aggregated information about one round of measurements in each time interval $L(X)$. □

4.2 Implementing the Phase Scheduler

The crucial component when devising a data gathering scheme with high worst-case rate is an efficient phase scheduler. The difficulty of this stems from the fact that intuitive scheduling and power assignment schemes fail in achieving a good performance. It was shown in [21] that neither uniform power allocation nor the frequently studied distance-depending power allocation strategy $P \sim d^\alpha$ (i.e., proportionally to the length of the communication link) yields acceptable results. To see this, consider an exponential node chain depicted in Figure 2. If every node transmits with the same power, nodes on the left will experience too much interference and only a small constant number of nodes can send simultaneously, resulting in a low rate of $O(1/n)$. Similarly, as shown formally in [21], if every node sends at a power proportional to $P \sim d^\alpha$, at most a small constant number of nodes can transmit simultaneously because nodes close to the sink will face too much interference. Again, the rate with such a strategy cannot exceed $O(1/n)$.

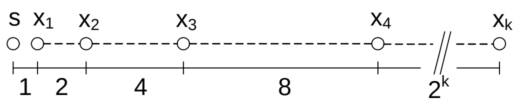

Figure 2: Network in which achieving a rate better than $O(1/n)$ is difficult.

The key insight that allows to increase the rate in the network shown in Figure 2 is to use an involved (and significantly more complicated) power assignment scheme that is based on the ideas exemplified in Figure 1. Intuitively, senders of short links (those close to the root) must transmit at a proportionally higher transmission power (higher than $P \sim d^\alpha$, but still less than uniform) in order to "overpower" the interference created by simultaneously transmitting nodes. Based on this high-level idea, we now present a phase scheduler, which manages to successfully schedule each link of the data gathering tree $T(X)$ at least once in $O(\log^2 n)$ time slots. In combination with Lemma 4.1, this leads to the following main theorem.

THEOREM 4.2. *In the physical model and for perfectly compressible functions, the achievable rate in worst-case sensor networks is at least $\Omega(1/\log^2 n)$ even without the use of block coding techniques.*

We begin by describing the phase scheduler whose details are given in Algorithm 3. The phase scheduler consists of three parts: a pre-processing step, the main scheduling-loop, and a test-subroutine that determines if a link is to be scheduled in a given time slot t, in which another set L_t of links is already allocated.

In the pre-processing step, every link is assigned two values τ_{ij} and γ_{ij}. The value γ_{ij} indicates into which of at most $\xi\lceil\log(\xi\beta)\rceil$ different "link sets" the link belongs. Each link is in exactly one such set and only links with the same γ_{ij} values are considered for scheduling in the same iteration of the main scheduling-loop. The reason for this partitioned scheduling is that all links with the same γ value have the property that their length is either very similar or vastly different, but not in between. This will be essential in the scheduling procedure. The value τ_{ij}, further partitions the requests. Generally, small values of τ_{ij} indicate long communication links. More specifically, it holds that for two links ℓ_{ij} and ℓ_{gh} with $\tau_{ij} < \tau_{gh}$, then $d_{ij} \geq \frac{1}{2}(\xi\beta)^{\xi(\tau_{gh}-\tau_{ij})} \cdot d_{gh}$.

Algorithm 3 Phase Scheduler for Tree $T(X)$

***pre-processing*($T(X)$):**

1: $\tau\text{cur} := 1$; $\gamma\text{cur} := 1$; $\xi := 2N(\alpha-1)/(\alpha-2)$;
2: last $:= d_{ij}$ for the longest $\ell_{ij} \in T(X)$;
3: **for each** $\ell_{ij} \in T(X)$ in decreasing order of d_{ij} **do**
4:   **if** last$/d_{ij} \geq 2$ **then**
5:     **if** $\gamma\text{cur} < \xi\lceil\log(\xi\beta)\rceil$ **then**
6:       $\gamma\text{cur} := \gamma\text{cur} + 1$;
7:     **else**
8:       $\gamma\text{cur} := 1$; $\tau\text{cur} := \tau\text{cur} + 1$;
9:     **end**
10:     last $:= d_{ij}$;
11:   **end**
12:   $\gamma_{ij} := \gamma\text{cur}$; $\tau_{ij} := \tau\text{cur}$;
13: **end**

***schedule*($T(X)$):**

1: Define a large enough constant c_1 and let $t := 0$;
2: **for** $k = 1$ **to** $\xi\lceil\log(\xi\beta)\rceil$ **do**
3:   Let $T_k := \{\ell_{ij} \in T(X) \mid \gamma_{ij} = k\}$;
4:   **while** not all links in T_k have been scheduled **do**
5:     $L_t := \emptyset$;
6:     Consider all $\ell_{ij} \in T_k$ in decreasing order of d_{ij}:
7:       **if** *check*(ℓ_{ij}, L_t) **then**
8:         $L_t := L_t \cup \{\ell_{ij}\}$; $T_k := T_k \setminus \{\ell_{ij}\}$
9:       **end if**
10:     For all $\ell_{ij} \in L_t$, set the intra-phase time slot $t(\ell_{ij}) := t$ and assign a transmission power $P_i(\ell_{ij}) = d_{ij}^\alpha \cdot (\xi\beta)^{\tau_{ij}}$;
11:     $t := t + 1$;
12:   **end while**
13: **end for**

check$(\ell_{\mathbf{ij}}, \mathbf{L_t})$

1: **for each** $\ell_{gh} \in L_t$ **do**
2:   **if** $\tau_{gh} = \tau_{ij}$ **and** $c_1 \cdot d_{ij} > d_{ig}$
3:     **then return false**
4:   **if** $\tau_{gh} \leq \tau_{ij}$ **and** $d_{gj} < d_{gh}$
5:     **then return false**
6:   **if** $\tau_{gh} < \tau_{ij} \leq \tau_{gh} + \log n$ **and** $d_{hj} < c_1 \cdot d_{gh}$
7:     **then return false**
8:   $\delta_{ig} := \tau_{ij} - \tau_{gh}$;
9:   **if** $\tau_{gh} + \log n < \tau_{ij}$ **and** $d_{hi} < n^{1/\alpha} \cdot d_{ij} \cdot (\xi\beta)^{\frac{\delta_{ig}+1}{\alpha}}$
10:     **then return false**
11: **end for**
12: **return true**

