Longest-Queue-First Scheduling under SINR Interference Model

Long Bao Le
Massachusetts Institute of Technology
longble@mit.edu

Eytan Modiano
Massachusetts Institute of Technology
modiano@mit.edu

Changhee Joo
Korea University of Technology and Education
cjoo@kut.ac.kr

Ness B. Shroff
The Ohio State University
shroff@ece.osu.edu

ABSTRACT

We investigate the performance of longest-queue-first (LQF) scheduling (i.e., greedy maximal scheduling) for wireless networks under the SINR interference model. This interference model takes network geometry and the cumulative interference effect into account, which, therefore, capture the wireless interference more precisely than binary interference models. By employing the ρ-local pooling technique, we show that LQF scheduling achieves zero throughput in the worst case. We then propose a novel technique to localize interference which enables us to decentralize the LQF scheduling while preventing it from having vanishing throughput in all network topologies. We characterize the maximum throughput region under interference localization and present a distributed LQF scheduling algorithm. Finally, we present numerical results to illustrate the usefulness and to validate the theory developed in the paper.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Theory

Keywords

Wireless scheduling, greedy maximal scheduling, longest-queue-first scheduling, throughput region, binary interference model, SINR interference model

1. INTRODUCTION

Scheduling has been recognized to be an important problem in designing cross-layer protocols for multihop wireless

networks. Developing an efficient scheduling algorithm is challenging due to complex interference coupling among simultaneous transmissions in the network. As a consequence, most existing works on wireless scheduling assume simplistic graph-based or binary interference models where transmissions on two different links are predetermined to conflict with each other independently of the transmissions of other neighboring links [2, 3, 5, 7–10, 12–14, 16, 17, 19]. In fact, graph-based interference models over-simplify interference coupling because interference experienced at a particular link is indeed equal to the total cumulative interference from all concurrent transmissions in the network.

In general, an efficient scheduling algorithm for multihop wireless networks aims to exploit spatial reuse to maximize the number of simultaneous transmissions in the network. This would result in high overall network throughput. When wireless nodes transmit at a fixed rate, there is a minimum required signal-to-interference-plus-noise ratio (SINR) for successfully decoding received signals [1, 6]. Although power control could improve network throughput, the optimal joint scheduling and power control problem can only be solved in some special cases, and it is usually difficult for decentralized implementation [4, 15].

Wireless scheduling is a difficult problem even with fixed power and binary interference models for which it has been shown to be NP-hard [16]. Existing works in the literature on wireless scheduling consider different optimization measures and assume different interference models. Some common optimization measures include finding a minimum-length schedule for a given traffic demand [1, 18], achieving optimal scaling laws for network capacity [6], and achieving full [2,5,7,10–17,19] or a fraction of the maximum stability (throughput) region [3,8,9]. Moreover, most existing works on wireless scheduling under the stability framework of [17] assume the graph-based or binary interference models. In this paper, we consider the scheduling problem under the practical SINR interference model.

In the seminal paper [17], Tassiulas and Ephremides show that the celebrated maximum weight scheduling (MWS) achieves 100% throughput. However, MWS is difficult to implement in a distributed manner, which is, however, required in most wireless networks. In [3], a simpler maximal scheduling is investigated where it is shown to achieve at least $1/d_G$ of the throughput region under certain binary interference models where d_G is the maximum interference degree for network

graph G. In particular, maximal scheduling achieves at least $1/2$ and $1/8$ of the throughput region for geometric network graphs under 1-hop and 2-hop interference models, respectively.

Another important scheduling policy which has been observed to achieve 100% throughput in most practical wireless networks is longest-queue-first scheduling (i.e., greedy maximal scheduling). There are several recent works that investigate the performance of LQF scheduling under different binary interference models [5, 7, 12]. Design of practical distributed algorithms for LQF scheduling under the k-hop interference model is done in [2, 10]. Specifically, Dimakis and Walrand show that LQF scheduling achieves 100% throughput if the network satisfies the so-called "local pooling" condition [5]. In [7], a deeper investigation of LQF scheduling is performed where the authors show that LQF scheduling achieves at least $1/6$ of the throughput region for geometric network graphs under the k-hop interference model. In [10], it is shown that LQF scheduling indeed achieves at least $1/4$ of the throughput region under the 2-hop interference model for wireless networks with at most 20 nodes. Unfortunately, these results strongly depend on the binary interference structure, which could not be applied to the more realistic SINR interference model.

In this paper, we investigate the performance and design practical decentralized algorithms for LQF scheduling under the SINR interference model. Specifically, we make the following contributions.

- We use the σ-local pooling notion developed in [7, 12] to show that LQF scheduling achieves zero throughput in the worst case. In addition, we present a sufficient condition for a network to achieve 100% throughput under LQF scheduling.

- We show that there is a finite coordination neighborhood around the receiver of each link such that the total interference from other links outside this neighborhood is negligible. Based on this result, we propose a novel interference localization technique that enables us to decentralize the LQF scheduling.

- We characterize the throughput region under interference localization. We show that a non-vanishing fraction of the throughput region with interference localization can be achieved by LQF scheduling.

- We propose a distributed LQF scheduling algorithm with linear complexity under interference localization. The proposed distributed LQF algorithm returns the same maximal schedule as the centralized LQF algorithm.

- We present numerical results to illustrate the different performance bounds derived in the paper and the usefulness of the interference localization technique.

There is one key difference between the SINR interference model with interference localization and other binary interference models such as the protocol model [6], the k-hop interference model [16], and the 802.11-based interference model [18]. Specifically, cumulative interference from a local neighborhood is considered under the SINR interference model with interference localization while it is not in any of the binary interference models.

The remainder of this paper is organized as follows. In Section 2, we describe the system model. In Section 3, we investigate the performance of LQF scheduling under the SINR interference model. We discuss the interference localization for the SINR interference model in Section 4 and study the performance of LQF scheduling under interference localization in Section 5. Practical scheduling designs are considered in Section 6. Some numerical results are presented in Section 7 followed by conclusions in Section 8.

2. SYSTEM MODEL

Consider a wireless network which is modeled as a graph $G = (V, E)$ where V is the set of nodes and E is the set of links. Let $|E|$ denote the number of links in the network. We assume that all transmissions use the same power level P. Also, let the ambient noise power measured in the signal bandwidth at the receiver of link l be N_l and let G_{lk} be the channel gain from the transmitter of link k to the receiver of link l. Now, suppose that the channel gain G_{lk} depends on the corresponding distance d_{lk} between the transmitter of link k and the receiver of link l as $G_{lk} = d_{lk}^{-\alpha}$ where α is the path loss exponent. From a communication perspective, if a receiver treats interference as noise, the SINR at the receiver should be large enough for successfully decoding of the signal. As a result of this, we define the SINR interference model as follows.

Definition 1. A feasible schedule under the SINR interference model is a set of activated links such that minimum SINR requirements of all activated links are satisfied. Specifically, let S denote a set of activated links that forms a feasible schedule. Then, we have

$$\mathsf{SINR}_l \triangleq \frac{PG_{ll}}{\sum_{k \in S, k \neq l} G_{lk}P + N_l} \geq \beta, \quad \forall l \in S \qquad (1)$$

where β is a predetermined threshold required to achieve a certain desired bit error rate.

