

Understanding the Capacity Region of the Greedy Maximal Scheduling Algorithm in Multi-hop Wireless Networks

Changhee Joo

Xiaojun Lin

Ness B. Shroff

Departments of ECE and CSE
The Ohio State University
Email: cjoo@ece.osu.edu

CWSA, School of ECE
Purdue University
Email: linx@ecn.purdue.edu

Departments of ECE and CSE
The Ohio State University
Email: shroff@ece.osu.edu

Abstract—In this paper, we characterize the performance of an important class of scheduling schemes, called *Greedy Maximal Scheduling (GMS)*, for multi-hop wireless networks. While a lower bound on the throughput performance of *GMS* is relatively well-known in the simple *node-exclusive* interference model, it has not been thoroughly explored in the more general *K-hop* interference model. Moreover, empirical observations suggest that the known bounds are quite loose, and that the performance of *GMS* is often close to optimal. In this paper, we provide a number of new analytic results characterizing the performance limits of *GMS*. We first provide an equivalent characterization of the efficiency ratio of *GMS* through a topological property called the *local-pooling factor* of the network graph. We then develop an iterative procedure to estimate the *local-pooling factor* under a large class of network topologies and interference models. We use these results to study the worst-case efficiency ratio of *GMS* on two classes of network topologies. First, we show how these results can be applied to tree networks to prove that *GMS* achieves the full capacity region in tree networks under the *K-hop interference model*. Second, we show that the worst-case efficiency ratio of *GMS* in geometric network graphs is between $\frac{1}{6}$ and $\frac{1}{3}$.

I. INTRODUCTION

Over the last few years there has been significant interest in studying the *scheduling* problem for multi-hop wireless networks [1]–[9]. In general, this problem involves determining which links should transmit (i.e., which node-pairs should communicate) and at what times, what modulation and coding schemes should be used, and at what power levels should communication take place. While the optimal solution of this scheduling problem has been known for a long time [2], the resultant solution has high computational complexity and is difficult to implement in multi-hop networks. For example, under the simplest 1-hop interference model (also known as the *node-exclusive* or *primary* interference model), where two links interfere with each other only if they are within 1-hop distance, the throughput-optimal policy of [2] corresponds to a *Maximum Weighted Matching (MWM)* policy and its complexity is roughly $O(N^3)$ [10], where N is the total number of nodes in the network. While the 1-hop interference model has been used as a reasonable approximation to Bluetooth or FH-CDMA networks ([1], [11], [12]), a large class of systems

can be modeled using the more general *K-hop* interference models, in which any two links within *K-hop* distance cannot be activated simultaneously. For example, the ubiquitous IEEE 802.11 DCF (Distributed Coordination Function) wireless networks is often modeled as the 2-hop interference model [13], [14]. The complexity of the throughput-optimal policy of [2] for the *K-hop* interference model is NP-Hard [15], and hence, it is difficult to implement in practice.

In this paper, we are interested in a well-known suboptimal scheduling policy called the *Greedy Maximal Scheduling (GMS)*, which determines its schedule by choosing links in a decreasing order of the backlog while conforming to interference constraints. *GMS* has low complexity [1], [16], [17] and may be implemented in a distributed manner. However, to date its performance is still not well-understood. We can characterize the performance of *GMS* through its efficiency ratio γ^* , which can be defined as the achievable fraction of the optimal capacity region (see Definition 2). Under the 1-hop interference model, it is relatively straightforward to show that the efficiency ratio of *GMS* is at least 1/2, i.e., *GMS* can sustain at least half of the throughput of the optimal policy. However, simulation results suggest that the performance of *GMS* is often much better than this lower bound in most network settings. Under the *K-hop* interference model, we do not even know how to characterize the efficiency ratio of *GMS* in most network graphs. It has been shown only in geometric network graphs that $\gamma^* \geq \frac{1}{8}$ in the 2-hop interference model [13] and $\gamma^* \geq \frac{1}{49}$ in the *K-hop* interference model with $K \geq 2$ [15]. However, again simulation results suggest that the actual performance of *GMS* is often much better.

Recently, Dimakis and Walrand [18] have shown that if the network topology satisfies the so-called *local-pooling* condition, then *GMS* can in fact achieve the full capacity region. The idea is extended in [19], [20] to find network topologies that maximize the throughput under *GMS*. Unfortunately, realistic network topologies may not satisfy the *local-pooling* condition. Hence, the true efficiency ratio of *GMS* in many network scenarios remains unknown.

The main objective of this paper is to understand the achievable efficiency ratio of *GMS* for a large class of network topologies and interference models. Understanding the performance limits of *GMS* is important for the following

This work has been supported in part by ARO MURI Award No. W911NF-07-10376 (SA08-03) and NSF awards CNS-0626703, CNS-0721236, ANI-0207728, and CCF-0635202, USA, and in part by IT Scholarship Program supervised by IITA and MIC, Republic of Korea.

reasons. First, it has been empirically observed in [3] that the centralized *GMS* outperforms many distributed scheduling schemes and achieves virtually the same throughput as the throughput-optimal scheduling for a variety of networking scenarios. Second, although there have been some recently developed distributed scheduling schemes [8], [9] that can achieve the maximum achievable throughput, the study of *GMS* continues to remain attractive because, empirically, *GMS* performs better than these schemes in terms of the resultant queueing delay [3], [6]. Third, it has been known in [21] that *GMS* can be also implemented in a distributed manner, which is critical from the point of view of many multi-hop wireless systems and applications. Finally, recent studies have proposed even simpler constant-time-complexity random algorithms [3], [4], [7] that appear to approximate the performance of *GMS* by giving a larger weight to a link with a larger queue length.

In this paper, we provide a number of new analytical results along this direction. We first generalize the notion of local-pooling in [18] to the notion of local-pooling factor, which is a topological property of a graph. We show that, under arbitrary interference models, the efficiency ratio of *GMS* for a given network graph is equal to the local-pooling factor of the graph. We then develop an iterative procedure to determine a lower bound on the local-pooling factor of a network graph, and a sufficient condition for a lower bound on the worst-case local-pooling factor over a class of network topologies. We next apply these results to two classes of network topologies. First, we show how these results can be applied to tree networks to prove that *GMS* achieves the full capacity for any tree network under a K -hop interference model. (This result was shown in [20] by using a different approach.) Second, We also develop much sharper bounds on the worst-case efficiency ratio for geometric graphs than those known in the literature.

The rest of the paper is organized as follows. We first describe our system model in Section II. We then provide in Section III an equivalent characterization of the efficiency ratio of *GMS* through the local-pooling factor of the underlying network graph. We develop an iterative analysis method estimating the local-pooling factor of a network graph in Section IV. Using the new methodology, we show that *GMS* achieves the full capacity region in tree topologies under the K -hop interference model in Section V. We also provide new results bounding the efficiency ratio of *GMS* in geometric network graphs in Section VI. We conclude in Section VII.

II. NETWORK MODEL

We model a wireless network by a graph $G(V, E, I)$, where V is the set of nodes, E is the set of undirected links, and I represents interference constraints (e.g., an $|E| \times |E|$ interference matrix). For each link l , let $I(l)$ denote the set of links interfering with l . For convenience, we adopt the convention that $l \in I(l)$. We define the interference degree $d(l)$ as the maximum number of links in $I(l)$ that do not interfere with each other. We assume a time-slotted system, where the length of each time slot is of unit length. We assume that in each time slot, link l can transmit one packet provided that no

other links in $I(l)$ are transmitting at the same time. If two interfering links transmit at the same time, neither of them can transmit any data. A set of active (i.e., transmitting) links forms a *feasible schedule* in E if none of them interfere with each other. Note that this model is very general and can represent a large class of wireless networks. For example, in the so-called K -hop interference model, two links within K -hop distance interfere with each other. We can correspondingly define $I(l)$ as, for all links $l \in E$,

$$I(l) = \{k \in E \mid \text{the distance between links } l \text{ and } k \\ \text{is less than or equal to } K \text{ hops}\}.$$

A *maximal schedule* \vec{M} on E is defined as a feasible schedule, such that, when all links in \vec{M} are activated, no more links can be activated without violating the interference constraint. We use a vector in $\{0, 1\}^{|E|}$ to denote a maximal schedule \vec{M} such that the k -th element M_k is set to 1 if link $k \in E$ is included in \vec{M} , and to 0 otherwise. Let \mathcal{M}_E be the set of all possible \vec{M} 's and let $Co(\mathcal{M}_E)$ denote its convex hull. We define a *maximal scheduling vector* $\vec{\phi}$ in E as a vector $\vec{\phi} \in Co(\mathcal{M}_E)$.