The main loop uses the subroutine **check**$(\ell_{\mathbf{ij}}, \mathbf{L_t})$ in order to determine whether—given a set of links L_t that is already selected for scheduling in intra-phase time slot t—an additional link $\ell_{ij} \notin L_t$ should simultaneously be scheduled or not. The subroutine evaluates to true, if ℓ_{ij} can be scheduled in t, and false otherwise. The decision criteria that the link must satisfy relative to every other link $\ell_{gh} \in L_t$ depends on their *relative length difference* $\delta_{ig} := \tau_{ij} - \tau_{gh}$. If there is no relative length difference ($\tau_{ij} = \tau_{gh}$), then the two senders transmit at roughly the same transmission power and a standard spatial-reuse distance suffices (Line 2). It is well-known that in scenarios in which all nodes transmit at roughly the same transmission power, leaving some "extra space" (the exact amount of which depends on α, β,...) between any pair of transmitters is enough to keep the interference level sufficiently low at all receivers.

As pointed out, our algorithm does not employ a uniform (or near-uniform) power allocation scheme, because any such strategy is doomed to achieve a bad worst-case capacity. That is, in scenarios with widely different link lengths (e.g. Figures 2), nodes must transmit at widely different transmission powers, which makes it difficult to select the right "spatial reuse distance". In fact, the example in Figures 1 shows that the very notion of a "reuse distance" is questionable. In our algorithm, a link ℓ_{ij} is only scheduled if for every link $\ell_{gh} \in L_t$, it holds that $d_{hj} \geq c_1 \cdot d_{gh}$, if $\tau_{ij} \leq \tau_{gh} + \log n$ (Line 6); and $d_{hi} \geq n^{1/\alpha} \cdot d_{ij} \cdot (\xi\beta)^{(\delta_{ig}+1)/\alpha}$, if $\tau_{ij} > \tau_{gh} + \log n$ (Line 9), respectively. Intuitively, as long as the length of two links is not too far away, the algorithm performs a standard spatial reuse procedure. As soon as the relative length difference becomes too large, $\delta_{ig} > \log n$, however, the standard spatial reuse concept is no longer sufficient and the algorithm uses a more elaborate scheme which achieves high spacial reuse even in worst-case networks.

EXAMPLE 4.1. *Consider two links ℓ_{ij} and ℓ_{gh} with $\gamma_{ij} = \gamma_{gh}$, $\tau_{ij} = 16$ and $\tau_{gh} = 3$, and let $n = 64$. Assume that ℓ_{gh} is already set to be scheduled in a given time slot, i.e., $\ell_{gh} \in L_t$. When ℓ_{ij} is considered, it holds that $\delta_{ig} = \tau_{ij} - \tau_{gh} = 13$. In this case, because $\tau_{ij} > \tau_{gh} + \log n$, ℓ_{ij} is added to the set of scheduled links L_t only if $d_{hi} \geq n^{1/\alpha} \cdot d_{ij} \cdot (\xi\beta)^{\frac{14}{\alpha}}$ holds.*

The intuition behind this "reuse distance" function which increases exponentially in δ_{ig} is that in the power assignment scheme adopted, the transmission power of senders with small links is scaled up significantly. Therefore, scheduling small links requires an enlarged safety zone in order to avoid interference with simultaneously scheduled longer links. This selection criterion is necessary to guarantee that even in worst-case networks, many links can be scheduled in parallel and yet, no receiver faces too much interference.

The phase scheduler's main procedure executes $\xi\lceil\log(\xi\beta)\rceil$ iterations, in each of which it attempts to quickly schedule all links $\ell_{ij} \in T_k$, i.e., all links having $\gamma_{ij} = k$. Essentially, the procedure greedily considers all remaining nodes in T_k in non-increasing order of d_{ij}, and verifies for each link in this order whether it can be scheduled using the $\mathbf{check}(\ell_{\mathbf{ij}}, \mathbf{L_t})$ subroutine. If a link can be scheduled, its intra-phase time slot $t(\ell_{ij})$ is set to the current value of t, and its transmission power is set to $P_i(\ell_{ij}) = d_{ij}^{\alpha} \cdot (\xi\beta)^{\tau_{ij}}$. In the following section, we will argue that each set T_k can be scheduled in at most $O(\log n)$ time slots, where the hidden constant depends on the values of α, β, as well as the noise power level N, all of which we consider to be a fixed constants. It then follows that the entire procedure requires $O(\log^2 n)$ time slots.

4.3 Analysis

In order to prove Theorem 4.2, we need to show that the assumptions of Lemma 4.1 are satisfied: every link of the tree $T(X)$ can be successfully scheduled at least once in time $L(X) \in O(\log^2 n)$ time. We start by proving that the number of intra-phase time slots assigned by the phase scheduler is bounded by $O(\log^2 n)$ in every network.

As shown in [6], the data gathering tree $T(X)$ has the following property: If we draw a disk with radius d_{ij} around the sender x_i of each link ℓ_{ij}, then every node is covered by at most $6 \log n$ different disks. The following lemma makes use of this result.