In the following, we will use the term *activation set* to refer to a particular schedule, which may or may not be feasible. In addition, a schedule will be denoted either as a set S of activated links or a vector $\vec{S}$ of dimension $|E|$ (i.e., the number of network links) where its k-th element $\vec{S}(k) = 1$ if link k is activated or $\vec{S}(k) = 0$, otherwise. We assume that a wireless link l exists if its corresponding transmitting and receiving nodes want to communicate with each other and they have relative distance satisfying $d_{\mathsf{min}} \leq d_l \leq d_{\mathsf{max}}$. Here, d_{max} must be smaller than the maximum distance such that the minimum SINR is satisfied. Suppose that the power of ambient noise measured in the signal bandwidth is N_0 for all links, then d_{max} is upper-bounded by $d_{\mathsf{max}} \leq (P/(\beta N_0))^{1/\alpha}$. Also, d_{min} is the minimum distance between any two nodes that want to communicate with each other (i.e., $d_{\mathsf{min}} = \min_{l \in E} d_l$).

We assume time-slotted wireless systems where time slots are of unit length. It is assumed that when a link is scheduled, one packet can be transmitted in each time slot. We consider single-hop flows where each flow carries traffic on one wireless link. We assume that packets arrive at the transmitting end of each link l according to a stationary stochastic process with average arrival rate λ_l. Wireless links are scheduled in each time slot according to the SINR interference model described above.

In this paper, we are interested in investigating the performance of LQF scheduling under the SINR interference model. The performance measure that we consider is the guaranteed fraction of the maximum throughput region (or throughput region for brevity) that a particular scheduling policy can achieve [17]. The definitions of throughput region, and scheduling efficiency ratios are given in the following.

Definition 2 ([17]). The throughput region contains all possible arrival rate vectors such that there exists some scheduling policy that can stabilize the network (average queue lengths of all queues in the network are finite).

In [17], the throughput region is well characterized. Specifically, the throughput region can be described as,

$$\Lambda \triangleq \left\{ \vec{\lambda} : \vec{\lambda} \preceq \vec{\phi}, \text{ for some } \vec{\phi} \in \text{Co}(\Omega) \right\} \tag{2}$$

where $\vec{\lambda}$ denotes the traffic arrival rate vector whose l-th element λ_l is the traffic arrival rate of link l, Ω denotes the set of all feasible maximal schedules, $\text{Co}(\Omega)$ denotes the convex hull of Ω, and $\preceq$ denotes element-wise inequality. In [17], it has been shown that MWS can stabilize the network for all arrival rate vectors strictly inside the throughput region where MWS activates a maximal schedule with the largest total queue length in each time slot. However, MWS is difficult to implement even under the binary interference model. It is, therefore, desirable to look for a simple and easy-to-implement scheduling policy that achieves a guaranteed fraction of the throughput region. One such strategy is to find a *maximal schedule* whose definition is as follows:

Definition 3. A maximal schedule S is a feasible schedule such that if we add any link $l \notin S$ to the schedule S (i.e., link l is not currently activated by the schedule S) then the SINR constraint of at least one activated link in schedule S is violated (i.e., its SINR becomes smaller than β).

In this paper, we consider the well-known policy that finds a maximal schedule in a greedy manner, called the longest-queue-first (LQF) scheduling policy. LQF scheduling makes scheduling decisions based on queue length information as follows: it starts with an empty schedule. Then, it adds the link with the largest queue length to the schedule. Then, it looks for the link with the largest queue length among the remaining links. This chosen link will be added to the schedule if this addition creates a feasible schedule (i.e., the set of added links that satisfy the SINR constraints) or it is discarded otherwise. This process continues until no link is left. Note that given the queue length vector, the schedule obtained by LQF scheduling is maximal and unique if the queue lengths of all links are different.

In general, LQF scheduling does not maintain network stability for all traffic arrival rates inside the throughput region. However, simulation results often show that LQF scheduling achieves maximum throughput in many wireless networks [8]. In the following, we give a definition of the efficiency ratio of a scheduling policy [7].

Definition 4. The efficiency ratio $\gamma(G)$ of a scheduling policy for a network graph G is the supremum of all γ such that the scheduling policy stabilizes all traffic arrival rates that lie inside γ fraction of the throughput region, i.e.,

$$\gamma(G) \triangleq \sup \left\{ \gamma | \text{the network is stable for all } \vec{\lambda} \in \gamma\Lambda \right\}. \tag{3}$$

In practice, network graphs may have different structure and topology. Therefore, it is also useful to quantify the worst-case efficiency ratio of a scheduling policy.

Definition 5. The worst-case efficiency ratio γ^* of a scheduling policy is the infimum of all efficiency ratios $\gamma(G)$ for all possible network graphs G, i.e.,

$$\gamma^* \triangleq \inf_G \gamma(G). \tag{4}$$

In the following, we investigate the efficiency ratio (both worst-case and for some specific network G) of LQF scheduling under the SINR interference model.

3. PERFORMANCE OF LQF SCHEDULING UNDER SINR INTERFERENCE MODEL

We investigate the performance of LQF scheduling under the SINR interference model using the σ-local pooling technique [7, 12]. In particular, there are three different notions of local pooling factors, namely local pooling factors for a link, a set of links or the whole network. In the following, we will also refer to these factors as link, set, and network local pooling factors, respectively. The local pooling factor for network G, which is equal to the minimum of local pooling factors of all links, is equal to the efficiency ratio of LQF scheduling. A set local pooling factor can be calculated by a primal or dual formulation of a special optimization problem [12]. More detailed description of the σ-local pooling technique is given in Appendix A. In the following, we will present the worst-case performance and a sufficient condition for LQF scheduling to achieve 100% throughput using these σ-local pooling notions.

In general, it would be useful to know properties of network topologies where LQF scheduling achieves 100% throughput. In addition, if a particular network has the network local pooling factor strictly smaller than one, then it is useful to calculate or estimate its network local pooling factor, which is also the efficiency ratio of LQF scheduling. In theory, this can be done by calculating local pooling factors of all possible subsets of links. Unfortunately, in order to calculate a set local pooling factor one must generate all possible maximal schedules of that set, which is a very complex task for a large set.

It can be observed that the set local pooling factor represents the scaling factor between the most compact time-sharing and the least compact time-sharing of maximal schedules [12]. Intuitively, the most compact time-sharing is achieved by a convex combination of maximal schedules with large number of activated links while the least compact time-sharing is achieved by a convex combination of maximal schedules with small number of activated links. Therefore, the worst-case performance of LQF scheduling may be quite poor for certain network topologies. We formally state this worst-case performance in the following theorem.

Theorem 1. The efficiency ratio of LQF scheduling under the SINR interference model is zero in the worst case.

We will prove Theorem 1 by using results in the following lemma. Let Ω_g be a set of maximal schedules that covers the whole network, i.e., let $\vec{S}_{\text{tot}}^{\text{g}} = \sum_{i \in \Omega_g} \vec{S}_i$ then $\vec{S}_{\text{tot}}^{\text{g}}(k) \geq 1, \forall k \in E$ (i.e., every link is included in some

schedule $\vec{S_i}$). And let Ω_b be a another set of maximal schedules also covering the whole network and let $\vec{S}^{\mathsf{b}}_{\mathsf{tot}} = \sum_{i \in \Omega_b} \vec{S_i}$. In addition, let $K_1 = |\Omega_g|$, $K_2 = |\Omega_b|$ (i.e., the number of maximal schedules in the corresponding sets of schedules), and $k^* = \min\left\{h \geq 1, h \in Z | h\vec{S}^{\mathsf{g}}_{\mathsf{tot}} \succeq \vec{S}^{\mathsf{b}}_{\mathsf{tot}}\right\}$. Then, we have the following result.

Lemma 1. Given a network G and parameters K_1, K_2, and k^* defined above, the network local pooling factor of G satisfies $\sigma^*(G) = \gamma(G) \leq k^* K_1 / K_2 \triangleq \sigma^*_{\mathsf{ub}}(G)$.