We assume that packets arrive to each link l according to a stationary and ergodic process, and the average arrival rate is λ_l . The *capacity region* (or *stability region*) under a given scheduling policy is defined as the set of arrival rate vectors $\vec{\lambda} = \{\lambda_1, \lambda_2, \dots, \lambda_{|E|}\}$ such that the system is stable (i.e., all queues are kept finite). We define the *optimal capacity region* Λ as the union of the capacity regions of all scheduling policies. It is known that Λ is given by [2], [22], [23],

$$\Lambda = \left\{ \vec{\lambda} \mid \vec{\lambda} \preceq \vec{\phi}, \text{ for some } \vec{\phi} \in Co(\mathcal{M}_E) \right\}, \quad (1)$$

where $\vec{x} \preceq \vec{y}$ denotes that \vec{x} is component-wise dominated by \vec{y} . Let $\dot{\Lambda}$ denote the interior of Λ . It is well-known that the scheduling policy of [2], which we refer to as the Maximal Weighted Scheduling (*MWS*) policy, achieves the capacity region $\dot{\Lambda}$. *MWS* chooses a schedule at each time slot t that maximizes the total queue weighted rate sum as

$$\vec{M}^*(t) = \operatorname{argmax}_{\vec{M} \in \mathcal{M}_E} \sum_{l \in E} q_l(t) M_l,$$

where $q_l(t)$ is the backlog of link l at time t . However, this policy has high computational complexity. The complexity is $O(N^3)$ under the 1-hop interference model and is in general NP-Hard under K -hop interference models ($K \geq 2$). In this paper, we are interested in a suboptimal (but much simpler) policy called *Greedy Maximal Scheduling (GMS)* or *Longest Queue First (LQF)* policy. *GMS* can be viewed as an approximation of *MWS*. It operates as follows: at each time slot, it first picks the link l with the largest backlog; it then discards all links that interfere with link l ; it then picks the link with the largest backlog from the remaining links; and this process continues until no links left. Our goal of the paper is to characterize the efficiency ratio of *GMS* under arbitrary network topologies. The efficiency ratio is defined as follows.

Definition 1: For a suboptimal scheduling policy, e.g.,

GMS, we say that it achieves a *fraction* γ of the capacity region under a given network topology if it can keep the system stable for any offered load $\vec{\lambda} \in \gamma\Lambda$.

Definition 2: The *efficiency ratio* $\gamma^*(G)$ of a scheduling policy under a given network graph $G(V, E, I)$ is the supremum of all γ such that the policy can achieve a fraction γ of the capacity region, i.e.,

$$\gamma^*(G) := \sup\{\gamma \mid \text{the system is stable under all offered load vectors } \vec{\lambda} \text{ such that } \vec{\lambda} \preceq \gamma\vec{\phi} \text{ for some } \vec{\phi} \in Co(\mathcal{M}_E)\}. \quad (2)$$

III. AN EQUIVALENT CHARACTERIZATION OF THE EFFICIENCY RATIO OF *GMS*

In this section, we provide an equivalent characterization of the efficiency ratio of *GMS* through its topological properties. We first borrow the following definitions from [24].

Definition 3: A set of links L satisfies *σ -local pooling*, if $\sigma\vec{\mu} \not\preceq \vec{\nu}$ for all $\vec{\mu}, \vec{\nu} \in Co(\mathcal{M}_L)$, where \mathcal{M}_L is the set of all possible maximal schedules of links in L and $Co(\mathcal{M}_L)$ is its convex hull. In other words, for all $\vec{\mu}, \vec{\nu} \in Co(\mathcal{M}_L)$, there must exist some $k \in L$ such that $\sigma\mu_k < \nu_k$.

Definition 4: The *local-pooling factor* $\sigma^*(G)$ of a graph $G(V, E, I)$ is the supremum of all σ such that every subset $L \subset E$ satisfies σ -local pooling. In other words,

$$\begin{aligned} \sigma^*(G) &:= \sup\{\sigma \mid \sigma\vec{\mu} \not\preceq \vec{\nu} \text{ for all } L \text{ and all } \vec{\mu}, \vec{\nu} \in Co(\mathcal{M}_L)\} \\ &= \inf\{\sigma \mid \sigma\vec{\mu} \succeq \vec{\nu} \text{ for some } L \text{ and some } \vec{\mu}, \vec{\nu} \in Co(\mathcal{M}_L)\}. \end{aligned} \quad (3)$$

The notion of local-pooling is first introduced in [18]. Their definition of *local-pooling* is equivalent to our definition of a *local-pooling factor* of 1. It was shown in [18] that, if the local-pooling factor of an arbitrary graph is 1, *GMS* can achieve the efficiency ratio of 1. However, realistic network topologies often do not have a local-pooling factor of 1. In our earlier work [24], we show that under the 1-hop interference model, the efficiency ratio of *GMS* under a given network graph is equivalent to the local-pooling factor of the graph. We next generalize this result to arbitrary interference models.

Proposition 1: The efficiency ratio $\gamma^*(G)$ of *GMS* under a given network graph $G(V, E, I)$ is equal to its local-pooling factor $\sigma^*(G)$.

Remark: Since both $\gamma^*(G)$ and $\sigma^*(G)$ are determined by the network G , in the sequel we will simply use γ^* and σ^* when there is no source of confusion regarding the network G .

The proof of Proposition 1 is a straightforward extension of that of Proposition 8 in [24] and its supporting lemmas. We next sketch the main idea of the proof and refer the readers to [24] for the details. First, given the network G , we can obtain $\gamma^* \geq \sigma^*$ by showing that the network is stable under *GMS* at any offered load $\vec{\lambda}$ strictly within $\sigma^*\Lambda$. Specifically, we can show that in the fluid limit the longest queue always decrease under *GMS*. Then, we can show $\gamma^* \leq \sigma^*$ by constructing a particular traffic pattern with rate outside $\sigma^*\Lambda$ such

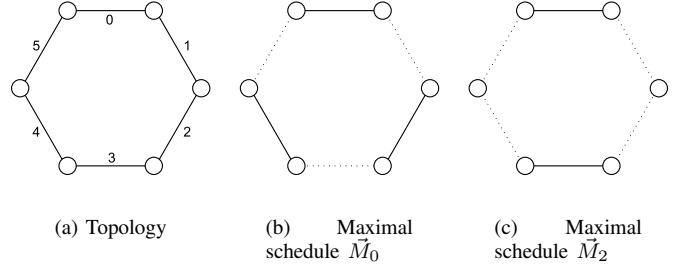


Fig. 1. The 6-link cyclic network and the instances of maximal schedule under the 1-hop interference model. The solid lines in (b) and (c) are the active links.

that the system is unstable. Specifically, from two maximal scheduling vectors $\vec{\mu}, \vec{\nu} \in Co(\mathcal{M}_L)$ satisfying $\sigma^*\vec{\mu} \succeq \vec{\nu}$, we can construct a deterministic traffic pattern that makes the set of queues in L unstable under *GMS* and whose offered load is $\vec{\lambda} = \vec{\nu} + \epsilon\vec{e}_L \in \Lambda$ for some $\epsilon > 0$, where \vec{e}_L is a vector with $e_k = 1$ for $k \in L$ and $e_k = 0$ for $k \notin L$. Hence, we have $\gamma^* \leq \sigma^*$ and the proposition follows.