LEMMA 4.3. *Consider all links ℓ_{ij} in $T(X)$ of length $d_{ij} \geq R$. It holds that in any disk of radius R, there can be at most $C \log n$ end-points (receivers) of such links, for a constant C.*

PROOF. Since every node in $T(X)$ is covered by at most $6 \log n$ disks around links, the proof follows by standard geometric arguments. Partition any disk D into a constant number of cones of equal angle. For small enough angles and for each cone, there must be a sender x_i of a link ℓ_{ij} located in the cone, which is covered by at least a constant fraction of the disks around other sending nodes in this cone. With the result in [6], the lemma can thus be proven. □

LEMMA 4.4. *Consider two links ℓ_{ij} and ℓ_{gh} with $\gamma_{ij} = \gamma_{gh}$. If $\tau_{ij} \geq \tau_{gh}$, it holds that $d_{gh} \geq \frac{1}{2}(\xi\beta)^{\xi\delta_{ig}} \cdot d_{ij}$, where $\delta_{ig} = \tau_{ij} - \tau_{gh}$.*

PROOF. Note that ℓ_{ij} and ℓ_{gh} differ only in their τ values (and not in their γ values). By definition of the preprocessing phase, it holds that in order to reach $\gamma_{ij} = \gamma_{gh}$ for the next higher value of τ, γ_{ij} must be increased exactly $\xi\lceil\log(\xi\beta)\rceil$ times (and reset to 0 once). Hence, it must hold that γ_{ij} was increased at least $\xi(\tau_{ij} - \tau_{gh})\lceil\log(\xi\beta)\rceil$ times since processing ℓ_{gh}. By the condition of Line 4, all but one of these increases implies a halving of the length d_{ij}. From this, the lemma can be derived by simple calculations. □

In order to bound the number of time slots required to schedule all links in the main loop, we use the notion of *blocking links* [21].

DEFINITION 4.1. *A link ℓ_{gh} is a* blocking link *for ℓ_{ij} if $\gamma_{gh} = \gamma_{ij}$, $d_{gh} \geq d_{ij}$, and* $\mathbf{check}(\ell_{\mathbf{ij}}, \mathbf{L_t})$ *evaluates to false if $\ell_{gh} \in L_t$. B_{ij} denotes the set of blocking links of ℓ_{ij}.*

Blocking links $\ell_{gh} \in B_{ij}$ are those links that may prevent a link ℓ_{ij} from being scheduled in a given time slot. Because a single blocking link can prevent ℓ_{ij} from being scheduled in at most a single time slot per phase (when it is scheduled itself), it holds that even in the worst-case, ℓ_{ij} is scheduled at the latest in time slot $t \leq |B_{ij}| + 1$ of the for-loop iteration when $k = \gamma_{ij}$. In the sequel, we bound the maximum cardinality of $|B_{ij}|$ of links ℓ_{ij}. Notice that only larger links can be blocking links due to the decreasing order in Line 6. Let $B_{ij}^{=}$ and $B_{ij}^{>}$ be the set of blocking links $\ell_{gh} \in B_{ij}$ with $\tau_{ij} = \tau_{gh}$ and $\tau_{ij} > \tau_{gh}$ (i.e., significantly longer blocking links), respectively. Lemmas 4.5 and 4.6 bound these sets.

LEMMA 4.5. *For all links $\ell_{ij} \in T(X)$, the number of blocking links in $B_{ij}^{=}$ is at most $|B_{ij}^{=}| \in O(\log n)$.*

PROOF. Because of $\tau_{ij} = \tau_{gh}$ for every $\ell_{gh} \in B_{ij}^{=}$, it follows from Lemma 4.4 and the decreasing order in Line 6 that $d_{ij} \leq d_{gh} \leq 2d_{ij}$. By Lemma 4.3, we know that there can be at most $C \log n$ receivers of blocking links with length at least d_{ij} in any disk of radius d_{ij}. Because $c_1 d_{ij} > d_{ig}$ holds for any blocking link in $B_{ij}^{=}$, every receiver of a blocking link must be located inside a disk D of radius $(c_1+2)d_{ij}$ centered at x_i. Because this disk D can be covered by smaller disks of radius d_{ij} in such a way that every point in the plane is covered by at most two small disks, it follows that

$$|B_{ij}^{=}| \;\leq\; C \log n \cdot \frac{2\pi(c_1+2)^2 d_{ij}^2}{\pi d_{ij}^2} \;=\; 2(c_1+2)^2 C \log n. \qquad \square$$

Bounding the cardinality of $B_{ij}^{>}$ is significantly more involved. In particular, we need to distinguish three kinds of blocking links in $B_{ij}^{>}$, depending on which line of the $\mathbf{check}(\ell_{\mathbf{ij}}, \mathbf{L_t})$ subroutine caused the returning of **false**.

LEMMA 4.6. *For all links $\ell_{ij} \in T(X)$, the number of blocking links in $B_{ij}^{>}$ is at most $|B_{ij}^{>}| \in O(\log^2 n)$.*

PROOF. The proof unfolds in a series of three separate bounds that characterize the number of blocking links that can block ℓ_{ij} in Lines 4, 6, and 9, respectively. It follows directly from the property proven in [6] that the number of blocking links that may prevent ℓ_{ij} from being scheduled in Line 4 of the subroutine is at most $C \log n$.

We now bound the number of links that may block ℓ_{ij} in Line 6. By the definition of the $\mathbf{check}(\ell_{ij}, \mathbf{L_t})$ subroutine, the receiver of each such potential blocking link must be located within distance $c_1 \cdot \ell_{gh}$ of x_i. Consider a set of smaller disks of radius $\ell_{gh}/2$, that completely cover the large disk of radius $c_1 \cdot \ell_{gh}$ centered at x_i. By a covering argument, it holds that $8c_2^2$ smaller disks are sufficient to entirely cover the larger disk. From Lemma 4.3, we know that each such small disk may contain at most $C \log n$ receivers of links of length $\ell_{gh}/2$ or longer. From this, it follows that at most $8c_2^2 C \log n$ links with a specific $\tau = \tau_{gh}$ may be blocking in Line 6. Because only link-classes τ_{gh} with $\tau_{ij} - \log n < \tau_{gh} < \tau_{ij}$ may cause a blocking in Line 6, the total number of blocking nodes for ℓ_{ij} in Line 6 cannot surpass $\log n \cdot 8c_2^2 C \log n \in O(\log^2 n)$.

