Distributed Approximation Algorithms for Maximum Link Scheduling and Local Broadcasting in the Physical Interference Model

Guanhong Pei* and Anil Kumar S. Vullikanti†
*Dept. of Electrical and Computer Engineering and Virginia Bioinformatics Institute, Virginia Tech, Blacksburg, VA, USA
†Dept. of Computer Science and Virginia Bioinformatics Institute, Virginia Tech, Blacksburg, VA, USA

***Abstract*—In this paper, we develop the first rigorous distributed algorithm for link scheduling in the SINR model under any length-monotone sub-linear power assignments. Our algorithms give constant factor approximation guarantees, matching the bounds of the sequential algorithms for these problems, with provable bounds on the running time in terms of the graph topology. We also study a related and fundamental problem of local broadcasting for uniform power levels, and obtain similar bounds. These problems are much more challenging in the SINR model than in the more standard graph based interference models, because of the non-locality of the SINR model. Our algorithms are randomized and crucially rely on physical carrier sensing for the distributed communication steps. We find that the specific wireless device capability of duplex/half-duplex communication significantly impacts the performance. Our main technique involves the distributed computation of affectance and a construct called a ruling, which are likely to be useful in other scheduling problems in the SINR model. We also study the empirical performance of our algorithms, and find that the performance depends on the topology, and the approximation ratio is very close to the best sequential algorithm.**

I. Introduction

Node and link scheduling problems are among the most fundamental problems in wireless networks. These problems have been studied extensively under various interference models, which model the conflicts between links— most of these are based on the idea of "conflict graphs" [1]. The decision version of MaxLSP is NP-Complete under many models, and constant factor approximation algorithms are known for many interference models [1]. In recent years, a more realistic interference model based on SINR constraints (henceforth referred to as the SINR model) [2] has gained a lot of interest. This model is much more challenging to analyze than models based on conflict graphs [3].

We study two basic link and node scheduling problems in the SINR model (defined formally in Section III). The first is the *Maximum Link Scheduling* problem (MaxLSP), in which the goal is to select the largest subset from a given set L of links that can be scheduled simultaneously with SINR constraints. The second is the *Maximum Local Broadcast* problem (MaxLBP), in which the goal is to select a subset of nodes V' from a given set V, such that the SINR constraints are satisfied for all possible receivers within a given range of each node in V' when all nodes in V' broadcast — this generalizes MaxLSP, and is an "independent set" version of the *local broadcasting* problem of [4] (but with important differences, discussed in Section II).

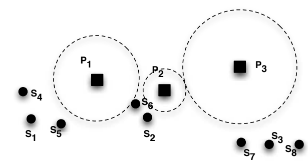


Fig. 1. Scheduling constraints in cognitive networks: the nodes P_1, P_2, P_3 represent primary users, whereas nodes $S_1, \ldots, S_8$ represent secondary users. The disks around the primary users represent regions where the SINR constraints need to be higher than some threshold.

Since link scheduling is a common subroutine in many other problems, distributed algorithms with low complexity are crucial. Efficient (*e.g.*, polylogarithmic time) distributed algorithms have been developed for many graph based interference models (generically referred to as the "Radio Broadcast Network (RBN)" model) for fundamental problems, including independent sets, coloring and dominating set, e.g., [5], [6]. In the RBN model, a receiver faces a collision if transmissions from multiple senders reach it simultaneously. Algorithms in this model rely on having spatial separation between senders to ensure interference free communication; however, schedules in the RBN model are not feasible with respect to SINR constraints,* and a different model of distributed computing is needed for handling SINR constraints. This is especially important in emerging technologies such as cognitive networks, where secondary (unlicensed) users need to be scheduled with given constraints that do not disrupt primary (licensed) users [8]. Primary user constraints can be modeled by specifying the regions around them within which the SINR needs to be lower than some threshold— this is shown by the disks in Figure 1. The set V represents secondary users, which have to sense the spectrum and use it opportunistically, which motivates a non-uniform power assignment. In such settings, the MaxLSP and MaxLBP problems necessarily need to be solved in a distributed manner. Indeed, the gains from cognitive networks are unlikely to be realized unless these basic scheduling problems can be implemented with low complexity.

MaxLSP and MaxLBP are algorithmically very challenging problems, and even centralized algorithms for these problems are much harder in the SINR model, than in the graph based interference model; recent work by [9], [10], [11], [12], [13] gives constant factor approximation algorithms for various instances of MaxLSP in the SINR model. The

*It is easy to construct instances where with $\Omega(n)$ gap between the minimum schedule lengths in the graph based and SINR models [7].

978-1-4673-5946-7/13/$31.00 ©2013 IEEE

centralized algorithms of [9], [10], [11], [13] are based on a greedy ordering of the links, which requires estimating the "affectance," (which, informally, is a measure of interference), at each stage (this is discussed formally later in Section III) — this is one of the challenges in distributed solutions to MAXLSP. We note that efficient time distributed algorithms for MAXLSP are known for the special case of uniform power levels, when each link ℓ uses the same transmission power level [14]. However, no distributed algorithm is known for the case of non-uniform power levels. Further, no results are known for the MAXLBP in the SINR model, to the best of our knowledge.

Our contributions. In this paper, we develop the first distributed constant factor approximation algorithms for MAXLSP and MAXLBP problems, with provable bounds on the running time and performance. For the MAXLSP problem, we consider non-uniform length monotone sub-linear power levels (defined in Section III), whereas for MAXLBP, we consider uniform power levels. We find that the performance bounds of our algorithms depend crucially on whether or not the devices have the capability for duplex/half-duplex communication (*i.e.*, whether nodes can transmit and receive simultaneously with the same frequency). We summarize some of the key aspects of the results and main challenges below.

(1) For MAXLSP with length-monotone sub-linear power levels, we design a distributed algorithm that runs in time $O(g(L)\rho \log n)$, with high probability, for duplex communication in the SINR model; here $g(L)$ denotes the "length diversity", which refers to the number of different classes of links possible, and ρ denotes the "local density", a parameter depending on the topology — informally, ρ is the maximum number of links in L of length in the range $[d/2, d]$ such that their sender nodes fall in the same disk of any radius d. In practice, and in instances such as random distributions, this parameter is at most $O(\log n)$. When the communication model is the more restrictive but more realistic half-duplex model, our algorithm takes time $O(g(L)\rho \log^2 n)$, *w.h.p.*

(2) For the MAXLBP problem with uniform power assignment, but non-uniform ranges, we design a distributed constant factor approximation algorithm with running time $O(g(V) \log^2 n)$ with high probability, in a duplex model of communication, here $g(V)$ denotes the "range diversity", defined analogously to the link diversity (see Section III). Our approach involves adapting the approach of [10], [13], [14]. Further, for a half-duplex model of communication, the running time of our algorithm is $O(g(V) \log^3 n)$, *w.h.p.*

(3) We study the performance of our algorithms through simulations. We find that the approximation factor for our distributed algorithms for MAXLSP is within a factor of 2. Thus, the distributed ruling step, which selects a subset of links from the same length class, does not significantly affect the quality of the solution. Further, we find that the topology has a significant impact on the performance.

Key technical contributions and focus. The key aspect of our distributed algorithms is the distributed implementation of the affectance computation step. We adapt the notion of "ruling" developed in [14] to compute the affectance bounds for all links in a given length class. Our algorithms crucially use physical carrier sensing and the distributed decisions are made based on the Received Signal Strength Indication (RSSI) — this approach was used by [14], [15] for distributed algorithms for constant density dominating set construction and MAXLSP for uniform power levels. Further, our algorithms use constant size messages, and all the steps can be implemented within the model without any additional capabilities or assumptions (e.g., such as those made in [16]).