To illustrate how to construct such a traffic pattern in the latter part of the proof, we provide the following example.

Example: We consider the 6-link cyclic network graph under the 1-hop interference model. We illustrate its topology in Fig. 1(a) and number all links clockwise from 0 to 5. All possible maximal schedules under this network graph are listed below.

- $\vec{M}_0 = \{1, 0, 1, 0, 1, 0\}$, $\vec{M}_1 = \{0, 1, 0, 1, 0, 1\}$,
- $\vec{M}_2 = \{1, 0, 0, 1, 0, 0\}$, $\vec{M}_3 = \{0, 0, 1, 0, 0, 1\}$, $\vec{M}_4 = \{0, 1, 0, 0, 1, 0\}$.

Note that the number of links included in a maximal schedule is three for \vec{M}_0 and \vec{M}_1 , and is two for \vec{M}_2 , \vec{M}_3 , and \vec{M}_4 . Figs. 1(b) and 1(c) show the two instances of the maximal schedules, i.e., \vec{M}_0 and \vec{M}_2 . Note that if we take two convex combinations $\vec{\mu}, \vec{\nu}$ from maximal schedules (i.e., $\vec{\mu}, \vec{\nu} \in Co(\{\vec{M}_i\})$) as

$$\begin{aligned} \vec{\mu} &= \frac{1}{2}\vec{M}_0 + \frac{1}{2}\vec{M}_1 = \left\{\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right\}, \\ \vec{\nu} &= \frac{1}{3}\vec{M}_2 + \frac{1}{3}\vec{M}_3 + \frac{1}{3}\vec{M}_4 = \left\{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right\}, \end{aligned}$$

and hence, $\frac{2}{3}\vec{\mu} \succeq \vec{\nu}$. This implies that the 6-link cyclic network does not satisfy σ -local pooling with $\sigma \geq \frac{2}{3}$, i.e., its local-pooling factor σ^* must be no larger than $\frac{2}{3}$.

We now show that the efficiency ratio of *GMS* is no larger than $\frac{2}{3}$ by constructing a particular traffic pattern with offered load $\vec{\lambda} = \vec{\nu} + \frac{\epsilon}{3}\vec{e}$ such that the system is unstable under *GMS*, where $\vec{e} = \{1, 1, 1, 1, 1, 1\}$ and ϵ is a small positive number. Assume that all queues in the system are of the same length at time 0.

- 1) *1st time slot:* One packet is applied to links 0 and 3. Since *GMS* gives priority to links with a longer queue, it will serve links 0 and 3. Therefore, at the end of time slot 1, all queues will still have the same length.
- 2) *2nd time slot:* One packet is applied to links 1 and 4. For the same reason as above, *GMS* will serve links 1

and 4, and all queues will still have the same length at the end of time slot 2.

- 3) *3rd time slot*: With probability $1 - \epsilon$, one packet is applied to links 2 and 5. With probability ϵ , two packets are applied to links 2 and 5, and one packet is applied to all other links. In both cases, links 2 and 5 have the longest queue and will be served by *GMS*. At the end of time slot 3, all queues still have the same length. However, with probability ϵ , the queue length increases by 1.

The pattern then repeats itself.

Over all links, the arrival rate is $\frac{1}{3} + \frac{\epsilon}{3}$ and the queue length increases by 1 with probability ϵ every three time slots. Hence, the system with offered load $\vec{v} + \frac{\epsilon}{3}\vec{e}$ is unstable under *GMS*. However, the optimal policy (*MWS*) can support the offered load $\vec{\mu} = \frac{2}{3}\vec{v}$ in this example. Hence, the efficiency ratio of *GMS* is no greater than $\frac{2}{3}$, i.e., $\gamma^* \leq \frac{2}{3}$ in this 6-link cyclic network under the 1-hop interference model.

Remark: Note that the key in constructing the above traffic pattern is that (i) we keep all queues in L of the same length at all time, (ii) we inject packets according to the maximal schedules that form the vector \vec{v} so that these maximal schedules will be picked by *GMS* at all time, and (iii) the offered load is slightly larger than \vec{v} , i.e., $\vec{\lambda} = \vec{v} + \epsilon\vec{e}_L$ so that the queues of L grow to infinity together. In [24], we show that such a traffic pattern can be constructed for all $\vec{\mu}, \vec{v}$ such that $\sigma\vec{\mu} \succeq \vec{v}$.

Proposition 1 provides an equivalent characterization of the efficiency ratio of *GMS* through the topological properties (i.e., the local-pooling factor) of the given graph. However, it can still be quite difficult to compute the local-pooling factor for an arbitrary network graph. In the next section, we will extend the methodology of Proposition 1 to develop new approaches to estimate the efficiency ratio and the local-pooling factor of arbitrary network graphs.

IV. ESTIMATES OF THE LOCAL-POOLING FACTOR FOR ARBITRARY NETWORK GRAPHS

In this section, we would like to answer the following questions: (i) how do we estimate the local-pooling factor of a given graph? and (ii) what types of graphs will have low local-pooling factors? We now argue that both questions are intimately related to the characterization of a set of unstable links. We first state the following lemma.

Lemma 1: Given network graph $G(V, E, I)$ with the local-pooling factor σ^* , there exist a subset of links $L \subset E$, and two maximal scheduling vectors $\vec{\mu}^*, \vec{v}^* \in Co(\mathcal{M}_L)$ such that $\sigma^*\vec{\mu}^* \succeq \vec{v}^*$.

Proof: Assume that $|E|$ is finite. Since for all $L \subset E$, the set of maximal schedules \mathcal{M}_L has finite elements. Then its convex hull $Co(\mathcal{M}_L)$ is bounded and closed, and thus, compact.

By definition of σ^* , for any $k > 0$, there must exist a subset L_k , and two vectors $\vec{\mu}_k, \vec{v}_k \in Co(\mathcal{M}_{L_k})$ satisfying $(\sigma^* + \frac{1}{k})\vec{\mu}_k \succeq \vec{v}_k$. Hence, we can obtain a sequence $\{(\vec{\mu}_k, \vec{v}_k)\}$. Since the number of subsets of E is finite, there

must exist a subsequence $(\vec{\mu}_{k_n}, \vec{v}_{k_n}) \in Co(\mathcal{M}_L) \times Co(\mathcal{M}_L)$ for some $L \subset E$. Since $Co(\mathcal{M}_L)$ is a compact set, $Co(\mathcal{M}_L) \times Co(\mathcal{M}_L)$ is compact and hence, $\{(\vec{\mu}_{k_n}, \vec{v}_{k_n})\}$ has a convergent subsequence that converges to some element of $Co(\mathcal{M}_L) \times Co(\mathcal{M}_L)$. Let $(\vec{\mu}_{k_i}, \vec{v}_{k_i})$ denote the subsequence converging to $(\vec{\mu}^*, \vec{v}^*) \in Co(\mathcal{M}_L) \times Co(\mathcal{M}_L)$. Hence, from $(\sigma^* + \frac{1}{k_i})\vec{\mu}_{k_i} \succeq \vec{v}_{k_i}$ for all k_i , we obtain $\vec{\mu}^*, \vec{v}^* \in Co(\mathcal{M}_L)$ and $\sigma^*\vec{\mu}^* \succeq \vec{v}^*$. ■

The idea in the rest of the section is as follows. Suppose that we want to show that $\sigma^* \geq \sigma$ for some $\sigma > 0$. We want to prove by a contradiction. Assume in contrary that $\sigma^* < \sigma$. Given a network graph $G(V, E, I)$ with the local-pooling factor σ^* , there exist a set $Y \subset E$ and two $\vec{\mu}^*, \vec{v}^* \in Co(\mathcal{M}_Y)$ such that $\sigma^*\vec{\mu}^* \succeq \vec{v}^*$ from Lemma 1. According to the proof of Proposition 1 (see the example in Section III), we can then construct a traffic pattern with offered load $\vec{v}^* + \epsilon\vec{e}_Y$ such that the queues of all links in Y increase to infinity together under *GMS*. Let $\vec{\lambda}^*(\epsilon) = \vec{v}^* + \epsilon\vec{e}_Y \in \mathring{\Lambda}$ denote this offered load¹. We refer to this set Y as the unstable links. Clearly, if we can show that $Y = \emptyset$, then this leads to a contradiction, which then implies that $\sigma^* \geq \sigma$.