PROOF. The proof is given in Appendix B. □

Lemma 1 implies that given a network G, we have $\sigma^*(G) \leq \min_{\{K_1, K_2, k^*\}} k^* K_1 / K_2$ where K_1, K_2, and k^* correspond to any two sets of maximal schedules as described above. This will allow us to obtain an upper bound for the efficiency ratio of LQF scheduling. We are now ready to prove Theorem 1.

PROOF. We prove Theorem 1 by showing that there exists a class of network topologies whose network local pooling factors can be made arbitrarily small. Specifically, consider a network graph G_K with $|E| = 2K$ links such that it can be decomposed into either 2 large maximal schedules (each with K links) or K small maximal schedules (each with 2 links) and schedules of these two schedule sets do not share any common links within each set. The structure of this particular network is illustrated in Fig. 1. By applying the result of Lemma 1 to this network with $K_1 = 2$, $K_2 = K$, and $k^* = 1$, the network local pooling factor (equal to the efficiency ratio of LQF scheduling) can be upper bounded as $\sigma^*(G_K) = \gamma(G_K) \leq \sigma^*_{\mathsf{ub}}(G_K) = 2/K$. Therefore, the efficiency ratio of this network under LQF scheduling tends to zero as $K \to \infty$.

Physical construction of such a wireless network can be outlined as follows. For each maximal schedule with two links we place these links so that their SINRs are exactly equal to β when they are activated simultaneously. In addition, each large maximal schedule of K links is carefully constructed such that each receiver of the links has an SINR no smaller than β. □
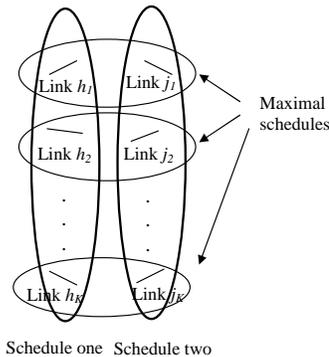

Figure 1: Network example for which the efficiency ratio of LQF scheduling can be vanishingly small under the SINR interference model.

It can be observed from the proof of Theorem 1 that the key reason for LQF scheduling to have small efficiency ratios in certain network topologies is the *scheduling starvation*. Specifically, consider network topologies where there are many maximal schedules of small sizes. Then, if traffic arrival patterns are such that LQF scheduling activates only maximal schedules of small sizes although good large maximal schedules are available then LQF scheduling may achieve very poor throughput performance. Although we cannot guarantee a non-vanishing performance lower bound for LQF scheduling in general, it is very unlikely that both "bad" network topology and traffic arrival pattern occur simultaneously in practice. In other words, LQF scheduling may still work well in most practical network topologies and traffic arrival patterns.

Now we provide a sufficient condition for LQF scheduling to achieve 100% throughput.

Lemma 2. A particular network G has a network local pooling factor $\sigma^*_G = 1$ if for any subset of links $L \subseteq E$, the number of activated links in L for any maximal schedules is the same.

PROOF. This lemma can be proved by showing that local pooling factors of any sets of links $L \subseteq E$ are equal to one. For brevity, the detailed proof is omitted. □

In the following, we will propose a novel technique to localize interference for the SINR interference model. This interference localization enables us to prevent LQF scheduling from having a vanishing small efficiency ratio and to decentralize the LQF scheduling.

4. INTERFERENCE LOCALIZATION

In the SINR interference model, any activated link will create interference to all other activated links in the network. However, the average channel gain between any two links attenuates rapidly over distance. Therefore, only simultaneous transmissions in an immediate neighborhood around the receiver of a particular link may create significant cumulative interference. In addition, the number of concurrent feasible transmissions in a pre-determined neighborhood is limited because each transmission needs to "reserve" some space around it to limit interference from neighboring transmissions. In the following, we exploit these facts to determine a neighborhood for each link such that interference beyond this neighborhood only has negligible impacts on its received signal. In particular, we propose a technique to localize interference while still maintaining the scheduling feasibility.

In fact, we will show that there exists a neighborhood around the receiver of each link such that it can "truncate" (i.e., ignore) interference beyond this neighborhood. For simplicity, we assume an interference-limited network so the effect of ambient noise can be ignored. Consider a particular link l with length d_l. Then, the maximum interference that can be tolerated at the receiver of link l is

$$I^{\mathsf{max}}_l \triangleq \frac{P d_l^{-\alpha}}{\beta}. \qquad (5)$$

Suppose that link l only performs scheduling coordination inside a circle with radius $K_l d_l$ centered around the receiver of link l. We will refer to this circular area as the *interference neighborhood* of link l. Let the set of links whose transmitting ends lie on the boundary or inside this circle be $\Phi_l(K_l)$. Now, given an activation set S (can be feasible or not), we will denote the total interference created to link l by other activated links k in the set $\Phi_l(K_l) \cap S$ as $I^{\mathsf{in}}_l(K_l, S)$,

i.e.,

$$I_l^{\text{in}}(K_l, S) \triangleq \sum_{k \in \Phi_l(K_l) \cap S} PG_{lk}. \qquad (6)$$

Also, the total interference created to link l by other activated links k in the set $\Phi_l(K_l)^c \cap S$ is denoted as $I_l^{\text{out}}(K_l, S)$, i.e.,

$$I_l^{\text{out}}(K_l, S) \triangleq \sum_{k \in \Phi_l(K_l)^c \cap S} PG_{lk} \qquad (7)$$

where $\Phi_l(K_l)^c$ denotes the complement of the set $\Phi_l(K_l)$.

Let $\vec{K}$ be a vector whose l-th element is K_l. In the following, we will consider a class of scheduling algorithms denoted by $\Psi(\vec{K}, \epsilon)$ which is parameterized by $\vec{K}$ and ϵ. A particular scheduling algorithm will belong to the scheduling class $\Psi(\vec{K}, \epsilon)$ if its constructed activation sets S satisfy the following *interference localization constraints*: for any link $l \in S$ we have $I_l^{\text{in}}(K_l, S) \leq (1-\epsilon)I_l^{\text{max}}$. For notational convenience, we will say an activation set $S \in \Psi(K_l, \epsilon)$ if S is constructed by any scheduling algorithm in the scheduling class $\Psi(\vec{K}, \epsilon)$ (i.e., S satisfies the property $I_l^{\text{in}}(K_l, S) \leq (1-\epsilon)I_l^{\text{max}}, \forall l \in S$). Clearly, an activation set S constructed by a scheduling algorithm in the scheduling class $\Psi(\vec{K}, \epsilon)$ is not guaranteed to be feasible in general. However, we will show that there exists $\vec{K}(\epsilon)$ whose elements are finite such that any activation set constructed by any scheduling algorithm in the class $\Psi(\vec{K}, \epsilon)$ is always feasible. This result is stated in the following theorem.

Theorem 2. Given any $0 < \epsilon < 1$, $\alpha > 2$, and any wireless network topology G, there exists a finite vector $\vec{K}(\epsilon)$ such that any activation set $S \in \Psi(\vec{K}, \epsilon)$ is feasible under the original SINR interference model given in Definition 1.

PROOF. The proof is given in Appendix C. □

In fact, in order to prove this theorem we show that if we maintain the total interference inside some predetermined neighborhood of every link $k \in E, k \neq l$ to be at most $(1-\epsilon)I_k^{\text{max}}$ then there exists a finite neighborhood around the receiver of link l which is determined by K_l such that $I_l^{\text{out}}(K_l, S) \leq \epsilon I_l^{\text{max}}$ for any activation set S. Therefore, the SINR of this particular link l is satisfied. The result of this theorem implies that it is possible for network links to coordinate their scheduling operations with other links in a local neighborhood, which are specified by $\vec{K}(\epsilon)$.