The focus of our paper is theoretical, and our main contribution is the development of distributed techniques in the SINR model, and a formal study of the impact of technological aspects of wireless capabilities on distributed algorithms. As our results show, developing algorithms in this model is significantly more challenging, and using specific cognitive network capabilities can help in improving the efficiency.

Organization. We discuss related work in Section II and the network model and relevant definitions in Section III. We present the algorithm for MAXLSP and the restricted ruling computation in Sections IV and V respectively, and then the algorithm for MAXLBP in Section VI. We show the empirical performance of our algorithm in Section VII and conclude in Section VIII. Some of the proofs are omitted due to space constraints; the complete version is [17] (available online).

II. RELATED WORK

Distributed algorithms are known for node and link scheduling (and many related problems) in the graph-based model [5], [6], [18], [19], [20], [21]. In recent work, Afek et al. [20] develop a distributed algorithm for MAXLSP under graph-based interference model, with a constant approximation factor.

Link scheduling in the SINR model is considerably harder than in graph based models. Several papers developed $O(g(L))$-approximations for MAXLSP, e.g., [2], [22], which have been improved to constant factor approximations by [9], [10], [11] for specific power assignments. Some of these papers use "capacity" [9], [10] to refer to the maximum link scheduling; however, we prefer to avoid the term capacity in order to avoid confusion with the total throughput in a network, which has been traditionally referred to as the capacity (e.g., [23]). Recently, Halldórsson and Mitra [13] extend the $O(1)$ approx. ratio to any setting with a fixed length-monotone, sub-linear power assignment (which includes uniform, mean and linear power assignments). This has been improved by Kesselheim [12], who developed the first $O(1)$-algorithm for the maximum independent set problem and an $O(\log m)$-algorithm for the minimum length scheduling problem. Wan et al. [24] propose algorithms for a set of problems including maximum link scheduling and minimum length scheduling

Most of the above algorithms for scheduling in the SINR model are centralized and it is not clear how to implement them in a distributed manner efficiently. The closest results to ours are [14], [16], [25]. The results of [16], [25] use a game theoretic approach, but lead to much higher running times than ours. The result of [14] gives a fast distributed algorithm for MAXLSP under the uniform power level assumption. It develops the notion of ruling, which we build on here for non-uniform power assignment.

A related set of problems of the MAXLSP and MAXLBP involves "coloring" in which the goal is to schedule *all* the transmission/broadcast requests in the smallest number of time slots, whereas in MAXLSP and MAXLBP we target to pick the largest feasible subset. Recently, for the "coloring" version of MAXLSP, Kesselheim and Vöcking [26] propose an $O(\log^2 m)$-approximate distributed algorithm for any fixed length-monotone and sub-linear power assignment. Its analysis has been improved to $O(\log m)$ by Halldórsson and Mitra [27], who also prove that if all links uses the same randomized strategy, there exists a lower-bound of $\Omega(\log m)$ on the running time. It is not clear how to use these results for the "coloring" problem to get a constant factor approximation for MAXLSP, in which the senders and receivers of all links know their status. Goussevskaia et al. [4] study the "coloring" version of MAXLBP (*i.e.*, to complete all the broadcast requests in shortest time). They design a sublinear-time random-access based distributed algorithm under the uniform power assignment and uniform broadcast range setting, depending on the information about the local topology. Nodes do not exchange information to determine who was successful. It is not clear how to adapt their algorithm for MAXLBP, especially with non-uniform ranges and the requirement that all nodes know whether they have been selected or not.

III. Preliminaries and Definitions

TABLE I. NOTATION.

G	network graph	$d(u,v)$	dist. of u and v
V	set of nodes	L	set of links
n	#nodes	$g(L)$	link diversity
m	#links	$OPT()$	optimum instance
α	path-loss exponent	$x(l)$	sender of link l
β	SINR threshold	$r(l)$	receiver of link l
N	background noise	$d(l)$	length of link l
A	affectance	SP	sensed power

Table I summarizes the notation symbols we use frequently. We let V denote a set of tranceivers (henceforth, referred to as nodes) in the Euclidean plane. We assume L is a set of links with end-points in V, which form the set of communication requests for the maximum link scheduling problem, and $|L| = m$. Links are directed, and for link $l = \big(x(l), r(l)\big)$, $x(l)$ and $r(l)$ denote the transmitter (or sender) and receiver respectively. For a link set L', let $X(L')$ denote the set of senders of links in L', and likewise $R(L')$ the set of receivers. Let $d(u,v)$ denote the Euclidean distance between nodes u,v. For link l, let $d(l) = d(x(l), r(l))$ denote its link length. For links l,l', let $d(l',l) = d(x(l'), r(l))$. Let d_{min} and d_{max} denote the smallest and the largest transmission link lengths respectively. Let $B(v,d)$ denote the ball centered at node v with a radius of d. Each sender $x(l)$ uses power $P(l)$ for transmission on l; we assume commonly used path loss models [2], [22], where a transmission on link l is possible only if:

$$\frac{\frac{P(l)}{d^{\alpha}(l)}}{N(1+\phi)} \geq \beta, \tag{1}$$

where $\alpha > 2$ is the "path-loss exponent", $\beta > 1$ is the minimum SINR required for successful reception, N is the background noise, and $\phi > 0$ is a constant (note that α, β, ϕ and N are all constants).

We partition the set of transmission links into non-overlapping link classes. We define *link diversity* $g(L) = \lceil \log_2 \frac{d_{max}}{d_{min}} \rceil$. Partition $L = \{L_i\}, i = 1, 2, \ldots, g(L)$, where each $L_i = \{l \mid 2^{i-1} d_{min} \leq d(l) < 2^i d_{min}\}$ is the set of links of roughly similar lengths. Let $d_i = 2^i d_{min}$, such that d_i is an upperbound of link lengths in L_i; and $\forall i, \forall l \in L_i$, we define $\hat{d}(l) = d_i$. In a distributed environment, nodes use their shared estimates of minimum and maximum possible link length to replace d_{min} and d_{max}, as stated in the previous section. Therefore, $g(L)$ in most cases $\leq \log 10^6$ and remains a constant; further, as discussed earlier, each link can compute which link class it belongs to. We define the *local density* denoted by ρ as the maximum number of links in the same link class L_i such that their sender nodes fall in the same disk of radius d_i. The *reverse link* of a link l, denoted by $\overleftarrow{l}$, is the same link with transmission direction inverted. For a link set L', We use $\overleftarrow{L'}$ to denote the set of reverse links of L'. Analogously to the link diversity $g(L)$, we define range diversity $g(V)$ of a set V of nodes in MAXLBP, as the $g(V) = \lceil \log \frac{\max_{v\in V} d(v)}{\min_{v \in V} d(v)} \rceil$, where $d(v)$ denotes the requested local broadcast range for node v.

Wireless Interference. We use physical interference model based on geometric SINR constraints (henceforth referred to as the SINR model), where a subset $L' \subseteq L$ of links can make successful transmission simultaneously if and only if the following condition holds for each link $l \in L'$:

$$\frac{\frac{P(l)}{d^{\alpha}(l)}}{\sum_{l' \in L' \setminus \{l\}} \frac{P(l')}{d^{\alpha}(l',l)} + N} \geq \beta. \tag{2}$$

Such a set L' is said to be *independent* in the context.