Towards this end, we first study the properties of this set Y of unstable links.

A. Properties of unstable links

For a subset $L \subset E$, we let $I_L(l) = I(l) \cap L$ denote the set of links in L that interfere with link l , and define the interference degree $d_L(l)$ as the maximum number of links in $I_L(l)$ that can be scheduled at the same time without interfering with each other. We begin with the following two lemmas.

Lemma 2: If $\vec{\lambda} \in \mathring{\Lambda}$, then $\sum_{j \in I_L(l)} \lambda_j \leq d_L(l)$ for all $l \in L$ and all $L \subset E$.

Proof: The lemma can be proven by a contradiction. We assume that there exist a subset $L \subset E$ and a link $l \in L$ such that $\sum_{j \in I_L(l)} \lambda_j > d_L(l)$. Since $\vec{\lambda}$ is within $\mathring{\Lambda}$, it can be stabilized by some scheduling policy. However, at any time, any schedule must satisfy the interference constraints and thus, cannot serve more than $d_L(l)$ links out of $I_L(l)$. Hence, the summation of any feasible service rate over $I_L(l)$ cannot exceed $d_L(l)$, which is smaller than the sum of the rates with which packet arrives at $I_L(l)$. Therefore, the network is unstable, which contradicts our assumption. The result of Lemma 2 then follows. ■

Lemma 3: Assume that Y is the set of unstable links under *GMS* with the offered load $\vec{\lambda}^*(\epsilon) = \vec{v}^* + \epsilon\vec{e}_Y$, then for all $l \in Y$, $\sum_{j \in I_Y(l)} \lambda_j^*(\epsilon) > 1$.

Proof: Note that $\vec{v}^* \in Co(\mathcal{M}_Y)$ is a convex combination of elements of \mathcal{M}_Y . For each element $\vec{M} \in \mathcal{M}_Y$, if none of links in $I_Y(l) \setminus \{l\}$ is picked, then l must be picked. Hence, $\sum_{j \in I_Y(l)} M_l \geq 1$. We then have $\sum_{j \in I_Y(l)} v_l^* \geq 1$ and $\sum_{j \in I_Y(l)} \lambda_j^*(\epsilon) > 1$ for all $\epsilon > 0$. ■

Lemma 4: Assume that $\vec{v}^* \in \frac{1}{d}\mathring{\Lambda}$ for some $d \geq 1$ and that Y is the corresponding set of unstable links under *GMS*, then

¹Note that there exists some small $\epsilon > 0$ such that $\vec{\lambda}^*(\epsilon) \in \sigma\mathring{\Lambda}$ because $\vec{v}^* \in \sigma^*\mathring{\Lambda} \subset \sigma\mathring{\Lambda}$.

for all links $l \in Y$, its interference degree in Y must be larger than d , i.e., $d_Y(l) > d$.

Proof: We prove the lemma by a contradiction. Suppose that there is a link $l \in Y$ with $d_Y(l) \leq d$. Pick $\vec{\lambda}^*(\epsilon) = \vec{\nu}^* + \epsilon \vec{e}_Y$ such that $\vec{\lambda}^*(\epsilon)$ is strictly within $\frac{1}{d}\Lambda \subset \frac{1}{d_Y(l)}\Lambda$, we have $\sum_{j \in I_Y(l)} \lambda_j^*(\epsilon) \leq 1$ from Lemma 2. However, by Lemma 3, we should have $\sum_{j \in I_Y(l)} \lambda_j^*(\epsilon) > 1$. This is a contradiction. Hence, $d_Y(l)$ must be larger than d . ■

From Lemma 4, we can derive the following main result.

Proposition 2: Given a network graph $G(V, E, I)$, assume that a sequence of links $\{l_1, l_2, \dots, l_{|E|}\}$ and a sequence of sets $\{L_1, L_2, \dots, L_{|E|}, L_{|E|+1}\}$ with $L_1 = E$ and $L_{|E|+1} = \emptyset$ satisfy that $L_{i+1} = L_i \setminus \{l_i\}$ and $d_{L_i}(l_i) \leq d$ for all $1 \leq i \leq |E|$ with some $d \geq 1$. Then the local-pooling factor is bounded by $\frac{1}{d}$, i.e., $\sigma^* \geq \frac{1}{d}$.

Proof: We prove the proposition by a contradiction. Suppose that $\sigma^* < \frac{1}{d}$, then there exists $\vec{\mu}^*, \vec{\nu}^* \in Co(\mathcal{M}_Y)$ such that $\sigma^* \vec{\mu}^* \succeq \vec{\nu}^*$. Further, $\vec{\nu}^* \in \frac{1}{d}\dot{\Lambda}$ since $\sigma^* < \frac{1}{d}$. Let Y denote the corresponding set of unstable links. We now show that Y must be \emptyset , which is a contradiction.

If $Y \neq \emptyset$, we can pick the link $l \in Y$ with the smallest index in the set $\{l_1, l_2, \dots, l_{|E|}\}$, say $l = l_j$. Then we have that all links $l_i \notin Y$ for $1 \leq i < j$ and hence, $Y \subset L_j$. Since $\vec{\nu}^* \in \frac{1}{d}\dot{\Lambda}$ and $l_j \in Y$, we have $d_Y(l_j) > d$ from Lemma 4. Since we also have $d_Y(l_j) \leq d_{L_j}(l_j)$ from $Y \subset L_j$, and $d_{L_j}(l_j) \leq d$ from our assumption, we arrive at a contradiction and, thus $Y = \emptyset$. ■

Clearly, if there exists a number d such that $d_E(l) \leq d$ for all $l \in E$, then the assumption of Proposition 2 holds for any sequence of links $\{l_1, l_2, \dots, l_{|E|}\}$. Hence, $\sigma^* \geq \frac{1}{d}$ and the efficiency ratio of *GMS* is no smaller than $\frac{1}{d}$. Note that a similar conclusion has been drawn for Maximal Scheduling. In [13], it has been shown that if $d_E(l) \leq d$ for all $l \in E$, then given $\vec{\lambda} \in \frac{1}{d}\dot{\Lambda}$, a Maximal Scheduling policy can stabilize the network. However, Proposition 2 is in fact much stronger than the results in [13] and the efficiency ratio of *GMS* can often be shown to be larger than that of Maximal Scheduling. We highlight this important difference with the following example.

Example: Consider $N + 1$ nodes n_1, n_2, \dots, n_{N+1} lying in a straight line from left to right. Each node is connected only to its immediate neighbors. We denote link (n_i, n_{i+1}) by l_i . Assume the 1-hop interference model. For this network graph, since $d_E(l) \leq 2$ for all links, the efficiency ratio of *GMS* is no smaller than $\frac{1}{2}$. However, *GMS* in fact achieves the full capacity for this graph. The reason is that there always exists a link at the end of the line with interference degree of 1. The existence of this link in fact determine the efficiency ratio of *GMS*. To see this, we pick the sequence of links in Proposition 2 as $\{l_1, l_2, \dots, l_N\}$. We first look at link l_1 on the end of the line. Let $L_1 = E$. Since $d_{L_1}(l_1) = 1$, the assumption of Proposition 2 holds for $i = 2$. Now, we let $L_2 = L_1 \setminus \{l_1\}$ and move our attention to the next link l_2 . Since $d_{L_2}(l_2) = 1$, the assumption of Proposition 2 holds for $i = 2$. We can apply this procedure iteratively to links l_3, l_4, \dots, l_N . Therefore, after the N -th iteration, we will have sequences of

$\{l_1, l_2, \dots, l_N\}$ and $\{E = L_1, L_2, \dots, L_{N+1} = \emptyset\}$ satisfying $L_{i+1} = L_i \setminus \{l_i\}$ and $d_{L_i}(l_i) \leq 1$ for all $1 \leq i \leq N$. Then from Proposition 2, $\sigma^* \geq 1$.