Let $\vec{K}^{\text{min}}(\epsilon)$ be the minimum vector $\vec{K}$ (in the componentwise sense) such that any activation sets S which satisfy the interference localization constraints are also feasible under the original SINR interference model given in Definition 1. Then, *an activation set S that satisfies the interference localization constraints for any given $\vec{K} \succeq \vec{K}^{\text{min}}(\epsilon)$ will be feasible under the original SINR interference model.* However, the reverse is not true in general. Now, let $\Lambda_t(\vec{K}, \epsilon)$ denote the maximum throughput region with interference localization, which is parameterized by ϵ and $\vec{K}$. We have the following results.

Theorem 3. Given $0 < \epsilon < 1$ and $\vec{K} \succeq \vec{K}^{\text{min}}(\epsilon)$, we have

1. *$\Lambda_t(\vec{K}, \epsilon) \subseteq \Lambda$ where Λ is the throughput region under the original SINR interference model.*

2. *$\Lambda_t(\vec{K}^{(2)}, \epsilon) \subseteq \Lambda_t(\vec{K}^{(1)}, \epsilon)$ for $\vec{K}^{\text{min}}(\epsilon) \preceq \vec{K}^{(1)} \preceq \vec{K}^{(2)}$.*

3. *$\Lambda_t(\vec{K}, \epsilon) \to \Lambda$ as $\epsilon \to 0$ if each feasible schedule $S \in \Omega$ have $\text{SINR}_k > \beta$ for all links $k \in S$. In general, we have $1/2\Lambda \subseteq \Lambda_t(\vec{K}, \epsilon)$ as $\epsilon \to 0$ for any network graph.*

4. *Given any network G and $0 < \epsilon < 1$, there exists a finite X such that $\frac{1}{X}\Lambda \subseteq \Lambda_t(\vec{K}, \epsilon)$.*

PROOF. Let $\Omega_t(\vec{K}, \epsilon)$ be the set of feasible schedules that satisfy the interference localization constraints parameterized by ϵ and $\vec{K}$. As discussed above, we have $\Omega_t(\vec{K}, \epsilon) \subseteq \Omega$ where recall that Ω is the set of all feasible schedules under the original SINR interference model. Since the throughput region is the convex hull of all feasible schedules, claim 1 is obviously correct.

Note that we have $\Omega_t(\vec{K}^{(2)}, \epsilon) \subseteq \Omega_t(\vec{K}^{(1)}, \epsilon)$. This is because any activation set $S \in \Omega_t(\vec{K}^{(2)}, \epsilon)$ also belongs to $\Omega_t(\vec{K}^{(1)}, \epsilon)$. Therefore, claim 2 holds. It can be observed that as $\epsilon \to 0$, we have $\Omega_t \to \Omega$ if each feasible schedule $S \in \Omega$ have $\text{SINR}_k > \beta$ for all links $k \in S$. This is because when ϵ is sufficiently small all feasible schedules $S \in \Omega$ has $I_l^{\text{in}}(K_G, S) \leq (1-\epsilon)I_l^{\text{max}} \ \forall l \in S$. Therefore, claim 3 holds. For brevity, the proof of $1/2\Lambda \subseteq \Lambda_t(\vec{K}, \epsilon)$ as $\epsilon \to 0$ and the proof of claim 4 are given in Appendices D and E, respectively. □

In general, the smaller the ϵ, the larger the interference neighborhoods and the larger the throughput region with interference localization $\Lambda_t(\vec{K}, \epsilon)$. Therefore, ϵ can be used to control the tradeoff between achievable throughput and potential overhead of scheduling operations (i.e., larger interference neighborhood would typically result in higher scheduling overhead).

5. LQF SCHEDULING UNDER INTERFERENCE LOCALIZATION

We investigate the performance of LQF scheduling under interference localization in this section. To proceed, let $B_l(K_l, L)$ denote the set of all links $k \in L$ such that $k \in \Phi_l(K_l)$ or $l \in \Phi_k(K_k)$ where $L \subseteq E$. Let $\omega_{l,L}$ be the maximum number of links in $B_l(K_l, L)$ that can be activated simultaneously by any maximal schedules under the interference localization constraints. Also, let $\omega_L^{\text{min}} = \min_{l \in L} \omega_{l,L}$. In addition, let $\omega_E^{\text{max}} = \max_{l \in E} \omega_{l,E}$ where note that $\omega_{l,E}$ is the maximum number of links that can be activated in the set $B_l(K_l, E)$. We have the following result.

Theorem 4. The efficiency ratio of LFQ scheduling with interference localization constraints is bounded away from zero for any network graph (i.e., there exists $\gamma_{\text{lb}} > 0$ such that $\gamma^* \geq \gamma_{\text{lb}}$).

PROOF. In order to prove Theorem 4, we need to show that the local pooling factors of all links l are bounded away from zero under interference localization constraints. Because the efficiency ratio of LQF, which is equal to the network local pooling factor, is equal to the minimum of all link local pooling factors, the theorem is proved.

Now, consider a particular link $l \in E$. Let L be the set of links containing link l such that the local pooling factors of link l and set L are equal to each other (i.e., $\sigma_l^* = \sigma_L^*$). Then, we have the following lower bound for a local pooling factor of set L [12]

$$\sigma_L^* \geq \max_{L' \subseteq L} \frac{\min_{\{\vec{S}_i \in \Omega_L\}} |\vec{S}_i|_{L'}}{\max_{\{\vec{S}_i \in \Omega_L\}} |\vec{S}_i|_{L'}} \qquad (8)$$

where $|\vec{S}_i|_{L'}$ is the number of activated links in schedule $\vec{S}_i$ that belongs to the subset of links $L' \subseteq L$. Now, let $l^* = \operatorname{argmin}_{l \in L} \omega_{l,L}$. Then, the maximum number of links in $B_{l^*}(K_{l^*}, L)$ that can be activated is $\omega_L^{\min}$. Note that at least one link in $B_{l^*}(K_{l^*}, L)$ must be activated in any maximal schedules under the interference localization constraints. By applying the result in (8) for this particular choice of $L' = B_{l^*}(K_{l^*}, L)$, we have $\sigma_L^* \geq 1/\omega_L^{\min}$. Moreover, note that $\omega_E^{\max} \geq \omega_L^{\min}$ because $L \subseteq E$. Therefore, we have $\sigma_l^* = \sigma_L^* \geq 1/\omega_L^{\min} \geq 1/\omega_E^{\max}$.

Now, we prove the theorem by showing that $\omega_E^{\max}$ is finite. Recall that the interference localization constraints are defined by two set of parameters, namely, ϵ and $\vec{K} \succeq \vec{K}^{\min}(\epsilon)$. In addition, Theorem 2 shows that we can always find the vector $\vec{K}$ which is finite element-wise for any network graph (including networks with infinite number of links). Moreover, we can activate only a finite number of links in any set $B_l(K_l, E)$ for any link l given a finite K_l. This is because the set of links in $B_l(K_l, E)$ for any l lie in a finite area around link l (even for networks with infinite area). Therefore, $\omega_E^{\max}$ is finite (i.e., cannot be made arbitrarily large). Therefore, the theorem has been proved. $\square$

As discussed above, the small efficiency ratio of LQF scheduling in some network topologies results from the scheduling starvation problem. Using interference localization, we essentially prevent this scheduling starvation from happening. Specifically, if the SINR of a particular link in a schedule is equal to β then we cannot activate any other links regardless of their relative distances to the receiver of this link. Interference localization forces all activated links to operate above the SINR limit β (or at a slightly lower interference limit), which enables the activation of a significant number of links in a large network. In other words, interference localization prevents LQF scheduling from activating a globally "small" maximal schedule, which in turn guarantees a non-zero performance lower bound.

