PERFORMANCE OF GREEDY SCHEDULING ALGORITHM IN WIRELESS NETWORKS

By

BO LI

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2012

To my parents Wenshen Guan and Zixiang Li.

ACKNOWLEDGMENTS

I would like to thank my advisor, Professor Ye Xia, for all his advice, support, patience and encouragement. I am deeply impressed by his rigorous attitude and his enthusiastic to the research, which have been making great impact on me. With his guidance, I have learned the way of creative thinking, rigorous reasoning and organized expressing. Without his help, I could not imagine how difficult it would be for me to complete this dissertation. I would like to express my appreciation to my committee, my co-advisor Professor My T. Thai, Professor Shigang Chen and Alin Dobra who taught me a lot both in class and after class. I would like to thank my committee members Professor Arunava Banerjee and Yuguang Fang for all their valuable comments and suggestions. I am also thankful to the office-mates and friends that I met at the University of Florida, for supporting me over the years and for making my life at UF enjoyable. I should thank my colleague and friend, Cem Boyaci, who has been cooperating with me during almost my whole Ph.D years. In the end, please allow me to thank my dearest parents who have always loved me and encouraged me in my life. Especially when I faced challenges and pressure, their unconditional support and understanding help me through those tough time. Their strong mind, optimistic attitude and selfless love make me proud to have such a great family.

TABLE OF CONTENTS

LIST OF FIGURES

Figure page

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

PERFORMANCE OF GREEDY SCHEDULING ALGORITHM IN WIRELESS NETWORKS

By

Bo Li

Aug 2012

Chair: Ye Xia
Major: Computer Engineering

One of the major challenges in wireless networking is how to optimize the link scheduling decisions under interference constraints. Recently, a few algorithms have been introduced to address the problem. However, solving the problem to optimality for general wireless interference models typically relies on a solution of an NP-hard sub-problem. To meet the challenge, one stream of research currently focuses on finding simpler sub-optimal scheduling algorithms and on characterizing the algorithm performance.

In the first piece of our work, we investigate the performance of a specific scheduling policy called Longest Queue First (LQF), which has gained significant recognition lately due to its simplicity and high efficiency in empirical studies. There has been a sequence of studies characterizing the guaranteed performance of the LQF schedule, culminating at the construction of the σ-local pooling concept by Joo et al. We refine the notion of σ-local pooling and use the refinement to capture a larger region of guaranteed performance.

In the second piece of our work, we deeply analyze the performance guarantee of the LQF algorithm in order to shape the stability region furthermore. The contribution of this study is to describe three new achievable rate regions, which are larger than the previously known regions. In particular, the new regions include all the extreme points of the capacity region and are not convex in general. We also discover a counter-intuitive

phenomenon in which increasing the arrival rate may sometime help to stabilize the network. This phenomenon can be well explained using the theory developed in this study.

In the third piece of our work, we study the performance of LQF algorithm based on a more practical wireless network model, where channel fading effect is considered. Unlike the previous discussed network model, the wireless channel state is time varying under the effect of channel fading. As a result, the corresponding interference relationship among links could change under different channel states. Moreover, a subset of links, which determined by the current channel state, may be prohibited to transmit data and hence be pre-exclude from the scheduling process. We adopt a more generic channel fading model than that studied by Reddy et al, so that the variation of underlying interference relationship is allowed. We derive a larger stability region $\Sigma^*(G)\Lambda$, compared to the existed result, where $\Sigma^*(G)$ is a diagonal matrix. We also propose an estimation algorithm of $\Sigma^*(G)$, which provides a performance lower bound of LQF under any given channel fading structure.

CHAPTER 1
INTRODUCTION

Recent years have seen a great development in wireless networking technologies and their usage. One of the major challenges in wireless networking is how to utilize the communication medium efficiently under interference constraints. For different wireless technologies, different interference models have been established. The 1-hop interference model (also known as the node-exclusive or primary interference model), where two links interfere only if they share a common node, is suitable to characterize the interference in FH-CDMA and Bluetooth networks [25]. The hop count is measured in the interference graph in which a node represents a physical link in the network and an edge between two nodes means the two corresponding physical links interfere with each other. With the same interference-graph-based terminology, the 2-hop interference model can successfully capture the interference relationship in the IEEE 802.11 network [2]. In [36] and [35], the authors considered the more general k-hop interference model.