The Maximum Link Scheduling Problem (MAXLSP). Given a set of communication requests (links) L, and a power level $P(l)$ for each link $l \in L$, the goal is to find a largest independent subset (that can be scheduled simultaneously in the SINR model). We assume the power level assignment is length-monotone and sub-linear, which means that for any links l, l', $d(l') \geq d(l)$ implies $P(l') \geq P(l)$ and $\frac{P(l')}{d^{\alpha}(l')} \leq \frac{P(l)}{d^{\alpha}(l)}$. In this paper, we use $OPT^P(L)$ to denote an optimum solution for MAXLSP, i.e., the cardinality of the largest such independent set for the given length-monotone and sub-linear power assignment; we drop the super-script P to denote the underlying power assignment, whenever it is clear from the context. As discussed earlier, computing $OPT(L)$ is NP-hard, and we focus on approximation algorithms. We say an algorithm gives a C-approximation factor if it constructs an independent link set $L' \subseteq L$ with $|L'| \geq |OPT(L)|/C$.

The Maximum Local Broadcast Problem (MAXLBP). Given a set V of broadcast nodes, and a range $d(v)$ for each node $v \in V$, the goal is to select a subset $V' \subset V$, so that when all the nodes in V' broadcast simultaneously, for each node $v \in V'$, any potential receiver within distance $d(v)$ from v can receive v's broadcast message successfully. That is, we need to keep the SINR above β for any location within the range of each selected broadcast node in V'. In this problem, we assume each node uses a uniform power level $P(v) = P$. We use $OPT_{\text{MAXLBP}}(V)$ to denote the optimal solution for the instance of MAXLBP; we drop the subscript whenever it is clear from the context. Here again, we study approximation algorithms, and the goal is to approximate $OPT_{\text{MAXLBP}}(V)$.

Distributed Computing Model in the SINR-based Model. Traditionally, distributed algorithms for wireless networks have been studied in the radio broadcast model [5], [6], [19] and its variants. The SINR based computing model is relatively recent, and has not been studied that extensively. Therefore, we summarize the main aspects and assumptions underlying this model: (1) The network is synchronized and for simplicity we assume all slots have the same length. (2) All nodes have a common estimate of m, the number of links, within a polynomial factor; (3) For each link $l \in L$, $x(l)$ and $r(l)$ have an estimate of $d(l)$, but they do not need to know the coordinates or the direction in which the link is oriented; (4) All nodes share a common estimate of d_{min} and d_{max}, the minimum and maximum possible link lengths (5) We assume nodes have physical carrier sensing capability and can detect if the sensed signal exceeds a threshold. As discussed in [15], this can be done using the RSSI measurement possible through the Clear Channel Assessment capability in the 802.11 standard. Given a threshold $Thres$, we assume that a node is able to detect if the sensed power strength is $\geq Thres$.

Sensed Power-strength and Affectance. For ease of analysis based on links, we define *affectance* as that in [10], [13]: the affectance of link l caused by link l' is defined as $A(l',l) = c_l \dfrac{P(l')/d^\alpha(x(l'),r(l))}{P(l)/d^\alpha(l)}$, where $c_l = \beta/(1 - \beta N d^\alpha(l)/P(l))$. For a set L' of links, this is extended to $A(L',l) = \sum_{l' \in L'} A(l',l)$ and $A(l,L') = \sum_{l' \in L'} A(l,l')$. It can be verified that Inequality (2) is equivalent to $A(L' \setminus \{l\}, l) \leq 1$, signifying the success of data transmission on l. Following [10], [13], we say that a set L' of links is a δ-signal set if and only if $A(L' - \{l\}, l) \leq 1/\delta$ for each $l \in L'$.

To simplify the analysis based on nodes, we define *sensed power-strength* $SP(w,v)$, as the signal power that node v receives when only w is transmitting (which includes background noise); that is, $SP(w,v) = P(w)/d^\alpha(w,v) + N$. Likewise, we have SP from a node set W: $SP(W,v) = \sum_{w \in W} P(w)/d^\alpha(w,v) + N$.

Half/full Duplex Communication. Wireless radios are generally considered *half duplex*, *i.e.*, with a single radio they can either transmit or receive/sense but not both at the same time. *Full duplex* radios, which are becoming reality, enable devices to transmit and receive simultaneously in the same frequency.

IV. Distributed Algorithm for MaxLSP

In this section, we present the distributed algorithm for MaxLSP under a length-monotone sublinear power assignment. In order to simplify the discussion, we first explain the centralized algorithm of [13] and the main ideas of its analysis. We then discuss the distributed implementation.

A. The Centralized Algorithm

Brief Description. The centralized algorithm [13] for MaxLSP forms the basis for the distributed algorithm. The algorithm processes links in non-decreasing order of length. Recall the notion of affectance from Section III. Let $L = \{l_1, \ldots, l_n\}$ be the initial set of links in non-decreasing order of length. Let S be the set of links already chosen (which is empty initially). For link l: if $A(S,l) + A(l,S) < 1/2$, add l to S. After considering all links, the set $S' = \{l \in S : A(S,l) \leq 1\}$ is selected as the solution after verifying the affectance.

Proof ideas. By construction, the solution S' is feasible. The main idea in proving the approximation factor of the solution is the following "Red-Blue Lemma" from [13]. For a set Y of links, define $Y^+(l) = \{l' \in Y : d(l') \geq d(l)\}$ and $Y^-(l) = \{l' \in Y : d(l') \leq d(l)\}$.

Lemma IV.1. *(Red-Blue Lemma [13]) Let $RED, BLUE$ be disjoint link sets. If $|BLUE| > 4|RED|$ and $BLUE$ is a 3^α-signal set, then $\exists$ link $l \in BLUE$ s.t. $A(RED^-(l), l) + A(l, RED^-(l)) \leq 3^\alpha(A(BLUE, l) + A(l, BLUE))$ under any length-monotone sublinear power assignment.*

Implementing this algorithm in a distributed manner in the SINR model requires determining the quantities $A(S,l)$ and $A(l,S)$ for multiple links l simultaneously. Of these, estimating $A(S,l)$ is simpler, because it can be done by estimating the total signal strength of the transmissions by all $x(l')$ for $l' \in S$ at $r(l)$. However, estimating $A(l,S)$ requires additional ideas. Further, in order to make the distributed implementation efficient, these estimates need to be done simultaneously for many links, which have to decide to be part of the final solution (or not). In order to facilitate this, the links l within a link class L_i work in a phase to determine $A(\cup_{j<i} S_j, l) + A(l, \cup_{j<i} S_j)$, where S_j denotes the links from L_j that have been selected in earlier phases. In order to ensure that the links selected from L_i satisfy the affectance constraints, they need to be spatially separated— the ruling construction in Section IV-B helps satisfy such properties in a distributed manner.

B. Additional Definitions: Cover and Ruling

Let W, W' denote two node sets. We say a node u is ω-*covered* by W', if and only if $\exists u' \in W', d(u,u') \leq \omega$; based on that, we say W is ω-*covered* by W', or equivalently W' ω-*covers* W, if and only if every node in W is ω-covered by W'. An (ω_1, ω_2)-*ruling* (where $\omega_1 < \omega_2$) of W, informally a spatially separated node cover, is a node set denoted by $R_{\omega_1,\omega_2}(W)$, such that

(1) $R_{\omega_1,\omega_2}(W) \subseteq W$;
(2) all the nodes in $R_{\omega_1,\omega_2}(W)$ are at least ω_1-separated; that is, $\forall u, u' \in R_{\omega_1,\omega_2}(W)$, $d(u,u') \geq \omega_1$; and
(3) W is ω_2-covered by $R_{\omega_1,\omega_2}(W)$.