The result for the efficiency ratio of LQF scheduling stated in Theorem 4 corresponds to the throughput region with interference localization $\Lambda_t(\vec{K}, \epsilon)$. Let $\gamma_{\mathsf{LQF}}(G)$ be the efficiency ratios of LQF scheduling with respect to the original throughput region Λ. Then, we have following results.

Lemma 3. Given $0 < \epsilon < 1$ and $\vec{K} \succeq \vec{K}^{\min}(\epsilon)$, then

1. *LQF scheduling under interference localization achieves a fraction of the original throughput region which is bounded away from zero for any $0 < \epsilon < 1$.*

2. *LQF scheduling under interference localization achieves at least $\frac{\gamma_{\mathsf{LQF}}(G)}{2}$ fraction of the original throughput region Λ as $\epsilon \to 0$.*

PROOF. According to Theorem 4, there exists some $\gamma_{\mathsf{lb}} > 0$ such that LQF scheduling under interference localization can stabilize the network for any arrival rate vector $\vec{\lambda} \in \gamma_{\mathsf{lb}}\Lambda_t(\vec{K}, \epsilon)$. Also, due to claim 4 of Theorem 3, we have $\frac{1}{X}\Lambda \subseteq \Lambda_t(\vec{K}, \epsilon)$. Therefore, LQF scheduling under interference localization can stabilize the network for any arrival rate vector $\vec{\lambda} \in \frac{\gamma_{\mathsf{lb}}}{X}\Lambda$. Because X is finite according to claim 4 of Theorem 3, $\frac{\gamma_{\mathsf{lb}}}{X}$ is bounded away from zero. Therefore, claim 1 has been proved. In addition, claim 2 follows immediately from the result in claim 3 of Theorem 3 (i.e., $1/2\Lambda \subseteq \Lambda_t(\vec{K}, \epsilon)$ as $\epsilon \to 0$). $\square$

Note that in the case $\Lambda_t(\vec{K}, \epsilon) = \Lambda$ as $\epsilon \to 0$, the LQF scheduling under interference localization achieves exactly $\gamma_{\mathsf{LQF}}(G)$ fraction of the original throughput region Λ.

6. PRACTICAL SCHEDULING DESIGNS UNDER SINR INTERFERENCE MODEL

We have shown how to localize interference using the two parameters ϵ and $\vec{K} \succeq \vec{K}^{\min}(\epsilon)$. In this section, we present a simple technique to determine $\vec{K}$ for a given ϵ. In addition, we propose a distributed LQF scheduling algorithm.

6.1 Determination of Interference Neighborhood

Given ϵ, we show how to find $K_l(\epsilon)$ for a particular link l in the following. This procedure needs to be applied to each link in the network to obtain $\vec{K}$. Note that this is a centralized procedure but it needs to be performed only once for a static wireless network. In fact, to determine interference neighborhood for link l, we only need to consider concentric circles around the receiver of link l whose radii are the distances from the receiver of link l to the transmitters of other links $k \neq l$. Let the set of these radii sorted in the increasing order be $\vec{\pi}_l$ ($\vec{\pi}_l$ has $|E|-1$ elements). Specifically, we should choose the interference neighborhood for each link l large enough such that for any scheduling policy belonging to the scheduling class $\Psi(\vec{K}, \epsilon)$, the total interference due to all activated links outside the interference neighborhood is not larger than $\epsilon I_l^{\max}$.

In the following, we describe a procedure to calculate an upper bound of the total interference from outside the interference neighborhood of link l for a given size of interference neighborhood of link l determined by K_l ($K_l = \vec{\pi}_l(h)/d_l$ for some h). Let denote this interference upper bound corresponding to a particular value of K_l be $I_l^{\mathsf{ub}}(K_l)$.

- Divide the network into a number of small areas (e.g., small square areas). We only find the radius of the interference neighborhood for any particular link which is large enough such that this interference neighborhood contains all other links belonging to the same small area with the underlying link.

- Determine all possible maximal sets of links that belong a particular small area and lie outside the interference neighborhood determined by K_l that can be activated simultaneously while not exceeding the interference limit $(1 - \epsilon)I_k^{\max}$ for any link k in that activated set. These activation sets are determined assuming that all other links in other small areas are silent. Then, the interference contribution from this small area to the receiver of link l is counted from the maximal activation set that creates the largest total interference to link l.

- Sum the maximum interference contributed by all small areas to $I_l^{\mathsf{ub}}(K_l)$ to obtain the interference upper bound.

Search the radii from the list $\vec{\pi}_l$, which is equal to $K_l d_l$, in the increasing order until the interference upper bound $I_l^{\mathsf{ub}}(K_l)$ satisfies $I_l^{\mathsf{ub}}(K_l) \leq \epsilon I_l^{\max}$ where the interference upper bound $I_l^{\mathsf{ub}}(K_l)$ is calculated as described above. Let the radius in the list $\vec{\pi}_l$ at the stopping iteration be $\vec{\pi}_l(h)$, then the corresponding value of K_l is $K_l = \vec{\pi}_l(h)/d_l$. In fact,

there will be an optimal size for the small areas, which results in the smallest K_l for any particular link l. In particular, if the size of each area is too small then the interference upper bound $I_l^{ub}(K_l)$ is too loose. In contrast, if the size of each area is too large then K_l is larger than necessary because we require that all other links belonging to the same area with link l lie completely inside its interference neighborhood.

6.2 Distributed LQF Scheduling Algorithm

Assume ϵ and $\vec{K}$ are given, we are interested in designing an efficient distributed LQF algorithm. Suppose that each time slot is divided into a scheduling period and a transmission period. A schedule is constructed in the scheduling period which is used to transmit data in the transmission period. Assume that each link l broadcasts its queue length information to other links in $B_l(K_l, E)$ at the beginning of each time slot. In addition, assume that each link l can estimate the interference power that each link $k \in \Phi_l(K_l)$ creates for itself in advance. Let $\Delta_l(K_l)$ be the set of all links k such that $l \in \Phi_k(K_k)$. That means the transmitter of link l is on the boundary or inside the circle of radius $K_k d_k$ around the receiver of such links k.

Algorithm 1 LQF Scheduling at Link l

1: Link l broadcasts SCH-REQ to all links in $\Delta_l(K_l)$
2: Links $k \in S \cap \Delta_l(K_l)$ temporarily calculate cumulative interference I_k^{tem} assuming that link l is added to S.
3: **if** any link $k \in S \cap \Delta_l(K_l)$ has $I_k^{tem} > (1-\epsilon)I_k^{max}$ **then**
4:   - Link k sends an NACK message to link l.
5: **end if**
6: **if** link l receives no NACK messages from other links in $\Delta_l(K_l)$ **then**
7:   - Link l is added to the schedule (i.e., $S = S + l$)
8:   - Link l sends SCH-SUCCESS message to other links in $B_l(K_l, E)$ who will remove l from their local active set of links.
9:   - Link l and other links $k \in \Delta_l(K_l)$ calculate their new cumulative interference.
10:   **if** any link $k \in \Delta_l(K_l)$ has its new cumulative interference exceeding its interference limit (i.e., $I_k > (1-\epsilon)I_k^{max}$) **then**
11:     - Link k sends REMOVE-REQ message to its neighboring links in $B_k(K_k, E)$
12:     - Links receiving REMOVE-REQ from link k remove k from their sets of local active links.
13:   **end if**
14: **else if** link l receives at least one NACK message from other links in $\Delta_l(K_l)$ **then**
15:   - Link l is not added to the schedule.
16:   - Link l sends REMOVE-REQ message to its neighboring links in $B_l(K_l, E)$ who will remove l from their sets of local active links.
17: **end if**

LQF scheduling can be implemented in a centralized manner where links are added to the schedule in the decreasing order of their queue lengths. In particular, a link is added to the schedule only if its SINR is satisfied and it can maintain the SINR requirements of other links already in the schedule. For brevity, we will refer to this centralized LQF algorithm as LQF algorithm.