Finally, consider the third case: the set of potential blocking links ℓ_{gh} in $B_{ij}^{>}$ for which $\delta_{ig} > \log n$. Again, we show that there are at most $O(\log^2 n)$ potential blocking nodes in this category. Notice that we need an entirely different proof technique for this case, because—unlike in the case of Line 6—there may be up to $n - \log n$ many different such link-classes τ_{gh}. Hence, it is not sufficient to bound the number of blocking nodes in each length class individually.

We begin by showing that there exist at most $O(\log n)$ blocking links ℓ_{gh} whose receiver x_h is located in the range

$$d_{ih} \ \leq \ n^{1/\alpha} \cdot (\xi\beta)^{\log n/\alpha} \cdot d_{ij}$$

from x_i. By Lemma 4.4, we know that for every link ℓ_{gh} with $\tau_{ij} > \tau_{gh} + \log n$, it holds that

$$d_{gh} \ \geq \ \frac{1}{2}(\xi\beta)^{\xi \log n} d_{ij} \ > \ n^{1/\alpha} \cdot (\xi\beta)^{\log n/\alpha} \cdot d_{ij},$$

where the second inequality is due to the definition of ξ. It follows that every potential blocking link of Line 9 is longer than the radius of a disk of radius $n^{1/\alpha} \cdot (\xi\beta)^{\log n/\alpha} \cdot d_{ij}$ around x_i. Hence, Lemma 4.3 implies that there can be at most $O(\log n)$ such potential blocking links in this range.

We now show that for any integer $\varphi \geq 0$, there are $O(\log n)$ different blocking links ℓ_{gh} for which d_{ih} is in the range

$$n^{\frac{1}{\alpha}} \cdot (\xi\beta)^{\alpha^{\varphi-1} \log n} \cdot d_{ij} \ < \ d_{ih} \ \leq n^{\frac{1}{\alpha}} \cdot (\xi\beta)^{\alpha^{\varphi} \log n} \cdot d_{ij}. \quad (2)$$

Let B_{ij}^{φ} denote the set of potential blocking links having its receiver in the range for a specific φ. By comparing the "spatial reuse" condition in the $\mathbf{check}(\ell_{ij}, \mathbf{L_t})$ subroutine with the above range, it can be observed that every link $\ell_{gh} \in B_{ij}^{\varphi}$ must satisfy $(\delta_{ig}+1)/\alpha > \alpha^{\varphi-1} \log n$, and therefore $\delta_{ig} \geq \alpha^{\varphi} \log n$. Plugging this lower bound on δ_{ig} into Lemma 4.4 allows us to derive a minimum length for each blocking link $\ell_{gh} \in B_{ij}^{\varphi}$ with d_{ih} in the specified range: The length of each such $\ell_{gh} \in B_{ij}^{\varphi}$ must be at least

$$d_{gh} \ \geq \ \frac{1}{2}(\xi\beta)^{\xi\alpha^{\varphi} \log n} \cdot d_{ij}. \quad (3)$$

The important thing to realize is that the length of each blocking link is therefore *longer* than the outer range of the spatial reuse interval we consider, because

$$n^{\frac{1}{\alpha}} (\xi\beta)^{\alpha^{\varphi} \log n} \cdot d_{ij} \ \leq \ \frac{1}{2}(\xi\beta)^{\xi\alpha^{\varphi} \log n} \cdot d_{ij},$$

Therefore, we can apply the bound given in Lemma 4.3. Particularly, at most $C \log n$ receivers of links in B_{ij}^{φ} can be located in any ring between the radii specified in (2).

With this result, we can now conclude the proof of the lemma. We know that for any integer $\varphi \geq 0$, there are at most $C \log n$ blocking links in B_{ij}^{φ}. The maximum and minimum value for δ_{ig} of potential blocking links ℓ_{gh} in Line 9 is $\log n$ and n, respectively. Hence, we only need to consider values of φ, such that $\alpha^{\varphi-1} \log n \leq n$. Solving this equation for φ shows that for constant α, there are no more than $O(\log n)$ such values for φ. In other words, there are at most $O(\log n)$ "rings" around x_i, each of which can contain the receivers of at most $C \log n$ blocking links. The total number of potential blocking links in Line 9 is therefore at most in the order of $O(\log^2 n)$. □

Because every blocking link can cause the $\mathbf{check}(\ell_{ij}, \mathbf{L_t})$ subroutine to evaluate to false for a link ℓ_{ij} at most once per phase (when it is scheduled itself), we can combine Lemmas 4.5 and 4.6 and prove the following theorem.

THEOREM 4.7. *The phase scheduler assigns each link $\ell_{ij} \in T(X)$ an intra-phase time slot $0 \leq t(\ell_{ij}) \leq C \log^2 n$ for some constant C.*

The first part of the analysis has shown that the intra phase scheduler is able to quickly schedule all links in $T(X)$. It now remains to show that the scheme is actually valid, i.e., the interference at all intended receivers remains low enough so that all messages arrive. The proof consists of four lemmas that bound the total cumulated interference created by a certain subset of simultaneously transmitting sensor nodes (depending on their τ value). Lemmas 4.8 and 4.9 start by bounding the interference created by all simultaneous transmitters of *shorter links*. In all proofs, $I_{x_i}(x_r)$ denotes the interference power at x_i created by x_r.

LEMMA 4.8. *Consider an arbitrary receiver x_j of a link $\ell_{ij} \in L_t$ scheduled in an intra-phase time slot t. The cumulated interference power $I_{x_j}^1$ at a receiver x_j created by all senders of links $\ell_{gh} \in L_t$ with $\tau_{ij} < \tau_{gh} \leq \tau_{ij} + \log n$ is at most $I_{x_j}^1 \leq \frac{1}{4}(\xi\beta)^{\tau_{ij}-1}$.*

PROOF. We first bound the cumulated interference at x_j created by all simultaneous transmitters having $\tau = \tau_{gh}$ for a specific value of τ_{gh} and then sum up over all possible values in the range $\tau_{ij} < \tau_{gh} \leq \tau_{ij} + \log n$. Let S_{gh} denote this set of senders. It follows from the definition of the **check** subroutine that no interfering sender $x_g \in S_{gh}$ can be within distance $c_1 d_{ij}$ of receiver x_j.