Here, we have adopted a generalized definition of ruling by considering Euclidean distance rather than graph distance differently from that first defined in [28]. Figure 2 gives an example to illustrate these notions. The concept of ruling has a vital role in our algorithm: it is used for choosing a set of spatially separated links and removing the nearby links of the chosen links.

Let X be an arbitrary subset of senders and R be an arbitrary subset of receivers of links in the same link class. We define a R-*restricted* (ω_1, ω_2)-*ruling* of X as $R_{\omega_1,\omega_2}(X')$, where $X' = \{v \in X : v\text{'s corresponding receiver node is in } R\}$.

C. The Distributed Algorithm

Algorithm 1 describes the distributed scheduling algorithm for MaxLSP with any length-monotone sublinear power assignment. We use the following constants in the algorithm

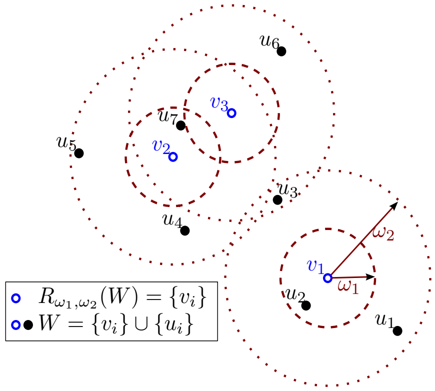


Fig. 2. Example of an (ω_1, ω_2)-ruling: $W = \{v_i\} \cup \{u_i\}$ is the set of all dots (open and dark), while $R_{\omega_1,\omega_2}(W) = \{v_i\}$ which is the set of all the open dots denotes a (ω_1, ω_2)-ruling of W. Note that all the nodes in W are ω_2-covered by $R_{\omega_1,\omega_2}(W)$, while all the nodes in $R_{\omega_1,\omega_2}(W)$ are ω_1 away from each other.

$\gamma_1 = \left(\frac{36\beta}{1-\psi}\frac{\alpha-1}{(\alpha-2)}\frac{1+\phi}{\phi}\right)^{1/\alpha} + 2$ and γ_2 as an arbitrary constant $> \gamma_1$, where α, β, ϕ are constants described in Inequality (1) and ψ is a constant that can take any value from $(0, 1)$. The algorithm consists of two steps:

(Step 1) selecting links according to affectance and ruling constraints (Lines 2-2), and
(Step 2) verifying affectance constraints (Lines 19-19).

In each step, it sweeps through the link classes in $g(L)$ phases; the $g(L)$ phased approach has been used in [29] for distributed throughput maximization. In the ith phase of Step 1, where $i \in [1, g(L)]$, it selects a subset of links S_i from L_i. At Lines 3-14, both the sender and the receiver of each link in L_i estimates and checks the affectance to and from selected links in $\cup_{j<i} S_j$, and obtain the sets X_i and R_i of nodes that survive this process. Then, at Line 17 we compute a restricted ruling R_i^r using Algorithm 2, such that R_i^r is an (ω_1, ω_2)-ruling of $\{x(k) : x(k) \in X_i \text{ and } r(k) \in R_i, \forall k \in L\}$, where $\omega_1 = \gamma_1 d_i$ and $\omega_2 = \gamma_2 d_i$. Then S_i is the set of links whose sender nodes are in R_i^r. For each link $l \in S_i$, due to the confirmation mechanism (Lines 10-12 in Algorithm 2) in the construction of R_i^r, the receiver $r(l)$ is also aware of that $x(l) \in R_i^r$. Therefore, both the sender and the receiver of each $l \in S_i$ knows the status of l, so that Lines 4 and 8 are valid distributed operations. Algorithm 1 produces a feasible solution as summarized in Lemma IV.2. The lemmas in this section and Theorem IV.5 are true under any length-monotone sublinear power assignment.

Lemma IV.2 (Correctness)**.** *The set $S' = \cup_i S_i'$ of links selected via Algorithm 1 is feasible.*

Lemma IV.2 can be proved through Lemma IV.3, which is a direct result of the definition of ruling and the choice of ω_1, ω_2. In the ith phase of Step 2, the algorithm selects a subset S_i' of links from S_i, such that $\sum_{j \neq i} A(S_j, l) < \psi, \forall l \in S_i'$. Due to Lemma IV.3, $A(S_i \setminus \{l\}, l) < 1 - \psi$ for all $l \in S_i'$ (and the communication on the links in S_i' from all receivers to their corresponding senders are successful).

Lemma IV.3. *For each S_i in Algorithm 1: (i) the set of sender nodes of links in S_i is an (ω_1, ω_2)-ruling of $\{x(k) : x(k) \in X_i$ and $r(k) \in R_i, \forall k \in L_i\}$, (ii) both S_i and $\overleftarrow{S_i}$ are independent sets, and (iii) $\forall l \in S_i$, $A(S_i \setminus \{l\}, l) < 1 - \psi$.*

Algorithm 1: Distributed MaxLSP

input : Set L of links
output: One-shot Schedule S'

/* Each $l \in L$ participates as follows */
1 $S_i \leftarrow \emptyset, S_i' \leftarrow \emptyset, X_i \leftarrow \emptyset, R_i \leftarrow \emptyset, \ \forall i = 1 \dots, g(L)$;
/* **Step 1 (Lines 2-2):** affectance & ruling */
2 **foreach** $i = 1, 2, \dots, g(L)$ **do** /* $g(L)$ phases */
  /* checking affectance: Lines 3-14 */
3   **foreach** $j = 1, 2, \dots, i - 1$ **do** /* $2(i-1)$ slots */
    /* in one time slot */
4   **if** $l \in S_j$ **then** $r(l)$ transmits;
5   **if** $l \in L_i$ **then**
6       $x(l)$ senses, $SP_j(l) \leftarrow$ power sensed by $x(l)$;
7       $A^E(l, S_j) \leftarrow$ estimated $A(l, S_j)$ from $SP_j(l)$;
    /* in a second time slot */
8   **if** $l \in S_j$ **then** $x(l)$ transmits;
9   **if** $l \in L_i$ **then**
10       $r(l)$ senses, $SP_j'(l) \leftarrow$ power sensed by $r(l)$;
11       $A^E(S_j, l) \leftarrow$ estimated $A(S_j, l)$ from $SP_j'(l)$;
  /* in one time slot */
12   **if** $l \in L_i$ *and* $\sum_{j<i} A^E(l, S_j) < 1/4$ **then**
13   add $x(l)$ to X_i;
14   **if** $l \in L_i$ *and* $\sum_{j<i} A^E(S_j, l) < 1/4$ **then**
15   add $r(l)$ to R_i;
  /* restricted ruling: Line 17 */
16   $\omega_1 \leftarrow \gamma_1 d_i, \omega_2 \leftarrow \gamma_2 d_i$;
17   $R_i^r \leftarrow$ RestrictedRuling$(\omega_1, \omega_2, X_i, R_i, \rho\gamma_1)$;
18   $S_i \leftarrow \{k : k\text{'s sender } x(k) \in R_i^r\}$;
/* **Step 2 (Lines 19-19):** verification */
19 **foreach** $i = 1, 2, \dots, g(L)$ **do** /* $g(L)$ phases */
  /* checking affectance: Lines 3-14 */
20   **foreach** $j = i + 1, \dots, g(L)$ **do** /* $g(L) - i$ slots */
    /* in one time slot */
21   **if** $l \in S_j$ **then** $x(l)$ transmits;
22   **if** $l \in L_i$ **then**
23       $r(l)$ senses, $SP_j'(l) \leftarrow$ power sensed by $r(l)$;
24       $A^E(S_j, l) \leftarrow$ estimated $A(S_j, l)$ from $SP_j'(l)$;
  /* in one time slot */
25   **if** $l \in L_i$ *and* $\sum_{j \neq i} A^E(S_j, l) < \psi$ **then**
26     $r(l)$ transmits CONFIRM message to $x(l)$;
27   **if** $l \in L_i$ *and* $x(l)$ *receives CONFIRM from* $r(l)$ **then**
28   add l to S_i';
29 **return** $\cup_i S_i'$