This example motivates us to develop the following iterative procedure to derive a lower bound for the local-pooling factor in arbitrary network graphs.

B. An iterative approach

We present the procedure in Algorithm 1, which bounds the local-pooling factor of the underlying graph, i.e., the efficiency ratio of *GMS*. At each iteration, the algorithm picks up a link and check the interference degree of the chosen link in the remaining network graph.

Algorithm 1 Do_iterations($G(V, E)$)

Initialization: $L_1 \leftarrow E, d \leftarrow 1$

1: **for** $1 \leq i \leq |E|$ **do**

2: Choose a link l_i from L_i

3: **if** $d_{L_i}(l_i) \geq d$ **then**

4: $d \leftarrow d_{L_i}(l_i)$

5: **end if**

6: $L_{i+1} \leftarrow L_i \setminus \{l_i\}$

7: **end for**

8: **return** d

Let d_e denote the returned value at the end of the algorithm. We show that the local-pooling factor σ^* of the graph $G(V, E, I)$ is at least $\frac{1}{d_e}$.

Lemma 5: Given $G(V, E, I)$, if we obtain d_e from Algorithm 1 with a sequence of $\{l_1, l_2, \dots, l_{|E|}\}$, then $\sigma^* \geq \frac{1}{d_e}$.

Proof: The lemma directly follows from Proposition 2. Note that the resulting sequence of links $\{l_1, l_2, \dots, l_{|E|}\}$ and the corresponding sequence of sets $\{E = L_1, L_2, \dots, L_{|E|+1} = \emptyset\}$ satisfy two conditions of $L_{i+1} = L_i \setminus \{l_i\}$ and $d_{L_i}(l_i) \leq d_e$ for all $1 \leq i \leq |E|$. Hence, by Proposition 2, $\sigma^* \geq \frac{1}{d_e}$. ■

The outcome of the algorithm depends on the sequence of links chosen. One possibility is to choose at each iteration i the link with the smallest interference degree in L_i , i.e., in line 2 of Algorithm 1, we choose l_i such that

$$l_i \leftarrow \underset{k \in L_i}{\operatorname{argmin}} d_{L_i}(k). \quad (4)$$

This choice of l_i tends to produce smaller value of d_e . This procedure can be used to estimate the local-pooling factors of arbitrary network graphs.

C. The worst-case local-pooling factor over a class of graphs

We are often interested in the worst-case efficiency ratio of a scheduling policy for a class of network graphs. This information is useful when the exact network topology is unknown. Let \mathbb{P} be a set of network graph with certain topological property and let $\sigma^+(\mathbb{P})$ and $\gamma^+(\mathbb{P})$ denote the

worst-case local-pooling factor and the worst-case efficiency ratio, respectively, over all graphs in \mathbb{P} , i.e.,

$$\begin{aligned}\sigma^+(\mathbb{P}) &= \inf\{\sigma^*(G) \mid G \in \mathbb{P}\}, \\ \gamma^+(\mathbb{P}) &= \inf\{\gamma^*(G) \mid G \in \mathbb{P}\}.\end{aligned}$$

We have $\sigma^+(\mathbb{P}) = \gamma^+(\mathbb{P})$ from Proposition 1.

We next use the methodology of Section IV-B to derive a condition for a lower bound of $\sigma^+(\mathbb{P})$. Given \mathbb{P} , define $d^+(\mathbb{P})$ to be a positive integer with the following property: *For any $G \in \mathbb{P}$, there must exist a link l^* such that $d(l^*) \leq d^+(\mathbb{P})$ and further, $G \setminus \{l^*\} \in \mathbb{P}$.* We call $d^+(\mathbb{P})$ as the *recurrent interference degree* of \mathbb{P} . The following proposition shows that if we can find such a recurrent interference degree $d^+(\mathbb{P})$, the worst-case efficiency ratio of *GMS* is bounded by $\frac{1}{d^+(\mathbb{P})}$, i.e., $\sigma^+(\mathbb{P}) \geq \frac{1}{d^+(\mathbb{P})}$.

Proposition 3: Given a network graph $G(V, E, I) \in \mathbb{P}$ with a recurrent interference degree $d^+(\mathbb{P})$, the local-pooling factor is bounded by $\sigma^*(G) \geq \frac{1}{d^+(\mathbb{P})}$.

Proof: Proposition 3 can be proven as Lemma 5. Since $G(V, E, I) \in \mathbb{P}$, there exists link $l_1^* \in E$ with $d(l_1^*) \leq d^+(\mathbb{P})$ and $G \setminus \{l_1^*\} \in \mathbb{P}$. Let $L_1 = E$ and $L_2 = E \setminus \{l_1^*\}$. Since $G \setminus \{l_1^*\} \in \mathbb{P}$, there exists link $l_2^* \in L_2$ with $d_{L_2}(l_2^*) \leq d^+(\mathbb{P})$ and $G \setminus \{l_1^*, l_2^*\} \in \mathbb{P}$. Repeating this procedure until no link remains, we obtain a sequence of links $\{l_1^*, l_2^*, \dots, l_{|E|}^*\}$ and a sequence of sets $\{E = L_1, L_2, \dots, L_{|E|+1} = \emptyset\}$ satisfying $L_{i+1} = L_i \setminus \{l_i^*\}$ and $d_{L_i}(l_i^*) \leq d^+(\mathbb{P})$ for all $1 \leq i \leq |E|$. Hence, from Proposition 2, we conclude that $\sigma^*(G) \geq \frac{1}{d^+(\mathbb{P})}$. ■

In the following section, we show how Proposition 3 can be applied to a class of network graphs.

V. TREE NETWORK GRAPHS UNDER THE K -HOP INTERFERENCE MODEL

We first study the efficiency ratio of *GMS* for tree networks. In [18], [19], it has been shown that *GMS* achieves the full capacity in tree networks under the 1-hop interference model. We now show how to use the result in the previous section to prove that *GMS* achieves full capacity for tree network topologies under K -hop interference model. (This result was shown in [20] by using a different approach.) Let \mathbb{T}^K be the set of network graphs whose topology forms a tree and the interference relationship is governed by the K -hop interference constraints. Recall that in the K -hop interference model, any two links within a K -hop distance cannot transmit at the same time.

Proposition 4: *GMS* achieves the full capacity for tree networks under the K -hop network model, i.e., $\sigma^+(\mathbb{T}^K) = 1$.

Proof: It is sufficient to show that $d^+(\mathbb{T}^K) = 1$ from Proposition 3.

Consider a tree network graph $G_t(V, E, I) \in \mathbb{T}^K$. We define the depth of link l in E , denoted by $D(l)$, as the number of hops from link l to the root node of the tree. Let l^* be the link with the largest depth, i.e., $l^* := \arg\max_{l \in E} D(l)$. Since l^* is a leaf link of the tree, $G_t \setminus \{l^*\}$ is still a tree, and thus it belongs to \mathbb{T}^K .

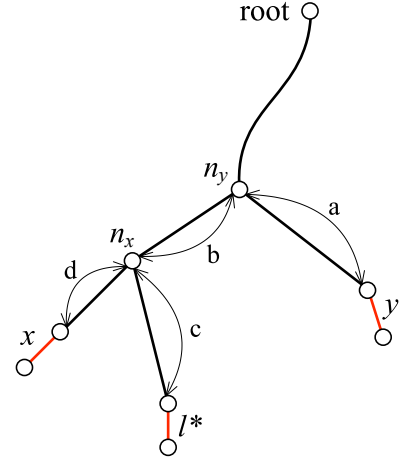


Fig. 2. Tree network graph with the deepest link l^* . Two links $x, y \in I_E(l^*)$ interfere with each other.