We now propose a distributed LQF (DLQF) scheduling algorithm. Let S be the schedule (i.e., the set of activated

links) under construction by the algorithm. At the high level, links with locally largest queue length (in their sets of local active links) can simultaneously attempt to add themselves to the schedule in DLQF. Therefore, the DLQF algorithm is more greedy than LQF algorithm because LQF algorithm adds one link at a time in the decreasing order of queue lengths. In order to obtain a maximal feasible schedule (under interference localization constraints), each active link needs to update its local active links in the DLQF algorithm.

The DLQF algorithm works as follows. Initially, each link l initializes its local active set of links as $E_l = B_l(K_l, E)$. Then, any particular link l that has longest queue length among other links in E_l (i.e., link l has the heaviest weight among its current local active neighbors), will run algorithm 1 to add itself to the schedule. This process is continued until $E_l = \emptyset$, $\forall l \in E$. Here, links need to update their sets of local active links throughout the course of the algorithm (lines 12-16).

It can be shown that LQF and DLQF return the same maximal schedule if all link weights are distinct. If link weights are not distinct, LQF and DLQF still produce the same maximal schedule given some specific deterministic tie-breaking rule (e.g., links with lower indices have higher priority). For brevity, these proofs are omitted. In addition, assume that locally heaviest links can add themselves to the schedule simultaneously in one mini-slot of the scheduling period. Then, the maximum number of mini-slots needed in the scheduling period can be bounded by the largest number of links in any set $B_l(K_l, E), l \in E$. Therefore, the overhead of DLQF is quite small in practice.

7. NUMERICAL RESULTS

We present numerical results to illustrate the performance of LQF scheduling under the SINR interference model and the effect of interference localization. First, consider a simple network with 12 links whose link lengths are all equal to $d = 80$m as shown in Fig. 2. We place the links for this network such that if we activate two links $2k+1$ and $2k+2$ for $0 \le k \le 5$ then the SINR of each link is equal to $\beta = 5$ (i.e., this means $d_0 = d\beta^{1/\alpha}$ where d_0 is shown in this figure). In addition, links are placed such that we have two "large" maximal schedules: S_1 activates 6 odd links (1, 3, 5, 7, 9, 11) and S_2 activates 6 even links (2, 4, 6, 8, 10, 12). In fact, this network is a special case of the one shown in Fig. 1 with $K = 6$. Therefore, the efficiency ratio of LQF scheduling in this network G can be upper bounded as $\gamma(G) \le 2/6 = 1/3$.
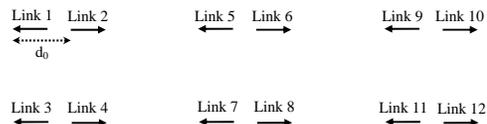

Figure 2: Network of 12 links.

Assume arrivals to all links in Fig. 2 follow independent Bernoulli processes with the same average arrival rate. We plot total average queue length versus the average arrival rate per link for three scheduling schemes: LQF, randomized maximal scheduling (MS), and PICK&COMPARE (P&C) scheduling. Average queue lengths are obtained by running
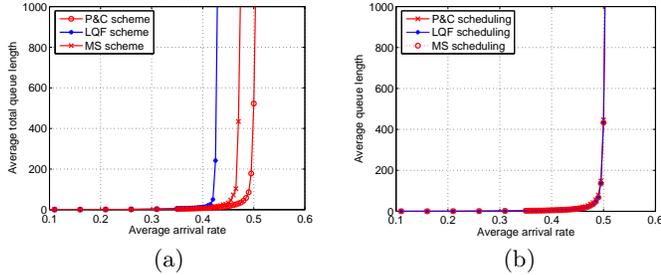
(a)   (b)

Figure 3: Average total queue length versus average arrival rate under different scheduling policies (a) For network in Fig. 2 (b) For modified network of Fig. 2.

(a)   (b)

Figure 4: (a) Sorted elements of $\vec{K}$ for the random network of 40 links (b) Average total queue length versus average arrival rate with and without interference localization.

the corresponding scheduling scheme over 10^4 time slots. For the MS scheme, we randomly choose one maximal schedule in each time slot. For the P&C scheme, which is known to achieve 100% throughput [14], we randomly generate a new maximal schedule in each time slot and choose the one with larger weight (total queue length) between this newly-generated schedule and the schedule used in the previous time slot for data transmission.

Fig. 3(a) shows that LQF scheduling achieves even smaller throughput than the MS scheme in this setting. However, by comparing the throughput achieved by LQF and P&C schemes we can observe that LQF achieves much larger throughput than the analytical bound of $1/3$. This is expected because the performance bound of $1/3$ corresponds to the worst-case performance under some bad arrival pattern while we use independent Bernoulli arrivals for different links.

Now, we modify the network topology in Fig. 2 slightly by changing the distance d_0 in this figure to $d_0 - \epsilon$ for some small $\epsilon > 0$. With this small change, any two links $2k + 1$ and $2k + 2$ for $0 \leq k \leq 5$ do not form a feasible schedule anymore. Instead all maximal schedules in this modified network has 6 links. We show the performance of the three scheduling schemes again under this modified network topology in Fig. 3(b). The results show that the three scheduling schemes achieve the same throughput performance. This can be interpreted as follows. Since all maximal schedules include the same number of active links, they both achieve 100% throughput. In fact, this modified network topology satisfies the condition of Lemma 2. Therefore, LQF scheduling has an efficiency ratio equal one in this case.

We validate the interference localization technique for a random network of 40 links whose lengths are all equal to $d = 80$m in an area 2000mx2000m. We plot sorted elements of vector $\vec{K}$, which determines the interference neighborhood of all network links, in Fig. 4(a) for $\alpha = 4$, $\beta = 5$, $\epsilon = 0.05$. The vector $\vec{K}$ is obtained by using the technique described in section VI.A and the network area is divided into 49 equal-size square areas. This figure shows that the radius of interference neighborhoods for all links are smaller than 5 times the link length for these chosen parameters. This means that each link only needs to coordinate its scheduling operations with other links in a relatively small neighborhood.

We investigate the performance of different scheduling schemes under SINR interference model with and without
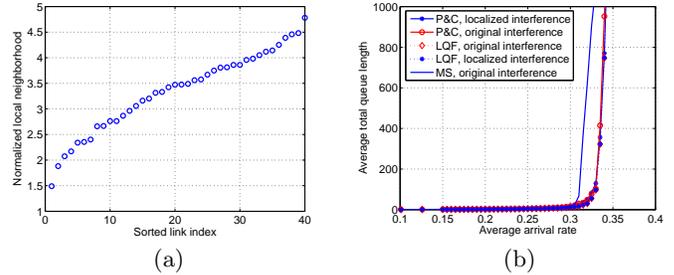
interference localization for this random network. We plot average total queue length versus the average arrival rate for different scheduling schemes in Fig. 4(b). The interference neighborhoods that determine interference localization constraints correspond to $\vec{K}$ shown in Fig. 4(a). The results show that the P&C scheme achieves the same throughput with and without interference localization, which implies that interference localization does not reduce the throughput region for this network with $\epsilon = 0.05$. In addition, LQF scheduling achieves 100% throughput for both the cases with and without interference localization. In contrast, the randomized MS scheme does not achieve maximum throughput as expected. The results in this section confirm the benefit of interference localization, with which we can decentralize the LQF scheduling while not compromising its throughput performance in a practical wireless network.