Estimation of affectance. At Line 7, for a link $l \in L_i$, $SP_j(l) = \sum_{l' \in S_j} \frac{P(l')}{d^{\alpha}(r(l'), x(l))} + N$, and $A(l, S_j) =$

$\sum_{l' \in S_j} c_{l'} \frac{d^\alpha(l')}{P(l')} \frac{P(l)}{d^\alpha(x(l),r(l'))}$. Since $x(l)$ knows $l' \in L_j$, it can estimate $P(l')$ and $d(l')$ within a constant factor. Therefore, $x(l)$ can obtain estimated affectance $A^E(l, S_j)$ such that $A^E(l, S_j) \geq A(l, S_j)$ and is within a constant factor of $A(l, S_j)$. Likewise, at Lines 11 and 24 $r(l)$ can calculate an estimate value $A^E(S_j, l)$ of $A(S_j, l)$ with similar properties.

Simplification of Algorithm 1. The affectance checking processes in both Step 1 (Lines 3-14) and Step 2 (Lines 20-20) take $O(g(L))$ iterations in each phase i. We present it this way for the ease of understanding. Now we describe how these can be simplified to $O(1)$-time operations in each phase i. At Line 2, after getting S_i, we immediately make all the receivers and senders of the links in S_i transmit separately in two slots, and all the senders and receivers of higher link classes senses the power and calculate $A^E(l, S_i)$ and $A^E(S_i, l)$. Then, the $O(g(L))$ iterations at Lines 4 through 11 can be omitted. In the same way, the $O(g(L))$ iterations in each phase i of Step 2 can be avoided. Lemma IV.4 summarizes the running time. The $\rho \log n$ term is from calling RESTRICTEDRULING$(\omega_1, \omega_2, X_i, R_i, \rho\gamma_1)$ (due to Theorem V.1).

Lemma IV.4 (Running Time)**.** *Running time of Algorithm 1 after simplification is $O(g(L)\rho \log n)$ for the case of full-duplex. The restriction of half-duplex communication increases the running time by a $O(\log n)$ factor.*

Theorem IV.5 (Constant Approx.)**.** *The size of the set produced by Algorithm 1 is within a constant factor of $|OPT(L)|$.*

Lemma IV.2 follows from the fact that for each link $l \in S_i, \forall i$, (i) $A(S_i \setminus \{l\}, l) < 1 - \psi$ (in Lemma IV.3) because the senders are ω_1-separated due to the ruling property, and (ii) $A(\sum_{j \neq i} S_j, l) < \psi$ due to Step 2 verifying the affectance.

Theorem IV.5 can be proved in two steps by (i) showing that $|S'|$ is within a constant factor of $|S|$, and (ii) showing that $|S|$ is within a constant factor of $|OPT(L)|$. We sketch the idea as follows, and details can be found in [17]. For the first step, we construct a $|S| \times |S|$ matrix where each element take the value of affectance from one link to another in S, and the affectance from links in the same link class is set to zero. By Claim 2.1 in [13], $|S'| \geq |S|(1 - \frac{1}{2\psi})$. The second step can be proved by contradiction. Recall that γ_1 is a constant. Imagine that we have a subset L' of links from the same link class L_i and the senders are $\gamma_1 d_i$ separated. Let δ be the smallest value such that any such L' is a δ-signal set. Since γ_1 is a constant, δ is a constant too. W.l.o.g., we assume $\frac{1}{\delta} < 1/4$. Let $\delta' = (3^{\alpha+1}2)$. By Claim 2.1 in [13], there exists a set $O \subseteq OPT_{2\delta'}(L)$, such that $|O| \geq |OPT_{2\delta'}(L)|/4$ and $\forall l \in O, A(l, O \backslash \{l\}) \leq \frac{1}{\delta'}$, where $OPT_{2\delta'}(L)$ denotes the maximum $2\delta'$-signal subset of L. Using the Red-Blue Lemma IV.1 with $RED = S$ and $BLUE = O \setminus S$, suppose that $|S| < |O|/5$, then there exists a link $l \in O \setminus S$, such that $A(S^-(l), l) + A(l, S^-(l)) \leq 3^\alpha (A(O \setminus S, l) + A(l, O \setminus S)) \leq \frac{1}{4}$.

W.l.o.g., assume $l \in L_i$. l must have passed the affectance constraint check but got removed because its sender is within at most a distance $\omega_2 = \gamma_2 d_i$ of the sender node $x(l')$ of some selected link l' in S_i during the restricted ruling construction (*i.e.*, Line 19 of Algorithm 2). Since $1/\delta$ is a constant, we can have at most a constant number $\frac{1/4}{1/\delta}$ of such links as l for each link l' in S; otherwise, the $A(S^-(l'), l')$ would exceed $1/4$ which leads to a contradiction. Therefore, $\frac{1/4}{1/\delta}|S| \geq |O|/5$. Hence, $|S| = \Omega(|O|) = \Omega(|OPT_{2\delta'}(L)|) = \Omega(|OPT(L)|)$.

V. DISTRIBUTED ALGORITHM: RESTRICTED (ω_1, ω_2)-RULING

We now present $\text{RestrictedRuling}(\omega_1, \omega_2, X, R, b_{max})$ in Algorithm 2, a distributed algorithm to compute a R-restricted (ω_1, ω_2)-ruling of X, for full duplex communication under the physical interference model; in the end of the section, we extend it to the half duplex setting (where a node can perform transmission and reception/sensing at the same time) with added running time. Here, X is an arbitrary subset of senders and R is an arbitrary subset of receivers of the links in the same link class. For the algorithm to function properly, we require the input parameter $\omega_2 \geq (36\frac{\alpha-1}{\alpha-2})^{\alpha-2}\omega_1$. Recall that $B(v, d)$ denotes the ball centered at v with a radius of d. Let n be the total number of nodes. The last input parameter b_{max} denote the estimate of the maximum number of nodes in the ball $B(v, \omega_1)$ of any node $v \in X$, in the worst case, $b_{max} \leq n$.

Preliminary. All the nodes participating the algorithm use the same transmission power, denoted by P'. This requirement can be relaxed so that nodes may choose power levels varying within a constant factor, by modifying ω_1, ω_2 with a constant factor. We define $Thres(d) = P'/d^\alpha + N$ as a function of distance d, such that for a node v, if any other node is transmitting in a range of d, its sensed power will exceed $Thres(d)$.

Algorithm structure. In this algorithm, we call an iteration of the outer loop (Line 1) a *phase*; we call an iteration of the inner loop (Line 2) a *round*, consisting of *coordination step* (Lines 4 through 7), *confirmation step* (Lines 10 through 12) and *decision step* (Lines 15 through 19). A node v is said to be *active* if v has not joined either $\hat{Y}$ or $\hat{Z}$; otherwise, v becomes *inactive*.