Consider a series of rings $R_1, R_2, \ldots, R_\infty$ around x_j with ring R_λ having inner radius $c_1 \lambda \ell_{ij}$ and outer radius $c_1(\lambda+1)\ell_{ij}$. Consider all senders $x_g \in S_{gh}$ that are located in a ring R_λ. Because all of these senders have the same τ and γ value, they all have the same length (up to a factor of 2), and hence, by Line 2 of the **check** subroutine, they must have a distance of at least $\frac{c_1}{2}\ell_{gh}$ from each other. From this, it follows that disks of radius $\frac{c_1}{4}\ell_{gh}$ around each $x_g \in S_{gh}$ do not overlap. Each such disk has an area of $(c_1^2/16)\ell_{gh}^2 \pi$ and is located entirely inside an extended ring R'_λ of inner radius $c_1\lambda\ell_{ij} - \frac{c_1}{4}\ell_{gh} \geq (\lambda - \frac{1}{4})c_1\ell_{ij}$ and outer radius $(\lambda+1)c_1\ell_{ij} + \frac{c_1}{4}\ell_{gh} \leq (\lambda + \frac{5}{4})c_1\ell_{ij}$. From this, it follows that there can be at most

$$\frac{\left[(\lambda+\frac{5}{4})^2 - (\lambda-\frac{1}{4})^2\right] \ell_{ij}^2 c_1^2 \pi}{\frac{c_1^2}{16}\ell_{gh}^2 \pi} < \frac{72\lambda\ell_{ij}^2}{\ell_{gh}^2}$$

simultaneous transmitters $x_g \in S_{gh}$ in R_λ. Because each of them has a distance of at least $c_1\lambda\ell_{ij}$ from x_j, the cumulated interference from nodes in $S_{gh} \cap R_\lambda$ is at most

$$\begin{aligned} I_{x_j}(S_{gh}\cap R_\lambda) &\leq \frac{(\xi\beta)^{\tau_{gh}}(2\ell_{gh})^\alpha}{(c_1\lambda\ell_{ij})^\alpha} \cdot \frac{72\lambda\ell_{ij}^2}{\ell_{gh}^2} \\ &\leq \frac{72\cdot 2^\alpha}{c_1^\alpha} \cdot \frac{(\xi\beta)^{\tau_{gh}}}{\lambda^{\alpha-1}} \cdot \left(\frac{\ell_{gh}}{\ell_{ij}}\right)^{\alpha-2} \\ &\underset{\text{Lm 4.4}}{\leq} \frac{C'(\xi\beta)^{\tau_{ij}+\delta_{gi}}}{\lambda^{\alpha-1}(\xi\beta)^{\xi\delta_{gi}(\alpha-2)}} \leq \frac{C'(\xi\beta)^{\tau_{ij}-\delta_{gi}}}{\lambda^{\alpha-1}}, \end{aligned}$$

for some constant C'. Summing up over all rings R_λ gives

$$I_{x_j}(S_{gh}) \leq C' \cdot (\xi\beta)^{\tau_{ij}-\delta_{gi}} \sum_{\lambda=1}^{\infty} \frac{1}{\lambda^{\alpha-1}} < C'(\xi\beta)^{\tau_{ij}-\delta_{gi}} \frac{\alpha-1}{\alpha-2}.$$

Summing up over all possible values of τ_{gh} in the range $\tau_{ij} < \tau_{gh} < \tau_{ij} + \log n$ yields a total cumulated interference from simultaneous transmitters in this category of at most $I^1_{x_j} \leq \sum_{\tau_{gh}=\tau_{ij}+1}^{\tau_{ij}+\log n} I_{x_j}(S_{gh}) \leq 2C'(\xi\beta)^{\tau_{ij}-1}\frac{\alpha-1}{\alpha-2}$ because the terms $I_{x_j}(S_{gh})$ form a geometric series for increasing τ_{gh}. Choosing C' large enough concludes the lemma. □

The next lemma considers the interference from all those links ℓ_{gh} that are even shorter compared to to ℓ_{ij}. Bounding the interference from such links is crucial, because they transmit at a high power, relative to their length due to the algorithm's power scaling.

LEMMA 4.9. *Consider an arbitrary receiver x_j of a link $\ell_{ij} \in L_t$ scheduled in an intra-phase time slot t. The cumulated interference $I^2_{x_j}$ at x_j created by all senders of links $\ell_{gh} \in L_t$ with $\tau_{ij} + \log n < \tau_{gh}$ is at most $I^2_{x_j} \leq (\xi\beta)^{\tau_{ij}-1}$.*

PROOF. The interference created by x_g at x_j is given by $\left((\xi\beta)^{\tau_{gh}}\ell_{gh}^\alpha\right)/d_{gj}^\alpha$. Because the **check** subroutine evaluated to **true** at the time ℓ_{gh} was scheduled, we know that $d_{gj} \geq n^{1/\alpha}(\xi\beta)^{\frac{\delta_{gj}+1}{\alpha}}\ell_{gh}$ and hence,

$$I_{x_j}(x_g) \leq \frac{(\xi\beta)^{\tau_{gh}}\ell_{gh}^\alpha}{n\cdot(\xi\beta)^{\delta_{gj}+1}\ell_{gh}^\alpha} = \frac{1}{n}(\xi\beta)^{\tau_{ij}-1}.$$

Since there are at most n nodes in the network, the lemma follows by summing up the interference over all nodes. □

Having bounded the interference from shorter links, the next two lemmas bound the cumulated interference created by simultaneously transmitting senders of *longer* links and roughly *equal-length* links with same τ value. Since the ideas of the respective proofs are already contained in the proof of Lemma 4.8, we defer the proof to the appendix (Lemma 4.10) and omit it (Lemma 4.11), respectively.