8. CONCLUSIONS

We investigated the performance and designed practical distributed algorithms for LQF scheduling under the SINR interference model. Specifically, we showed that LQF scheduling achieves zero throughput in the worst case, and provided a sufficient condition for LQF scheduling to achieve 100% throughput. Moreover, we proposed a novel interference localization technique for which each link only needs to coordinate its scheduling operations within its local neighborhood while still maintaining the scheduling feasibility. We showed that LQF scheduling achieves strictly positive throughput under interference localization constraints. Finally, we proposed a distributed LQF scheduling algorithm that returns the same maximal schedule as the centralized counterpart under interference localization constraints.

9. ACKNOWLEDGMENTS

10. REFERENCES

[1] S. A. Borbash and A. Ephremides. Wireless Link Scheduling With Power Control and SINR Constraints. *IEEE Trans. Inf. Theory*, 52(11):5106–5111, November 2006.

[2] A. Brzezinski, G. Zussman, and E. Modiano. Distributed Throughput Maximization in Wireless Mesh Networks via Pre-Partitioning. *IEEE/ACM Trans. Netw.*, 16(6):1406–1419, December 2008.

[3] P. Chaporkar, K. Kar, and S. Sarkar. Throughput Guarantees Through Maximal Scheduling in Wireless Networks. *IEEE Trans. Inf. Theory*, 54(2):572–594, February 2008.

[4] R. L. Cruz and A. V. Santhanam. Optimal Routing, Link Scheduling, and Power Control in Multi-hop Wireless Networks. In *IEEE INFOCOM*, April 2003.

[5] A. Dimakis and J. Walrand. Sufficient Conditions for Stability of Longest-Queue-First Scheduling: Second-order Properties using Fluid Limits. *Advances in Applied Probability*, 38(2):505–521, 2006.

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

[7] C. Joo, X. Lin, and N. B. Shroff. Understanding the Capacity Region of the Greedy Maximal Scheduling Algorithm in Multi-hop Wireless Networks. *IEEE/ACM Trans. Netw.*, 17(4):1132–1145, August 2009.

[8] C. Joo and N. B. Shroff. Performance of Random Access Scheduling Schemes in Multi-hop Wireless Networks. *IEEE/ACM Trans. Netw.*, 17(5), October 2009.

[9] L. B. Le, E. Modiano, and N. B. Shroff. Optimal Control of Wireless Networks with Finite Buffers. In *IEEE INFOCOM*, March 2010.

[10] M. Leconte, J. Ni, and R. Srikant. Improved Bounds on the Throughput Efficiency of Greedy Maximal Scheduling in Wireless Networks. In *ACM MOBIHOC*, pages 165–174, New York, NY, USA, 2009. ACM.

[11] H.-W. Lee, E. Modiano, and L. B. Le. Distributed Throughput Maximization in Wireless Networks via Random Power Allocation. In *WiOpt*, 2009.

[12] B. Li, C. Boyaci, and Y. Xia. A Refined Performance Characterization of Longest-queue-first Policy in Wireless Networks. In *ACM MOBIHOC*, pages 65–74, New York, NY, USA, 2009. ACM.

[13] X. Lin and N. B. Shroff. The Impact of Imperfect Scheduling on Cross-Layer Congestion Control in Wireless Networks. *IEEE/ACM Trans. Netw.*, 14(2):302–315, April 2006.

[14] E. Modiano, D. Shah, and G. Zussman. Maximizing Throughput in Wireless Networks via Gossiping. *Sigmetrics Performance Evaluation Review*, 34(1):27–38, 2006.

[15] M. J. Neely, E. Modiano, and L. Chih-Ping. Fairness and Optimal Stochastic Control for Heterogeneous Networks. *IEEE/ACM Trans. Netw.*, 16(2), April 2008.

[16] G. Sharma, N. B. Shroff, and R. R. Mazumdar. On the Complexity of Scheduling in Wireless Networks. In *ACM MOBICOM*, September 2006.

[17] L. Tassiulas and A. Ephremides. Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximal Throughput in Multihop Radio Networks. *IEEE Trans. Autom. Control*, 37(12):1936–1948, December 1992.

[18] P. J. Wan. Multiflows in Multihop Wirelress Networks. In *ACM MOBIHOC*, 2009.

[19] G. Zussman, A. Brzezinski, and E. Modiano. Multihop Local Pooling for Distributed Throughput Maximization in Wireless Networks. In *IEEE INFOCOM*, April 2008.

APPENDIX

A. REVIEW OF σ-LOCAL POOLING

We review some important definitions related to σ-local pooling technique [7,12] in this appendix. In general, the σ-local pooling notion is developed based on properties of the set of feasible maximal schedules for a given network and an interference model. The interference relationship under the SINR interference model is much more complicated than that under binary interference models such as the k-hop interference model. Therefore, performance analysis for LQF scheduling under the SINR interference model is more difficult. First, we provide definitions of set, link, and network σ-local pooling.

Definition 6. Given a non-empty set of links $L \subseteq E$, we say L has a set local pooling factor σ_L^* if

$$
\begin{aligned}
\sigma_L^* &\triangleq \sup \{\sigma | \sigma \vec{\mu}_L \not\succeq \vec{\nu}_L \text{ for all } \vec{\mu}_L, \vec{\nu}_L \in Co\,(M_L)\} \\
&\triangleq \inf \{\sigma | \sigma \vec{\mu}_L \succeq \vec{\nu}_L \text{ for some } \vec{\mu}_L, \vec{\nu}_L \in Co\,(M_L)\}
\end{aligned}
$$

where M_L denotes the set of maximal schedules limited to the set of links L. We will also use M_L to denote a matrix whose columns represent maximal schedules limited to the set of links L where an element of a particular column is equal 1 if the corresponding link belongs to that maximal schedule and it is equal to 0 otherwise.

Definition 7. The local pooling factor for link l, denoted as σ_l^*, is the minimum of all σ_L^* for all sets $L \subseteq E$ that contain link l, i.e.,

$$
\sigma_l^* \triangleq \min_{\{L \subseteq E | l \in L\}} \sigma_L^*. \tag{9}
$$

And the network local pooling factor $\sigma^*(G)$ for network G is minimum of all link local pooling factors, i.e.,

$$
\sigma^*(G) \triangleq \min_{l \in E} \sigma_l^*. \tag{10}
$$

In [7], it has been shown that the efficiency ratio of LQF scheduling for a particular network G is exactly equal to its network local pooling factor (i.e., $\gamma(G) = \sigma^*(G)$). A set local pooling factor can be calculated from either primal or dual formulation of a specific optimization problem [12]. In particular, given a non-empty set of links $L \subseteq E$, σ_L^* is the optimal solution of the following optimization problem

$$
\begin{aligned}
\min_{\{\sigma, \vec{\mu}_L, \vec{\nu}_L\}} \quad & \sigma \\
\text{subject to} \quad & \sigma \vec{\mu}_L \succeq \vec{\nu}_L \\
& \vec{\mu}_L, \vec{\nu}_L \in Co\,(M_L).
\end{aligned} \tag{11}
$$

Equivalently, σ_L^* can be found from the dual problem of the above optimization problem, i.e., it is the optimal solution w^* of the following optimization problem

$$
\begin{aligned}
\max_{\{w, \vec{x} \succeq 0\}} \quad & w \\
\text{subject to} \quad & w \vec{e}^t \preceq \vec{x}^t M_L \preceq \vec{e}^t
\end{aligned} \tag{12}
$$

where $\vec{e}$ is a column vector of all ones of an appropriate dimension, $(.)^t$ denotes the vector or matrix transpose. Note

that if $\sigma_L^* = 1$, then there must exist a vector $\vec{x} \succeq 0$ such that $\vec{x}^t M_L = \vec{e}^t$. This is exactly the local pooling condition originally defined in [5]. It can be noticed from (11) that $\sigma_L^* = 1$ implies no maximal schedule dominates other maximal schedules.