We next show that the interference degree $d(l^*)$ of link l^* is 1. It suffices to show that any two links $x, y \in I(l^*)$ interfere with each other. Let n_x (or n_y) denote the closest common parent node of x (or y) and l^* . Note that both n_x and n_y lie on the line from link l^* to the root node. Without loss of generality, we assume that n_y is a parent of n_x as shown in Fig. 2. Let a , b , c , and d denote the number of links placed between link y and node n_y , between node n_y and node n_x , between node n_x and link l^* , and between node n_x and link x , respectively. We have the following constraints.

- $a + b + c \leq K - 1$ since link y interferes with link l^* .
- $d \leq c$ since link l^* has the maximum depth.

We thus have $a + b + d \leq K - 1$. In other words, any two links $x, y \in I(l^*)$ interfere with each other, and hence, $d(l^*) = 1$.

In summary, for a graph $G_t(V, E, I) \in \mathbb{T}^K$, there exists a link $l^* \in E$ with the largest depth and its interference degree is $d(l^*) = 1$. Further, $G_t \setminus \{l^*\} \in \mathbb{T}^K$. Therefore, we conclude that \mathbb{T}^K has a recurrent interference degree $d^+(\mathbb{T}^K) = 1$ and the result of Proposition 4 follows. ■

Proposition 4 show that *GMS* is a throughput-optimal scheduling policy in tree networks under K -hop interference models. However, when the network topology is not a tree, in general *GMS* will not have an efficiency ratio of 1. In fact, whenever $K \geq 2$, we can construct network topologies, in which the efficiency ratio of *GMS* can be arbitrarily small under the K -hop interference model. (We refer readers to our technical report [25] for the construction of these topologies.) As the reader can see in [25], these topologies are somewhat artificial and may not exist in practice. On the other hand, in our prior work [24], we have shown that *GMS* achieves $\frac{\tilde{d}}{2\tilde{d}-1}$ under the 1-hop interference model, where \tilde{d} is the largest node degree of the network graph. This suggests that the worst-case performance limits of *GMS* may be better when there are *additional constraints* on the graph. Therefore, in the next section, we focus on geometric graphs and revisit the question of the worst-case efficiency ratio of *GMS*.

VI. ESTIMATES OF THE LOCAL-POOLING FACTOR FOR GEOMETRIC NETWORK GRAPHS

In this section, we are interested in the performance of *GMS* in geometric network graphs, in which the connectivity between nodes and the interference between links depend on their geometric locations. We assume that nodes lie on a finite two-dimensional space. We assume that two nodes n_i and n_j form a link if their distance $s(n_i, n_j)$ is less than the communication range c , and two links $l_i(n_i^1, n_i^2)$ and $l_j(n_j^1, n_j^2)$ interfere with each other if the distance between any two of the four nodes $\{n_i^1, n_i^2, n_j^1, n_j^2\}$ is less than the interference range r . We say that a geometric network graph operates under the K -hop (interference) model if $r = (K - 1)c$. Chaporkar *et al.* [13] have shown that the efficiency ratio of Maximal Scheduling is bounded by $\frac{1}{8}$ in arbitrary geometric graphs under the 2-hop model, and Sharma *et al.* [15] have shown that it is no smaller than $\frac{1}{49}$ under any K -hop model. In this section, we will show that *GMS* typically has better efficiency ratios than Maximal Scheduling studied in [13], [15].

Our methodology is again based on Proposition 3. Note that the edge links in a geometric graph typically have a smaller interference degree than the links in the middle of the graph. If we can bound the interference degree of some edge links l to a number d , we can then use Proposition 3 to show that the efficiency ratio is $\frac{1}{d}$. We will use the methodology first on the 2-hop model, then on K -hop models.

A. Geometric graphs under the 2-hop model

Let \mathbb{G}_g denote the set of graphs conforming to geometric constraints. Given a geometric network graph $G_g(V, E, I) \in \mathbb{G}_g$, we can assign a two-dimensional coordinate (x, y) for each node. We say that node A is to the left of node B if A 's x -coordinate is less than B 's x -coordinate. Then for each link l , we can define the left end-point (i.e., node) $n^L(l)$ and right end-point $n^R(l)$. If the two end-points have the same x -coordinate, we assign them to the left or the right arbitrarily. We consider the set of all right nodes of all links $N_V^R = \{n^R(l) \in V \mid l \in E\}$. We say node n in N_V^R is located at the edge if there exist a line through node n such that all other nodes in N_V^R are in the *interior* of one of the half-planes. Note that since the graph is on a two-dimensional finite space, there always exist some right nodes that are on the edge. Let the left-most node n_V^R denote the edge node that has the smallest x -coordinate in N_V^R . All other nodes of N_V^R are in the interior of a half-plane (see Fig. 3) whose boundary is through n_V^R . We define a left-most link as a link whose right node is n_V^R . Assuming that every node in V is connected by some links of E (otherwise, we can remove the node from V), we can always find at least one left-most link l^* in E .

Let \mathbb{G}_g^K denote the set of geometric network graphs under the K -hop model. The following lemma specifies the performance limits of *GMS* in geometric network graphs under the 2-hop model.

Proposition 5: The worst-case efficiency ratio of *GMS* in geometric graphs under the 2-hop model is $\frac{1}{6}$, i.e., $\gamma^+(\mathbb{G}_g^2) \geq \frac{1}{6}$.

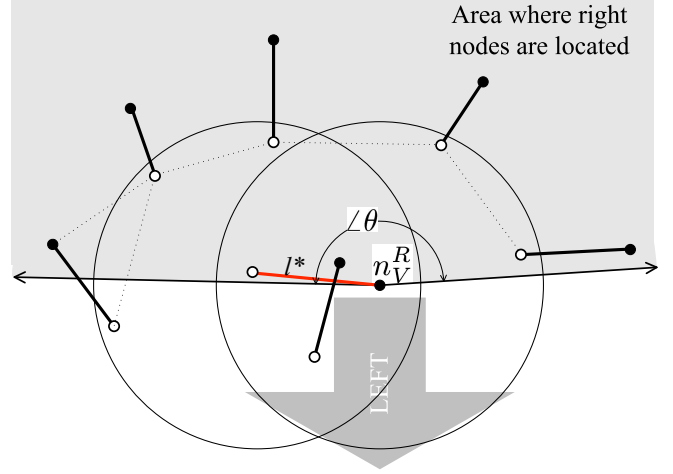


Fig. 3. Geometric network graph under the 2-hop model. Downward is the left direction of the coordinate system as indicated by a big arrow. For each link, its left node is colored in white and its right node in black. The node n_V^R is the left-most right node, the link l^* is the left-most link. Note that all other right nodes must be within an angle of less than 180° from n_V^R . This figure shows how 6 other links can be placed within the interference range of l^* and they do not interfere with each other. Note that each node of the 6 links must be outside an interference range of c of each other, and further, their right node must be inside an angle less than 180° from n_V^R .

From Proposition 1, it suffices to show $\sigma^+(\mathbb{G}_g^2) \geq \frac{1}{6}$. Since a geometric network graph $G(V, E, I) \in \mathbb{G}_g^2$ has at least one left-most link l^* and $G(V, E, I) \setminus \{l^*\} \in \mathbb{G}_g^2$, it suffices to show that $d(l^*) \leq 6$. It then follows that \mathbb{G}_g^2 has a recurrent interference degree $d^+(\mathbb{G}_g^2) \geq 6$, and $\sigma^+(\mathbb{G}_g^2) \leq \frac{1}{6}$ by Proposition 3. Due to lack of space, we refer the readers to [25] for the detailed proof of $d(l^*) \leq 6$. In Fig. 3, we show how 6 links that do not interfere with each other can be placed within the interference range of l^* . In [25], we show that this is the largest number of non-interfering links one can put in $I(l^*)$. Recall that Maximal Scheduling achieves an efficiency ratio of $\frac{1}{8}$ in geometric graphs under the 2-hop model. Our result shows that *GMS* indeed outperforms Maximal Scheduling. In the rest section, we show that the performance gap is even bigger for $K \geq 2$.