LEMMA 4.10. *Consider an arbitrary receiver x_j of a link $\ell_{ij} \in L_t$ scheduled in an intra-phase time slot t. The cumulated interference I_3 at x_j created by all senders of links $\ell_{gh} \in L_t$ with $\tau_{gh} < \tau_{ij}$ is at most $I^3_{x_j} \leq \frac{1}{4}(\xi\beta)^{\tau_{ij}-1}$.*

LEMMA 4.11. *Consider an arbitrary receiver x_j of a link $\ell_{ij} \in L_t$ scheduled in an intra-phase time slot t. The cumulated interference I_4 at x_j created by all senders of links $\ell_{gh} \in L_t$ with $\tau_{gh} = \tau_{ij}$ is at most $I^4_{x_j} \leq \frac{1}{4}(\xi\beta)^{\tau_{ij}-1}$.*

Combining the previous four lemmas and noting that the interference of *every* simultaneously transmitting node is captured in exactly one of these lemmas, we can derive the following theorem.

THEOREM 4.12. *Every message sent over a link $\ell_{ij} \in T(X)$ scheduled in intra-phase time slot t, i.e., $\ell_{ij} \in L_t$, is successfully received by the intended receiver.*

PROOF. Let S_{ij} be the set of nodes that are scheduled to transmit in the same intra-phase time slot as link ℓ_{ij}. Using Lemmas 4.8 through 4.11, we bound the total interference I_{x_j} at the intended receiver x_j as

$$I_{x_j} \leq \sum_{a=1}^{4} I^a_{x_j} \leq \frac{7}{4}(\xi\beta)^{\tau_{ij}-1} = \frac{7}{4\xi}\xi^{\tau_{ij}}\beta^{\tau_{ij}-1}.$$

All that remains to be done is to compute the signal-to-noise-plus-interference ratio at the intended receiver x_j, i.e.,

$$SINR(x_j) \geq \frac{(d_{ij}^\alpha\cdot(\xi\beta)^{\tau_{ij}})/d_{ij}^\alpha}{N + \frac{7}{4\xi}\xi^{\tau_{ij}}\beta^{\tau_{ij}-1}} > \beta.$$

Hence, every intended receiver x_j can correctly decode the packet sent by x_i. □

Theorem 4.7 shows that the number of intra-phase time slots assigned by the procedure is in $O(\log^2 n)$, and Theorem 4.12 that all messages arrive at their receiver correctly. Combining the two theorems therefore proves Theorem 4.7.

Remark 1: Notice that the algorithm as presented in this section assumes that, theoretically, every sensor node has the capability of sending at an arbitrarily high power level. While this is unrealistic in practice, the assumption can be alleviated by using techniques developed in [6].

Remark 2: One of the key techniques employed in the above algorithm is non-linear power scaling of senders transmitting over short links (compare Line 10 of the algorithm). Using a recent result presented in [21], it can be shown that this power scaling is a necessary condition to achieve a high rate in worst-case networks. In particular, it was shown that if nodes transmitted at constant power, or if nodes transmitted at a power proportional to d^α when transmitting over a distance d, then at most a *constant* number of nodes can transmit in parallel in worst-case networks. From this, it immediately follows that even in the physical model, the achievable rate when using either of these two (intuitive) power allocation methods is at most $O(1/n)$.

5. BLOCK CODING

So far, we have not allowed to algorithms to perform *block coding*, i.e., strategies that combine several consecutive function computations that correspond to long blocks of measurements. That is, data aggregation was only allowed between data of the same measurement cycle, not between subsequent cycles. As it turns out, the block coding techniques introduced and studied in [8] can help in significantly reducing the achievable worst-case rate for some perfectly compressible, so-called *type-threshold functions*. Intuitively, type-threshold functions are functions, whose outcome can be computed even if knowing only a fixed number of known arguments (see [8] for a formal definition). In the case of perfectly compressible functions studied in this paper, the max or min are type-threshold functions, whereas avg is not.

The following theorem can be derived by combining Theorem 4.2 with the techniques developed in the proofs of Theorems 4 and 5 of [8], respectively.

THEOREM 5.1. *In the physical model and for perfectly-compressible type-threshold functions, the achievable rate in a worst-case sensor network is $\Omega(1/\log\log n)$ when block coding techniques are allowed.*

6. SCHEDULING COMPLEXITY

The rate R of sensor network quantifies the *maximum amount of information* that can periodically be transmitted to the sink. In recent literature on wireless networks, the *scheduling complexity* of wireless networks [14, 21, 23] has been proposed as a complementing measure for characterizing the possibilities and limitations of communication in shared wireless media. Intuitively, the scheduling complexity of wireless networks describes the *minimum amount of time* required to successfully schedule a set of communication requests. Formally, it is defined as follows [21]:

Definition 6.1. *Let Γ be the set of communication requests (s_i, t_i) between two nodes. The* scheduling complexity *$T(\Gamma)$ of Γ is the minimal number of time slots T such that all requests in Γ can simultaneously be scheduled.*

The scheduling complexity therefore reflects how fast all requests in Γ can theoretically be satisfied (that is, when scheduled by an optimal MAC-layer protocol).

The results of this paper give raise to a number of novel results that bound the scheduling complexity in wireless networks. In particular, the following result is implicit in the proof of Section 4.

Theorem 6.1. *The* scheduling complexity of connectivity *[21] is bounded by $O(\log^2 n)$: In every network, a strongly connected topology can be scheduled using $O(\log^2 n)$ time slots.*

Notice that this improves on the best previously known bound on the scheduling complexity of connectivity by a logarithmic factor [23]. By simultaneously improving results on the capacity of sensor networks as well as the scheduling complexity of wireless networks, the algorithm makes a first step towards gaining a unified understanding of these two important concepts in wireless networks.