In [12], it was shown that the local pooling factor of a particular link l is limited by a particular set of links called *limiting set* instead of the whole network. In particular, let L be the limiting set of link l then we have $\sigma_l^* = \sigma_L^*$. In general, let $L \subseteq E$ be a non-empty set of links such that $l \in L$, then we have $\sigma_L^* \geq \sigma_l^*$.

B. PROOF OF LEMMA 1

Let us define the following vectors

$$\vec{\nu}_1 \triangleq \frac{1}{K_1}\vec{S}_{\text{tot}}^{\text{g}} = \frac{1}{K_1}\sum_{i\in\Omega_g}\vec{S}_i \tag{13}$$

$$\vec{\nu}_2 \triangleq \frac{1}{K_2}\vec{S}_{\text{tot}}^{\text{b}} = \frac{1}{K_2}\sum_{i\in\Omega_b}\vec{S}_i. \tag{14}$$

Then, $\vec{\nu}_1, \vec{\nu}_2 \in Co\,(\Omega)$ where Ω is the set of all possible maximal schedules. Recall that we have $k^*\vec{S}_{\text{tot}}^{\text{g}} \succeq \vec{S}_{\text{tot}}^{\text{b}}$. Hence, we have

$$k^*K_1\vec{\nu}_1 \succeq K_2\vec{\nu}_2 \quad\text{or}\quad \frac{k^*K_1}{K_2}\vec{\nu}_1 \succeq \vec{\nu}_2. \tag{15}$$

By using (11), we have $\sigma_E^* \leq k^*K_1/K_2 \triangleq \sigma_{\text{ub}}^*(G)$. Therefore, using the definitions in (9) and (10), we have $\sigma^*(G) \leq \sigma_E^* \leq k^*K_1/K_2 = \sigma_{\text{ub}}^*(G)$, which proves the lemma.

C. PROOF OF THEOREM 2

Note that for a particular link l and an activation set $S \in \Psi(K_l, \epsilon)$, if we have $I_l^{\text{out}}(K_l, S) \leq \epsilon I_l^{\text{max}}$, then we have $\mathsf{SINR}_l \geq \beta$. This is because the total interference experienced at the receiver of link l is smaller or equal to $\epsilon I_l^{\text{max}} + (1-\epsilon)I_l^{\text{max}} = I_l^{\text{max}}$ in this case where I_l^{max} is the maximum tolerable interference for link l given in (5). Hence, to prove the theorem we will show that for any particular link $l \in S$ where $S \in \Psi(K_l, \epsilon)$, there exists a finite K_l^* such that for any $K_l \geq K_l^*$, we have $I_l^{\text{out}}(K_l, S) \leq \epsilon I_l^{\text{max}}$. Then, the claim of the theorem holds for any $\vec{K} \succeq \vec{K}^*$ where the l-th elements of $\vec{K}^*$ are K_l^*. Therefore, the theorem is proved. Now, we show the existence of K_l^* by taking the following steps:

1. We construct a particular link activation set S whose links in the set $S \cap \Phi_l(K_l)^c$ for a given K_l create the total interference to link l as close to the worst-case scenario as possible. Find K_l such that $I_l^{\text{out}}(K_l, S) \leq \epsilon I_l^{\text{max}}$ under this activation set.

2. Prove that the total interference created to link l from links in the set $S \cap \Phi_l(K_l)^c$ due to the chosen activation set in step 1) is always larger than that due to any worst-case activation sets $S \in \Psi(K_l, \epsilon)$. Therefore, the obtained K_l in step 1 is no smaller than the minimum value of K_l calculated from any worst-case activation set.

Due to space limitation, the detailed proof is omitted.

D. PROOF OF CLAIM 3 IN THEOREM 3

We prove that $1/2\Lambda \subseteq \Lambda_t(\vec{K}, \epsilon)$ as $\epsilon \to 0$ for any network in this appendix. We consider the following two cases. For the first case, each feasible schedule S_i under the original SINR interference model (without interference localization constraints) satisfies $\mathsf{SINR}_k < \beta$ for all links $k \in S_i$. In this case, $\Lambda_t(\vec{K}, \epsilon) \to \Lambda$ as $\epsilon \to 0$ following the proof of Theorem 3.

For the second case, we assume that there exist feasible maximal schedules S_i under the original SINR interference model (without interference localization constraints) such that $\mathsf{SINR}_k = \beta$ for some link $k \in S_i$. In this case, we can easily show that any such schedule S_i can be decomposed into two non-overlapping schedules $S_{i,1}$ and $S_{i,2}$ (i.e., $S_{i,1} \cup S_{i,2} = S_i$ and $S_{i,1} \cap S_{i,2} = \emptyset$) that satisfy the interference localization constraints with a sufficiently small ϵ.

We are ready to prove the claim. For any vector $\vec{\eta} \in \Lambda$ we can represent it as $\vec{\eta} = \sum_i \alpha_i \vec{S}_i$ where $\sum_i \alpha_i \leq 1$ and $\vec{S}_i$ is a feasible schedule. As we have shown above, we can decompose that feasible schedule $\vec{S}_i$ into two schedules $\vec{S}_{i,1}$ and $\vec{S}_{i,2}$ that satisfy the interference localization constraints. Therefore, we have $\frac{\vec{\eta}}{2} = 1/2\sum_i \alpha_i \vec{S}_i = 1/2\sum_i\sum_{j=1}^2 \alpha_i \vec{S}_{i,j} \in \Lambda_t(\vec{K}, \epsilon)$. This is because $1/2\sum_i\sum_{j=1}^2 \alpha_i \vec{S}_{i,j}$ is a linear combination of different schedules $\vec{S}_{i,j}$ that satisfy interference localization constraints and $1/2\sum_i\sum_{j=1}^2 \alpha_i = \sum_i \alpha_i \leq 1$. Therefore, we have proved the claim.

E. PROOF OF CLAIM 4 IN THEOREM 3

Similar to the proof in Appendix D, if for each maximal schedule $S_i \in \Omega$ (the set of feasible maximal schedules under the original SINR interference model) we can find a set of at most X distinct schedules $S_{i,1}^t, S_{i,2}^t, \ldots, S_{i,X}^t \in \Omega_t(\vec{K}, \epsilon)$ (the set of schedules satisfying interference constraints) such that $S_i = \sum_{j=1}^{\gamma} S_{i,j}^t$ then we have $\frac{1}{X}\Lambda \subseteq \Lambda_t(\vec{K}, \epsilon)$.

Now consider a particular maximal schedule $S_i \in \Omega$ and let $A_l(S_i) \triangleq S_i \cap B_l(K_l, E)$ where recall that $B_l(K_l, E)$ denotes the set of all links $k \in E$ such that $k \in \Phi_l(K_l)$ or $l \in \Phi_k(K_k)$. We choose X as follows:

$$X \triangleq 1 + \max_{l\in E}|A_l(S_i)| \tag{16}$$

where $|A_l(S_i)|$ denotes the number of links in set $A_l(S_i)$. Note that X is finite because the number of activated links within each set of links $B_l(K_l, E)$ is finite. Then, it can be shown that each schedule $S_i \in \Omega$ can be decomposed into X distinct schedules that satisfy the interference localization constraints. In particular, we can use X different colors to paint links in schedule S_i such that links painted by the same color form a schedule that satisfies the interference localization constraints as follows. For each link l in S_i we paint it using a color that is not used by any other links in the set $S_i \cap B_l(K_l, E)$. Continue doing this if it is still possible. It can be easily shown that all links in S_i will be painted by this procedure if X is chosen according to (16). In addition, links painted by the same color form a schedule that satisfies the interference localization constraints. Therefore, we have completed the proof.