B. Geometric graphs under K -hop models

It is well-known in the literature that the worst-case efficiency ratio of Maximal Scheduling in geometric graphs degrades when K increases [13], [15]. We next show that this is not the case for *GMS*. In fact, the worst-case efficiency ratio of *GMS* tends to increase as K increases. In the next lemma, we compare two graphs $G_1(V, E_1, I) \in \mathbb{G}_g^{K_1}$ and $G_2(V, E_2, I) \in \mathbb{G}_g^{K_2}$ with $K_1 > K_2$. Note that both G_1 and G_2 have the same set of nodes V and have the same interference range r . However, the communication range of G_1 is $c_1 = \frac{r}{K_1 - 1}$, which is smaller than that of G_2 , (i.e., $c_2 = \frac{r}{K_2 - 1}$).

Proposition 6: Given a set of nodes V and their location, if $K_1 > K_2$, the local-pooling factor of the network graph $G_1(V, E_1, I) \in \mathbb{G}_g^{K_1}$ is no smaller than the local-pooling factor

of the network graph $G_2(V, E_2, I) \in \mathbb{G}_g^{K_2}$, i.e., $\sigma^*(G_1) \geq \sigma^*(G_2)$.

Proof: Note that the set of nodes is the same and the interference range is also identical for both G_1 and G_2 . Suppose that we have a subset $L \subset E_1$ in G_1 and two maximal scheduling vectors $\vec{\mu}, \vec{\nu} \in Co(\mathcal{M}_L)$ such that $\sigma\vec{\mu} \succeq \vec{\nu}$. If the same vectors $\vec{\mu}, \vec{\nu}$ are also valid maximal scheduling vectors in G_2 , then we have $\sigma^*(G_1) \geq \sigma^*(G_2)$ from the definition of the local-pooling factor. Toward this end, we first show that two maximal scheduling vectors in a subset of links in G_1 are also valid maximal scheduling vectors in G_2 .

Since the interference range is fixed, G_1 has a smaller communication range than G_2 . Hence, any link in G_1 is also a link in G_2 , i.e., $E_1 \subset E_2$. We consider the subset $L \subset E_1$. From $E_1 \subset E_2$, we have $L \subset E_2$. Further, since the interference range is identical, the interference constraints between links in L do not change. Specifically, two links in L that interfere with each other under the K_1 -hop model also interfere under the K_2 -hop model. Hence, maximal scheduling vectors $\vec{\mu}, \vec{\nu}$ in L under the K_1 -hop model are valid maximal scheduling vectors in L under the K_2 -hop model.

Therefore, if there exist two maximal scheduling vectors $\vec{\mu}, \vec{\nu} \in Co(\mathcal{M}_L)$ satisfying $\sigma\vec{\mu} \succeq \vec{\nu}$ and a subset L of links in G_1 , the same maximal scheduling vectors and the same subset L are valid for G_2 . This implies that the local-pooling factor under the K_2 -hop model is no greater than σ . Hence, $\sigma^*(G_1) \geq \sigma^*(G_2)$. ■

Combining Proposition 5 and 6, we obtain the following result.

Theorem 1: The worst-case efficiency ratio of *GMS* in geometric graphs under K -hop models is no smaller than $\frac{1}{6}$, i.e., $\gamma^+(\mathbb{G}_g^K) \geq \frac{1}{6}$ for $K \geq 2$.

How tight is this bound? We next present a network graph in \mathbb{G}_g^K with local-pooling factor of $\frac{1}{3}$.

Lemma 6: There exists a large number K_0 such that for all $K > K_0$ and σ arbitrarily close to $\frac{1}{3}$, some geometric graph $G(V, E, I) \in \mathbb{G}_g^K$ has the local-pooling factor no larger than σ , i.e., $\sigma^*(G) \leq \sigma$.

It suffices to construct a graph such that there exists two vectors $\vec{\mu}, \vec{\nu} \in Co(\mathcal{M}_E)$ that satisfy $\sigma\vec{\mu} \succeq \vec{\nu}$. Due to lack of space, we sketch the main idea in this paper. For the detailed proof, we refer the readers to [25].

We construct a network graph $G(V, E, I) \in \mathbb{G}_g^K$ as follows. First, when K is very large, we can think of a link as a point and its interference range as a circle with radius r because the communication range is close to zero. Second, we form two set of links L_1 and L_2 . Suppose that $|L_1|$ and $|L_2|$ are finite but very large, and $|L_1| = |L_2|$. Remember that we approximate a link by a point. The links from L_1 form a circle C_1 with radius R at origin O , and links from L_2 form another circle C_2 with radius $R + \frac{\sqrt{3}}{2}r$ at the same origin O . In Fig. 4, we show the small arcs from the two circles. Since the radius R is very large, the two arcs can be approximated by two parallel lines. Since $|L_1|$ and $|L_2|$ are very large, there exists a link at almost every point of the two arcs.

We now find two maximal scheduling vectors $\vec{\mu}, \vec{\nu} \in Co(\mathcal{M}_{L_1 \cup L_2})$. To form $\vec{\mu}$, take any maximal schedules of the form in Fig. 4(a), where active points (i.e., links) are colored in black. Since $|L_1|$ and $|L_2|$ are very large, there will be a large number of such maximal schedules and we produce $\vec{\mu}$ by taking the convex combination with equal weights of these schedules. Similarly, to form $\vec{\nu}$, we take maximal schedules of the form in Fig. 4(b) and produce $\vec{\nu}$ by taking the convex combination with equal weights of them. Clearly, the maximal schedules in Fig. 4(a) are more efficient than those in Fig. 4(b). We next show that the ratio of $\vec{\mu}, \vec{\nu}$ is close to $\frac{1}{3}$.

Assuming that points (i.e., links) are uniformly distributed on C_1 and C_2 , then the distance between activated links in Fig. 4(a) is approximately $\frac{1}{3}$ of the distances between activated links in Fig. 4(b). Hence, the schedules that form $\vec{\mu}$ serves 3 times more links than the schedules that form $\vec{\nu}$. We thus obtain that $\frac{1}{3}\vec{\mu}$ is approximately equal to $\vec{\nu}$. In the technical report [25], we show this with a more formal proof and conclude that $\sigma\vec{\mu} \succeq \vec{\nu}$ with σ close to $\frac{1}{3}$.

Lemma 6 leads to the following corollary.

Corollary 1: There exists a geometric network graph $G(V, E, I) \in \mathbb{G}_g^K$ with $K \geq 2$, in which the efficiency ratio of *GMS* is no more than $\frac{1}{3}$.

Proof: From Lemma 6, there exist a number K_0 and graphs $G(V, E, I) \in \mathbb{G}_g^K$ for all $K \geq K_0$ such that $\sigma^*(G) \leq \frac{1}{3}$. By Proposition 6, we also have network graphs $G_K \in \mathbb{G}_g^K$ for all $K \leq K_0$ such that $\sigma(G_K) \leq \sigma(G_{K_0}) \leq \frac{1}{3}$. Therefore, we have $\sigma^+(\mathbb{G}_g^K) \leq \frac{1}{3}$ for all $K \geq 2$. ■

From Theorem 1 and Corollary 1, we can bound the worst-case efficiency ratio of *GMS* in arbitrary geometric network graphs under the K -hop model as

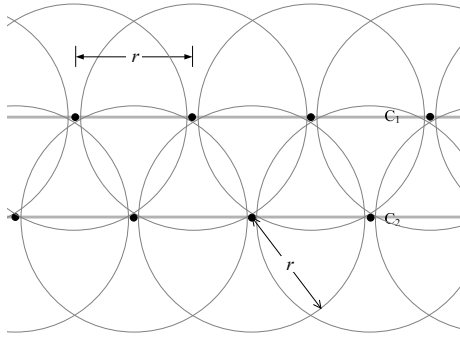
$$\frac{1}{6} \leq \gamma^+(\mathbb{G}_g^K) \leq \frac{1}{3}. \quad (5)$$

Recall that in [15], the worst-case efficiency ratio of *GMS* is $\frac{1}{49}$, if $K \geq 2$. Hence, our result indicates that *GMS* significantly outperforms Maximal Scheduling.