In each round, the coordination step provides a probabilistic mechanism for active nodes in X to compete to get in the ruling (at Line 5). Lines 7 through 7 constitute a module to resolve the issue of sensing and transmitting at the same time, such that two nearby nodes do not both enter the ruling (*i.e.*, Lemma V.3). Next, during the decision step, a subset of active nodes decide to join $\hat{Y}$ or $\hat{Z}$.

Before a node $v \in X$ that survived the coordination step with $U(v) = 1$ notify all the nodes in the neighborhood, the confirmation step (Lines 10 through 12) asks v's corresponding receiver node to "approve" its request to be included in the ruling. That is, v survives the confirmation step if and only if

(1) $U(v) = 1$ after the coordination step;
(2) the corresponding receiver v' of v appears in R;
(3) v' successfully receives its REQUEST message; and
(4) v successfully receives ACK from v'.

In each phase, there are $C_4 \log n$ rounds, such that we can ensure a fraction of the node population have either joined $\hat{Y}$ or $\hat{Z}$, and we expect the maximum number of active nodes in the nearby region of any active node to decrease by at least 1. After each phase, the probability for each active node to access the channel and compete increments by $1/b_{max}$ (at Line 5).

Algorithm 2: RestrictedRuling$(\omega_1, \omega_2, X, R, b_{max})$: Distributed algorithm for computing an R-restricted (ω_1, ω_2)-ruling of X with full duplex radios.

input : $\omega_1, \omega_2, X, R, b_{max}$
output: $\hat{Y}$: an R-restricted (ω_1, ω_2)-ruling of X

/* Each $v \in X \cup R$ participates as follows */
1 **for** $i_{out} = 1$ *to* b_{max} **do** /* $O(b_{max})$ phases */
2   **for** $i_{in} = 1$ *to* $C_4 \log n$ **do** /* $O(\log n)$ rounds */
3   **if** *v is active* **then**
      /* Coordination Step (Lines 4-7): 1 slot */
4       $U(v) \leftarrow 0$;
5   **if** $v \in X$ **then**
6         $U(v)$ flips to 1 w/ prob. $\frac{i_{out}}{b_{max}}$;
7   **if** $U(v) = 1$ **then**
8   v transmits and senses, $SP(v) \leftarrow$ the power v receives in this slot;
9   **if** $SP(v) > Thres(\omega_1, P(v))$ **then** $U(v) \leftarrow 0$;
      /* Confirmation Step (Lines 10-12): 2 slots */
10   **if** $U(v) = 1$ **then**
11   v transmits REQUEST containing its own and the receiver's id's;
12   **else if** *$v \in R$ successfully receives a REQUEST* **then**
13   **if** *REQUEST is from its corresponding sender* **then**
14   v transmits ACK;
      /* Decision Step (Lines 15-19): 1 slot */
15   **if** $U(v) = 1$ **then**
16   **if** *v successfully receives an ACK* **then**
17   v transmits CONFIRM, v joins $\hat{Y}$; /* v becomes inactive */
18   **else** v joins $\hat{Z}$; /* v becomes inactive */
19   **else if** $v \in X$ **then**
20   v senses, $SP(v) \leftarrow$ the power v receives in this slot;
21   **if** $SP(v) > Thres(\omega_1)$ **then** v joins $\hat{Z}$; /* v becomes inactive */
22 **return** $\hat{Y}$;

After the total of b_{max} phases, we have Lemmas V.2, V.4, V.5 that lead to Theorem V.1.

Theorem V.1 (Correctness)**.** *Algorithm 2 terminates in $O(b_{max} \log n)$ time. By the end of the algorithm: (1) $\hat{Y}$ forms an R-restricted (ω_1, ω_2)-ruling of X and (2) $\hat{Z} = X \setminus \hat{Y}$* w.h.p.

Theorem V.1 follows directly from the lemmas below. Lemmas V.2, V.3 and V.5 prove that $\hat{Y}$ is an R-restricted (ω_1, ω_2)-ruling of X, *w.h.p.* Lemmas V.2, V.4, V.5 together shows that $\hat{Z}$ complements $\hat{Y}$ in X, and all nodes in $\hat{Z}$ such that their corresponding receivers appear in R are ω_2-covered by $\hat{Y}$ *w.h.p.* To help the reading flow, we defer most of the technical details to the online technical report [17].

Lemma V.2 (Completion)**.** *By the end of the algorithm, all nodes in X have joined either $\hat{Y}$ or $\hat{Z}$,* i.e.*, all nodes in X become inactive,* w.h.p.

Lemma V.2 implies that $\hat{Z} = X \setminus \hat{Y}$. We say a node $v \in \hat{Y}$ is "*good*," if and only if $d(v, v') \geq \omega_1, \forall v' \in \hat{Y}$ and $v' \neq v$. In Algorithm 2, When a node enters $\hat{Y}$, it makes sure that there are no other ones entering $\hat{Y}$ within a range of ω_1, and it deactivate all the active nodes in the same range. Therefore, we have the following Lemmas V.3 and V.4.

Lemma V.3 (Quality of $\hat{Y}$)**.** *All nodes in $\hat{Y}$ are good.*

Lemma V.4 (Quality of $\hat{Z}$: Part 1)**.** *$\hat{Z}$ contains all the nodes ω_1-covered by $\hat{Y}$.*

Lemma V.5 (Quality of $\hat{Z}$: Part 2)**.** *Further, suppose all nodes in $\hat{Y}$ are good, then all nodes in $\hat{Z}$ such that their corresponding receivers appear in R are ω_2-covered by $\hat{Y}$, $\forall \omega_2 \geq (36\frac{\alpha-1}{\alpha-2})^{\frac{1}{\alpha-2}}\omega_1$.*

Half Duplex Communication. Now, we assume that nodes are in the half duplex mode, so that they cannot perform transmission and reception/sensing at the same time. In Algorithm 2, Lines 7 through 7 make use of the full duplex capability, such that Lemma V.3 is true. To account for the case of half duplex, if we replace the one-slot deterministic full duplex mechanism (Lines 7 through 7) with a randomized $O(\log n)$-time loop — illustrated by the following lines of pseudo code — we have Lemma V.6 for half duplex communication as the counterpart of Lemma V.3 for full duplex. The cost incurred includes (i) the increase in the total running time to obtain an (ω_1, ω_2)-ruling by $O(\log n)$, and (ii) a weakened statement in Lemma V.6 compared to Lemma V.3.

Algorithm 3: In replacement of Lines 7 through 7 in Algorithm 2 for using half duplex radios.

1 **for** $j = 1$ *to* $C_5 \log n$ **do** /* resolving half duplex communication */
  /* in each slot */
2   **if** $U(v) = 1$ **then** v transmits with prob. $1/2$;
3   **if** *v does not transmit* **then**
4   v senses, $SP(v) \leftarrow$ the power v receives in this slot;
5   **if** $SP(v) > Thres(\omega_1)$ **then** $U(v) \leftarrow 0$; /* stops */

Lemma V.6 (Quality of $\hat{Y}$: Half Duplex Mode)**.** *All nodes in $\hat{Y}$ are good,* w.h.p.

Since Lemmas V.2, V.4 and V.5 remain valid, we obtain the following theorem for the half duplex case.

Theorem V.7 (Half Duplex)**.** *There exists a modified version of* RestrictedRuling$(\omega_1, \omega_2, X, R, b_{max})$ *for the half duplex case, such that it finishes in $O(b_{max} \log^2 m)$ time and by the end of the algorithm: (1) $\hat{Y}$ forms an R-restricted (ω_1, ω_2)-ruling of X and (2) $\hat{Z} = X \setminus \hat{Y}$* w.h.p.