7. RELATED WORK

The study of capacity in wireless networks was initiated in the seminal work of Gupta and Kumar in [12]. Ever since, there has been a flurry of new results that characterize the capacity of different wireless networks in a variety of models. The first work to derive capacity bounds explicitly for the data aggregation problem in sensor networks is by Marco et al. [19]. In this work, the capability of large-scale sensor networks to measure and transport a two-dimensional stationary random field using sensors is investigated. Giridhar and Kumar in [8] study the more general problem of computing and communicating symmetric functions of the sensor measurements. They show that in a random planar multihop network with n nodes, the maximum rate for computing divisible functions—a subset of symmetric functions—is $\Theta(1/\log n)$. Using the block-coding technique, they further show that in networks in which nodes are deployed uniformly at random, so-called type-threshold functions can be computed at a rate of $\Theta(1/\log\log n)$, which is the same rate as we achieve with block coding even in worst-case networks.

More recently, Ying et al. have studied in [25] the problem of minimizing the total transmission energy used by sensor nodes when computing a symmetric function, subject to the constraint that this computation is correct with high probability. In [1], Barton and Zheng prove that no protocol can achieve a better rate than $\Theta(\log n/n)$ in collocated sensor network in the physical model. They improve on the rate of $\Theta(1/n)$ shown in [8] by employing cooperative time-reversal communication techniques. Further work on data aggregation/capacity in sensor networks includes [20, 2, 11, 4, 7].

All of the above works derive capacity bounds in either collocated networks (single-hop) or in multi-hop networks in which nodes are assumed to be randomly placed. In contrast, capacity problems in *worst-case networks* have received considerably less attention. In [18, 17], algorithmic aspects of wireless capacity are considered, however, without deriving explicit scaling laws that describe the achievable capacity in terms of the number of nodes. Moreover, the works of [18, 17] are based on the protocol model of wireless communication or simplistic graph models. Given the exponential gap between these models and the physical model proven in this paper, these models do not adequately capture the achievable *worst-case capacity* in wireless networks. There have been numerous proposals for efficient data gathering algorithms and protocols in sensor networks, many of which are graph-based and focus on the important aspect of energy-efficiency [10, 15, 3, 24].

The scheduling complexity of wireless networks has been introduced and first studied in [21]. Subsequent papers have improved and generalized the results in [21] and have applied the concepts to wide-band networks [14] as well as to topology control [23]. Earlier work on scheduling with power control includes for instance [5].

Finally, notice that our results have implications on the *design* of efficient data gathering protocols. In particular, our results show that any data gathering protocol that achieves a high rate must make explicit use of SINR properties: All protocols that operate using uniform power assignment (or even linear power assignment) must inherently perform suboptimally in certain networks. This, in turn, gives clear design-guidelines for protocol designers.

8. CONCLUSIONS

In this paper, we have initiated the study of *worst-case capacity* in wireless sensor networks. The achievable rate of $\Omega(1/\log^2 n)$ in worst-case sensor networks shows that in the physical model, the *price of worst-case node placement* is small, at most a logarithmic factor, whereas in the protocol model, it is significantly higher. In particular, by making use of specific physical SINR characteristics, the physical model allows for rates that exceed the rates achievable in the protocol model by an exponential factor. This sheds new light into the fundamental relationship between these two important models in wireless communication: Whereas in *randomly deployed networks*, the capacity of wireless networks has been shown to be robust with regard to the two models, the same is not the case when it comes to worst-case capacity. From a practical perspective, this implies that every sensor network data gathering protocol which adheres to the protocol model (for instance by assigning constant transmission power to nodes, or by using schedules that are based on colorings of an interference graph) is inherently suboptimal in worst-case networks.

It is interesting to point out that our results are positive in nature. While the seminal capacity result by Gupta and Kumar [12] has often been regarded as an essentially negative result that limits the possible scalability of wireless networks, our result shows that in sensor networks—even if node placement is worst-case—a high rate can be maintained. The theoretically achievable worst-case rate in sensor networks remains high even as network size grows.

9. REFERENCES

[1] R. J. Barton and R. Zheng. Order-Optimal Data Aggregation in Wireless Sensor Networks Using Cooperative Time-Reversal Communication. In *Proceedings of the 40^{th} Conference on Information Sciences and Systems (CISS)*, pages 1050–1055, 2006.

[2] A. Chakrabarti, A. Sabharwal, and B. Aazhang. Multi-Hop Communication is Order-Optimal for Homogeneous Sensor Networks. In *Proc. of the 3nd International Symposium on Information Processing in Sensor Networks (IPSN)*, 2004.

[3] R. Cristescu, B. Beferull-Lozano, and M. Vetterli. On Network Correlated Data Gathering. In *Proceedings of the 23^{th} Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2004.

[4] E. J. Duarte-Melo and M. Liu. Data-Gathering Wireless Sensor Networks: Organization and Capacity. *Computer Networks*, 43, 2003.

[5] T. ElBatt and A. Ephremides. Joint Scheduling and Power Control for Wireless Ad-hoc Networks. In *Proceedings of the 21^{th} Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2002.

[6] M. Fussen, R. Wattenhofer, and A. Zollinger. Interference Arises at the Receiver. In *Proc. of the International Conference on Wireless Networks, Communications, and Mobile Computing (WirelessCom)*, June 2005.

[7] J. Gao, L. Guibas, J. Hershberger, and L. Zhang. Fractionally Cascaded Information in a Sensor Network. In *Proc. of the 3nd International Symposium on Information Processing in Sensor Networks (IPSN)*, 2004.

[8] A. Giridhar and P. R. Kumar. Computing and Communicating Functions over Sensor Networks. *IEEE Journal on Selected Areas in Communications*, 23(4):755–764, 2005.

[9] A. Giridhar and P. R. Kumar. Towards a Theory of In-Network Computation in Wireless Sensor Networks. *IEEE Communications Magazine*, 44(4), 2006.

[10] A. Goel and D. Estrin. Simultaneous Optimization for Concave Costs: Single Sink Aggregation or Single Source Buy-at Bulk. In *Proc. of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2003.

[11] P. K. Gopala and H. E. Gamal". On the Scaling Laws of Multi-Modal Wireless Sensor Networks. In *Proceedings of the 23^{rd} Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2004.

[12] P. Gupta and P. R. Kumar. The Capacity of Wireless Networks. *IEEE Trans. Information Theory*, 46(2):388–404, 2000.