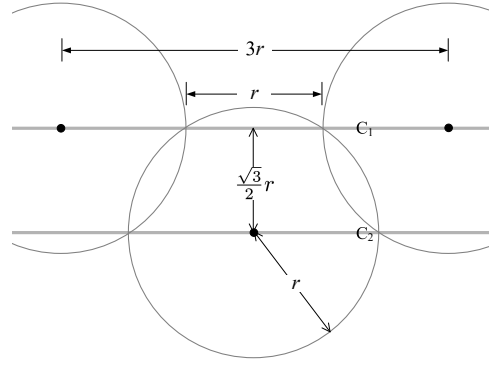
VII. CONCLUSION

In this paper, we have provided new analytical results on the achievable performance of *GMS* for a large class of network topologies under general K -hop interference models. We first provide an equivalent characterization of the efficiency ratio of *GMS* through the local-pooling factor of the underlying graph. We then provide an iterative procedure to estimate the local-pooling factor of arbitrary graphs. This new procedure allows us to estimate the worst-case efficiency ratio of *GMS* for a large set of network graphs and interference models. In particular, we observe that *GMS* achieves the optimal capacity region in tree networks under the K -hop interference model. Further, in geometric network topologies under the K -hop interference model, we show that the worst-case efficiency ratio of *GMS* increases with K , and is between $\frac{1}{6}$ and $\frac{1}{3}$.

For future work, there remain many interesting open problems in these directions. For example, the bounds of the worst-case efficiency ratio in geometric network graphs are still not



(a) An instance of (dense) maximal schedules $\vec{M}_i; \vec{\mu} \in Co(\{\vec{M}_i\})$



(b) An instance of (sparse) maximal schedules $\vec{M}_j; \vec{\nu} \in Co(\{\vec{M}_j\})$

Fig. 4. A geometric network graph $G(V, E, I) \in \mathbb{G}_g^K$ and $\vec{\mu}, \vec{\nu} \in Co(\mathcal{M}_E)$ such that $\frac{1}{3}\vec{\mu} \succeq \vec{\nu}$. With $K \rightarrow \infty$, we assume that a link is a point and its interference range is a circle with radius r . Figures illustrate an instance of maximal schedules from $\vec{\mu}$ and $\vec{\nu}$, respectively. Note that since links are uniformly and closely placed on circles C_1 and C_2 (a small fraction of them is shown in the figures), the interference range of active links in each maximal schedule must cover C_1 and C_2 . Let $\vec{\mu}$ consist of dense maximal schedules and let $\vec{\nu}$ consist of sparse maximal schedules. From the uniform placement of (finite) links on C_1 and C_2 , the time required to serve all links for a unit time is determined by the distance between two neighboring active links in C_1 (or C_2). Since the distance is r in dense maximal schedules and $3r$ in sparse maximal schedules, we have $\frac{1}{3}\vec{\mu} \succeq \vec{\nu}$.

tight. Further, we may study interference models other than the geometric model.

REFERENCES

- [1] X. Lin and N. B. Shroff, "The Impact of Imperfect Scheduling on Cross-Layer Congestion Control in Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 14, no. 2, pp. 302–315, April 2006.
- [2] L. Tassiulas and A. Ephremides, "Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximal Throughput in Multihop Radio Networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1948, December 1992.
- [3] C. Joo and N. B. Shroff, "Performance of Random Access Scheduling Schemes in Multi-hop Wireless Networks," in *IEEE INFOCOM*, May 2007.
- [4] X. Lin and S. Rasool, "Constant-Time Distributed Scheduling Policies for Ad Hoc Wireless Networks," in *IEEE CDC*, December 2006.
- [5] L. Tassiulas, "Linear Complexity Algorithms for Maximum Throughput in Radio Networks and Input Queued Switches," in *IEEE INFOCOM*, April 1998.
- [6] G. Sharma, C. Joo, and N. B. Shroff, "Distributed Scheduling Schemes for Throughput Guarantees in Wireless Networks," in the *44th Annual Allerton Conference on Communications, Control, and Computing*, September 2006.
- [7] A. Gupta, X. Lin, and R. Srikant, "Low-Complexity Distributed Scheduling Algorithms for Wireless Networks," in *IEEE INFOCOM*, May 2007.
- [8] S. Sanghavi, L. Bui, and R. Srikant, "Distributed Link Scheduling with Constant Overhead," in *ACM SIGMETRICS*, June 2007.
- [9] E. Modiano, D. Shah, and G. Zussman, "Maximizing Throughput in Wireless Networks via Gossiping," *Sigmetrics Performance Evaluation Review*, vol. 34, no. 1, pp. 27–38, 2006.
- [10] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, 1982.
- [11] B. Hajek and G. Sasaki, "Link Scheduling in Polynomial Time," *IEEE Transactions on Information Theory*, vol. 34, no. 5, September 1988.
- [12] S. Sarkar and L. Tassiulas, "End-to-end Bandwidth Guarantees Through Fair Local Spectrum Share in Wireless Ad-hoc Networks," in *IEEE CDC*, December 2003.
- [13] P. Chaporkar, K. Kar, and S. Sarkar, "Throughput Guarantees in Maximal Scheduling in Wireless Networks," in the *43rd Annual Allerton Conference on Communication, Control and Computing*, September 2005.
- [14] X. Wu and R. Srikant, "Bounds on the Capacity Region of Multi-hop Wireless Networks Under Distributed Greedy Scheduling," in *IEEE INFOCOM*, April 2006.
- [15] G. Sharma, N. B. Shroff, and R. R. Mazumdar, "On the Complexity of Scheduling in Wireless Networks," in *ACM Mobicom*, September 2006.
- [16] E. Leonardi, M. Mellia, F. Neri, and M. A. Marsan, "On the Stability of Input-Queued Switches with Speed-Up," *IEEE/ACM Transactions on Networking*, vol. 9, no. 1, February 2001.
- [17] N. McKeown, "Scheduling Algorithms for Input-Queued Cell Switches," Ph.D. dissertation, University of California at Berkeley, 1995.
- [18] A. Dimakis and J. Walrand, "Sufficient Conditions for Stability of Longest-Queue-First Scheduling: Second-order Properties using Fluid Limits," *Advances in Applied Probability*, vol. 38, no. 2, pp. 505–521, 2006.
- [19] A. Brzezinski, G. Zussman, and E. Modiano, "Enabling Distributed Throughput Maximization in Wireless Mesh Networks: A Partitioning Approach," in *ACM Mobicom*. New York, NY, USA: ACM Press, 2006, pp. 26–37.
- [20] G. Zussman, A. Brzezinski, and E. Modiano, "Multihop Local Pooling for Distributed Throughput Maximization in Wireless Networks," MIT/LIDS, Tech. Rep. 2719, July 2006. [Online]. Available: <http://stuff.mit.edu/people/gilz/pub.html>
- [21] J.-H. Heopman, "Simple Distributed Weighted Matchings," eprint, October 2004. [Online]. Available: <http://arxiv.org/abs/cs/0410047>
- [22] M. J. Neely, E. Modiano, and C. E. Rohrs, "Power Allocation and Routing in Multibeam Satellites with Time-varying Channels," *IEEE/ACM Transactions on Networking*, vol. 11, no. 1, pp. 138–152, 2003.
- [23] R. L. Cruz and A. V. Santhanam, "Optimal Routing, Link Scheduling and Power Control in Multi-hop Wireless Networks," in *IEEE INFOCOM*, San Francisco, April 2003.
- [24] C. Joo, X. Lin, and N. B. Shroff, "Performance Limits of Greedy Maximal Matching in Multi-hop Wireless Networks," in *IEEE CDC*, December 2007.
- [25] —, "Understanding the Capacity Region of the Greedy Maximal Scheduling Algorithm in Multi-hop Wireless Networks," Purdue University, Tech. Rep., 2007. [Online]. Available: <http://netlab.snu.ac.kr/~cjoo>