VI. Distributed Algorithm for MaxLBP

MaxLBP is different from MaxLSP, since only the set V of senders is specified as the input, instead of sender-receiver pairs. Further, all potential receiving locations in range $d(v)$ for each $v \in V$ need to be considered. We only provide an algorithm under uniform power assignment, due to the challenge of no assistance from receiver nodes. We define the notion of affectance of node w on node v as: $A(w,v) = c_v \frac{d^\alpha(v)}{d^\alpha(w,v)}$, where $c_v = \frac{\beta}{(1-\beta N d^\alpha(v)/P)}$. It is extended to a set S of nodes as $A(S,v) = \sum_{w \in S} A(w,v)$. We say S is *feasible* if and only if S is a solution (not necessarily optimal) to MaxLBP. Note that there exists some constant $\psi' < 1$ such that S is feasible if $\forall v \in S, A(S \setminus \{v\}, v) \leq \psi'$. The sequential MaxLBP algorithm involves the steps below.

(1) Consider the nodes $v_1, \ldots, v_n$ in V in a sorted non-decreasing order of their ranges; $S = \emptyset$ initially;
(2) For $i = 1$ to n: if $A(S, v_i) < \psi'/4$, add v_i to S;
(3) Output $S' = \{v \in S : \ A(S \setminus \{v\}, v) \leq \psi'\}$.

Note that this algorithm returns a feasible solution by construction. The constant factor approximation guarantee follows by an adaptation of the Red-Blue Lemma of [10].

Algorithm 4: Distributed MaxLBP

input : Set V of nodes, with range $d(v)$ for each $v \in V$. We assume V_i is the set of nodes v with range $d(v) \in (2^{i-1}, 2^i]$. $d_i = 2^i$.
output: Subset V' of V for MaxLBP

1 $S'_i \leftarrow \emptyset,\ \forall i = 1 \ldots, g(V)$; /* initialization */
2 $S_i \leftarrow \emptyset, S'_i \leftarrow \emptyset, X_i \leftarrow \emptyset, R_i \leftarrow \emptyset,\ \forall i = 1 \ldots, g(L)$;
3 **foreach** $i = 1, 2, \ldots, g(V)$ **do** /* $g(V)$ phases */
  /* check affectance */
4   **if** $v \in \cup_{j<i} S'_j$ **then**
5   v transmits;
6   **if** $v \in V_i$ **then**
7   v senses, $SP_{<i}(v) \leftarrow$ the sensed power at v;
8     $A^E(\cup_{j<i} S'_j, v) \leftarrow$ estimated $A(\cup_{j<i} S'_j, v)$;
9   **if** $A^E(\cup_{j<i} S'_j, v) < \psi'/4$ **then** add v to S_i;
10   $\omega_1 \leftarrow \gamma'_1 d_i, \omega_2 \leftarrow \gamma'_2 d_i$;
11   $S'_i \leftarrow R_{\omega_1,\omega_2}(S_i)$; /* subroutine from [14] */
12 **foreach** $i = g(V), \ldots, 2, 1$ **do** /* $g(V)$ phases */
13   **if** $v \in S'_i$ **then**
14   v transmits;
15   **if** $\sum_j A^E(S'_j, v) < \psi'$ **then** add v to V';
16   **if** $v \in \cup_{j<i} S'_j$ **then**
17   v senses, $SP_i(v) \leftarrow$ the sensed power at v;
18     $A^E(S'_i, v) \leftarrow$ estimated $A(S'_i, v)$;
19 **return** $\cup_i V'_i$

Algorithm 4 employs at Line 11 the Ruling construction subroutine from [14], which outputs an (ω_1, ω_2)-ruling of S_i. Theorem VI.1 summarizes the performance of algorithm 4; we discuss the extension to half-duplex communication in [17].

Theorem VI.1. *Algorithm 4 runs in time $O(g(V) \log^2 n)$, and produces a feasible solution for* MaxLBP *within a constant factor of $|OPT_{\text{MaxLBP}}(V)|$ with high probability, under uniform or linear power assignment.*

VII. Simulation Results

We study the empirical performance of our distributed algorithms here. Our main observations are (i) the approximation ratio of our algorithm is close (within a factor of 2) of the sequential algorithm, and (ii) the topology, especially the sizes of the different length classes impacts the performance, and the links which are removed based on different criteria in the algorithm. We expect these insights can be useful in improving the empirical performance of our distributed algorithms.

We used randomly generated network topologies with 5, 10, 50, 100, 500, 1000 links, with the links spanning a total of 8 length classes (corresponding to a range of link lengths from 0.01 to 1.4). We considered linear and uniform power assignment schemes, as they represent the two extremes of sublinear power assignments. A simulation scenario specifies the network size, the link classes involved, the scheduling algorithm and the power assignment in use. We fix the parameters in the algorithms for all simulation scenarios, and average over 500 iterations. Figure 3 shows the gap between the output sizes of the distributed and centralized algorithms. For topologies with 3 small length classes, the gap is close to 2. For topologies with up to 8 length classes, or with only 3 large length classes, the gap is even smaller. Figure 4 illustrates the number of selected links, the number of removed links due to affectance violation, and the number of removed links due to ruling construction in different scenarios for the distributed algorithm. The mechanism taking the most effect in each scenario can differ drastically, e.g., in Figure 4a 60% of the links were removed due to affectance violation, whereas in Figure 4a over 90% of the links were removed due to the ruling construction.

VIII. Conclusion

In this paper, we present the first set of fast distributed algorithms in the SINR model for the MaxLSP problem with non-uniform power assignment, and the MaxLBP problem with uniform power assignment, but non-uniform ranges. Our algorithms give constant factor approximation guarantees, matching the bounds of the sequential algorithms for these problems. Our algorithm is randomized and crucially relies on physical carrier sensing for the distributed communication steps. We find that the specific wireless device capability of duplex/half-duplex communication significantly impacts the performance. Our main technique is based on the distributed estimation of affectance and ruling computation, which are likely to be useful in the design of other distributed algorithms.

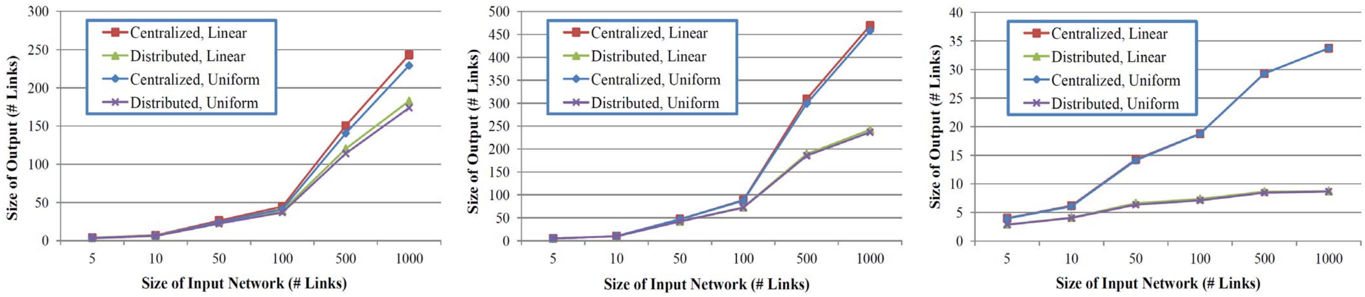
References

[1] S. Ramanathan and E. L. Lloyd, "Scheduling algorithms for multihop radio networks," *IEEE/ACM Trans. on Netw.*, vol. 1, pp. 166–177, 1993.