[13] M. Haenggi. Routing in Ad Hoc Networks - A Wireless Perspective. In *Proc. of the 1st International Conference on Broadband Networks (BroadNets)*, 2004.

[14] Q.-S. Hua and F. C. M. Lau. The Scheduling and Energy Complexity of Strong Connectivity in Ultra-Wideband Networks. In *Proc. of the 9th ACM Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pages 282–290, 2006.

[15] L. Jia, G. Lin, G. Noubir, R. Rajaraman, and R. Sundaram. Universal Approximations for TSP, Steiner Tree and Set Cover. In *Proc. of the 37^{th} ACM Symposium on Theory of Computing (STOC)*, 2005.

[16] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft. XOR in The Air: Practical Wireless Network Coding. In *Proc. of ACM SIGCOMM*, Pisa, Italy, 2006.

[17] M. Kodialam and T. Nandagopal. Characterizing Achievable Rates in Multi-Hop Wireless Networks: The Joint Routing and Scheduling Problem. In *Proceedings of the 9^{th} Annual International Conference on Mobile Computing and Networking (MOBICOM)*, 2003.

[18] V. S. A. Kumar, M. V. Marathe, S. Parthasarathy, and A. Srinivasan. Algorithmic Aspects of Capacity in Wireless Networks. In *Proc. International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS)*, pages 133–144, 2005.

[19] D. Marco, E. J. Duarte-Melo, M. Liu, and D. L. Neuhoff. On the Many-to-One Transport Capacity of a Dense Wireless Sensor Network and the Compressibility of its Data. In *Proc. of the 2nd International Workshop on Information Processing in Sensor Networks (IPSN)*, 2003.

[20] U. Mitra and A. Sabharwal. Complexity Constrained Sensor Networks: Achievable Rates for Two Relay Networks and Generalizations. In *Proc. of the 3nd International Symposium on Information Processing in Sensor Networks (IPSN)*, 2004.

[21] T. Moscibroda and R. Wattenhofer. The Complexity of Connectivity in Wireless Networks. In *Proc. of the 25^{th} IEEE INFOCOM*, 2006.

[22] T. Moscibroda, R. Wattenhofer, and Y. Weber. Protocol Design Beyond Graph-based Models. In *Proceedings of the 5^{th} ACM SIGCOMM Workshop on Hot Topics in Networks (HotNets)*, 2006.

[23] T. Moscibroda, R. Wattenhofer, and A. Zollinger. Topology Control meets SINR: The Scheduling Complexity of Arbitrary Topologies. In *Proc. of the 7^{th} ACM Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, 2006.

[24] P. von Rickenbach and R. Wattenhofer. Gathering Correlated Data in Sensor Networks. In *ACM Join Workshop on Foundations of Mobile Computing (DIALM-POMC)*, 2004.

[25] L. Ying, R. Srikant, and G. E. Dullerud. Distributed Symmetric Function Computation in Noisy Wireless Sensor Networks with Binary Data. In *Proc. of the 4th Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOPT)*, 2006.

Appendix: Proof of Lemma 4.10

PROOF. The proof is similar to the proof of Lemma 4.8. We know by Line 4 of the subroutine that $d_{gj} \geq d_{gh}$. We first bound the total amount of interference created at x_j from all links having $\tau = \tau_{gh}$ for a specific value of τ_{gh}. Summing up over all $1 \leq \tau_{gh} < \tau_{ij}$ will conclude the proof.

Consider rings $R_1, R_2, \ldots, R_\infty$ around x_j with ring R_λ having inner and outer radius $(2\lambda-1)\ell_{gh}$ and $2\lambda\ell_{gh}$, respectively. Since we consider only links having $\tau = \tau_{gh}$, we know by Line 2 of the subroutine that no two simultaneous senders can be too close to each other. In particular, it holds that disks of radius $\frac{c_1}{4}\ell_{gh}\pi$ around each sender do not overlap. Furthermore, if the sender x_g of such a link is located in ring R_λ, its corresponding disk of radius $\frac{c_1}{4}\ell_{gh}\pi$ is entirely contained in the ring R'_λ of inner and outer radius $(2\lambda - \frac{5}{4})\ell_{gh}$ and $(2\lambda + \frac{1}{4})\ell_{gh}$, respectively. This extended ring R'_λ has an area of $(6\lambda - \frac{3}{2})\ell_{gh}^2\pi$. By the standard area argument, it follows that the total interference from senders with $\tau = \tau_{gh}$ in this ring is at most

$$\begin{aligned} I_{x_j}(S_{gh} \cap R_\lambda) &\leq \frac{(\xi\beta)^{\tau_{gh}}(2\ell_{gh})^\alpha}{(2\lambda-1)^\alpha \ell_{gh}^\alpha} \cdot \frac{16(6\lambda-\frac{3}{2})}{c_2^2} \\ &< \frac{2^{\alpha+4}(\xi\beta)^{\tau_{gh}}}{c_1^2(2\lambda-1)^{\alpha-1}} \leq \frac{2^{\alpha+4}(\xi\beta)^{\tau_{gh}}}{c_1^2\lambda^{\alpha-1}}. \end{aligned}$$

Again, we can sum up over all rings to obtain the total amount of interference that simultaneous transmitters with τ_{gh} can cause, and then sum up over all possible values of $1 \leq \tau_{gh} < \tau_{ij}$. Specifically, the total amount of interference at x_j from these nodes is at most

$$I_{x_j}^3 \;\leq\; \sum_{\tau_{gh}=1}^{\tau_{ij}-1} \sum_{\lambda=1}^{\infty} \frac{2^{\alpha+4}(\xi\beta)^{\tau_{gh}}}{c_1^2\lambda^{\alpha-1}} \;\leq\; \frac{2^{\alpha+5}(\xi\beta)^{\tau_{ij}}}{c_1^2\lambda^{\alpha-1}} \frac{\alpha-1}{\alpha-2}, \quad (4)$$

which concludes the proof when setting the constant c_1 to a large enough value. □