[2] O. Goussevskaia, Y. A. Oswald, and R. Wattenhofer, "Complexity in geometric sinr," in *ACM MobiHoc*, 2007.

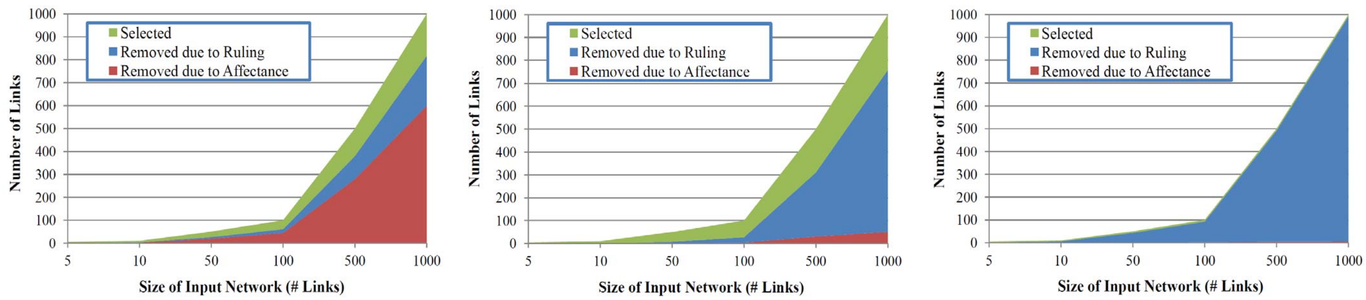
[3] C. Avin, Y. Emek, E. Kantor, Z. Lotker, D. Peleg, and L. Roditty, "Sinr diagrams: towards algorithmically usable sinr models of wireless networks," in *ACM PODC*, 2009, pp. 200–209.

[4] O. Goussevskaia, T. Moscibroda, and R. Wattenhofer, "Local Broadcasting in the Physical Interference Model," in *ACM SIGACT-SIGOPT DialM-POMC*, 2008.



(a) Network consisting of 8 link classes. (b) Network consisting of 3 small link classes. (c) Network consisting of 3 large link classes.

Fig. 3. Performance of our distributed algorithm comparing to the centralized algorithm in [13]. In the legends, ”Centralized” represents the centralized algorithm, ”Distributed” represents the distributed algorithm, ”Linear” represents linear power assignment, and ”Uniform” represents uniform power assignment.



(a) Network consisting of 8 link classes. (b) Network consisting of 3 small link classes. (c) Network consisting of 3 large link classes.

Fig. 4. Profiling of our distributed algorithm in terms of number of selected links, number of removed links due to affectance violation, and number of removed links due to ruling construction.

[5] T. Moscibroda and R. Wattenhofer, “Maximal independent sets in radio networks,” in *ACM symposium on Principles of distributed computing (PODC)*, 2005.

[6] J. Schneider and R. Wattenhofer, “A log* distributed maximal independent set algorithm for growth-bounded graphs,” in *ACM symposium on Principles of distributed computing (PODC)*, 2005.

[7] D. Chafekar, V. S. A. Kumar, M. Marathe, S. Parthasarathy, and A. Srinivasan, “Approximation algorithms for computing capacity of wireless networks with sinr constraints,” in *INFOCOM 2008*, 2008, pp. 1166–1174.

[8] I. Akyildiz, W. Lee, M. Vuran, and S. Mohanty, “Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey,” *Computer Networks*, vol. 50, pp. 2127–2159, 2006.

[9] O. Goussevskaia, M. Halldórsson, R. Wattenhofer, and E. Welzl, “Capacity of arbitrary wireless networks,” in *IEEE INFOCOM*, 2009.

[10] M. M. Halldórsson and R. Wattenhofer, “Wireless communication is in apx,” in *ICALP*, 2009, corrected version: http://www.hr.is/faculty/mmh/papers/juniform.pdf.

[11] P.-J. Wan, X. Jia, and F. Yao, “Maximum independent set of links under physical interference model,” in *Proceedings of WASA*, 2009, pp. 169–178.

[12] T. Kesselheim, “A constant-factor approximation for wireless capacity maximization with power control in the sinr model,” in *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2011.

[13] M. M. Halldórsson and P. Mitra, “Wireless capacity with oblivious power in general metrics,” in *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2011.

[14] G. Pei and A. K. S. Vullikanti, “Brief announcement: Distributed algorithms for maximum link scheduling under the physical interference model,” in *International Symposium on DIStributed Computing (DISC)*, 2012, full version at http://arxiv.org/abs/1208.0811.

[15] C. Scheideler, A. Richa, and P. Santi, “An $o(\log n)$ dominating set protocol for wireless ad-hoc networks under the physical interference model,” in *ACM MobiHoc*, 2008, pp. 91–100.

[16] E. I. Ásgeirsson and P. Mitra, “On a game theoretic approach to capacity maximization in wireless networks,” in *IEEE INFOCOM*, 2011.

[17] G. Pei and V. S. A. Kumar, “Distributed approximation algorithms for link scheduling and local broadcasting in the physical interference model,” 2012, technical Report, Virginia Bioinformatics Institute: http://filebox.vt.edu/users/somehi/papers/maxlspnonuniform12.pdf.

[18] M. Luby, “A simple parallel algorithm for the maximal independent set problem,” in *ACM STOC*, 1985.

[19] H. Balakrishnan, C. Barrett, V. S. A. Kumar, M. Marathe, and S. Thite, “The distance 2-matching problem and its relationship to the MAC layer capacity of adhoc wireless networks,” *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, pp. 1069–1079, 2004.

[20] Y. Afek, N. Alon, O. Barad, E. Hornstein, N. Barkai, and Z. Bar-Joseph, “A biological solution to a fundamental distributed computing problem,” *Science*, vol. 331, pp. 183–185, 2011.

[21] Y. Afek, N. Alon, Z. Bar-Joseph, A. Cornejo, B. Haeupler, and F. Kuhn, “Beeping a maximal independent set,” in *DISC*, 2011.

[22] D. Chafekar, V. S. A. Kumar, M. V. Marathe, S. P. 0002, and A. Srinivasan, “Cross-layer latency minimization in wireless networks with sinr constraints,” in *ACM MobiHoc*, 2007, pp. 110–119.

[23] P. Gupta and P. R. Kumar, “The capacity of wireless networks,” *IEEE Trans. on Info. Theory*, vol. 46, no. 2, pp. 388–404, 2000.

[24] P.-J. Wan, O. Frieder, X. Jia, F. F. Yao, X. Xu, and S. Tang, “Wireless link scheduling under physical interference model,” in *IEEE INFOCOM*, 2011, pp. 838–845.

[25] M. Dinitz, “Distributed algorithms for approximating wireless network capacity,” in *IEEE INFOCOM*, 2010.

[26] T. Kesselheim and B. Vöcking, “Distributed contention resolution in wireless networks,” in *DISC*, 2010.

[27] M. M. Halldórsson and P. Mitra, “Nearly optimal bounds for distributed wireless scheduling in the sinr model,” in *ICALP*, 2011.

[28] R. Cole and U. Vishkin, “Deterministic coin tossing and accelerating cascades: micro and macro techniques for designing parallel algorithms,” in *ACM symposium on Theory of computing (STOC)*, 1986, pp. 206–219.

[29] G. Pei and V. A. Kumar, “Low-complexity scheduling for wireless networks,” in *ACM Mobihoc*, 2012.