The research on joint link scheduling and routing strives to find the most efficient way to forward traffic from sources to destinations. In [41], the authors provided an algorithm that achieves the full capacity region of the wireless network. However, their algorithm requires solving a global maximization subproblem at each iteration. For the 1-hop interference model, this subproblem reduces to finding the maximum weighted matching in the backlog weighted network graph. While maximum matching can be solved in $O(|V|^3)$ time with a centralized algorithm [18], where V is the set of nodes in the network, the running time is considered inefficient for a network algorithm and a faster approach is desired. The same algorithm given in [41] can also be applied to more generic interference models, but the subproblem becomes intractable in those cases. For instance, under general interference models characterized by some interference graph, the subproblem is to find the maximum weighted independent set of the interference graph, which is NP-hard and makes the algorithm in [41] inapplicable. In [35], the authors showed that for the k-hop interference model where $k > 1$, the subproblem is also NP-Hard.

1.1 Longest-Queue-First Policy in Wireless Networks

Considering the fundamental difficulty in wireless scheduling, one of the main efforts by the research community is to find simpler sub-optimal scheduling algorithms that are also friendly to distributed implementation and to characterize their performance guarantee. Among the proposed solutions [9, 10, 15, 19, 24, 26, 31], the *Longest Queue First* (LQF, also known as the greedy maximal schedule) algorithm has distinguished itself due to its simplicity and high performance in empirical studies [38]. The LQF schedule chooses links in a decreasing order of the queue sizes while conforming to the interference constraints. In an effort to understand the surprising efficiency of this simple algorithm, Dimakis and Walrand have identified a sufficient condition for the algorithm to achieve the entire capacity region of the network for single-hop traffic [12]. In particular, they have shown that if the network topology and interference structure satisfy a condition known as *local pooling*, then the LQF algorithm achieves the entire capacity region. Brzezinski et al. have extended the definition of local pooling to the multi-hop traffic situation [7]. The same authors have also investigated classes of networks that satisfy single-hop local pooling [8].

In a different direction of generalizing local pooling, Joo et al. have investigated a fractional version called σ-local pooling [18]. Specifically, suppose the network is denoted by G, and the capacity region is denoted by Λ, i.e., the largest rate region that can possibly be supported by the network using some scheduling policy. They defined and studied the properties of the largest number σ, $0 < \sigma \leq 1$, for which the rate region $\sigma\Lambda$ is achievable (stabilizable) by the LQF policy. This largest number is denoted by $\sigma^*(G)$. It provides a way to measure the performance of LQF on an arbitrary network.

Several other authors studied how to check the local pooling condition or estimate $\sigma^*(G)$ for specific graphs [3, 22, 30].

1.2 A Multiple-Parameter Based Performance Characterization of LQF Policy

In this study, we extend the definition of σ-local pooling further to better characterize the performance of LQF. We show that the one-parameter characterization by σ-local pooling, although attractive for its parsimony, tends to underestimate the stability region that can be achieved by LQF. This leads to the investigation of multiple-parameter characterizations. We start by defining σ-local pooling *for a link*, denoted by σ_l^* for link l. We then construct a diagonal matrix $\Sigma^*(G) = \text{diag}(\sigma_l^*)_{l\in E}$, where E is the set of links. We then show that the linearly transformed region $\Sigma^*(G)\Lambda$ is achievable by LQF.

The relationship between the newly-defined link σ-local pooling and the original σ-local pooling (for the whole network) in [18] is intriguing. We show that $\sigma^*(G) = \min_{l\in E}\sigma_l^*$. In other words, the guaranteed stability region in [18] is derived by linearly transforming the capacity region (with the $\sigma^*(G)I$ matrix, where I is the identity matrix) using the smallest diagonal entry in $\Sigma^*(G)$. As a result, using $\sigma^*(G)$ can lead to severe underestimate of the stability region of LQF. Hence, our new local pooling concept leads to a more accurate performance characterization for LQF.

Throughout the chapter 2, we show that our multiple-parameter refinement of σ-local pooling is at an appropriate level of generality and structural richness, the study of which can provide tools for deeper understanding and operationalization of the local pooling concept. We argue that link σ-local pooling and the associated *limiting set* are fundamental concepts. We also define *set* σ-local pooling (for a set of links), which can be computed by linear programming, and show how it is related to *link* σ-local pooling. The duality theory of linear programming provides means for bounding or estimating the values of various σ-local pooling concepts. We provide an algorithm for estimating the local-pooling factors of links.

1.3 A Non-convex Performance Guarantee under LQF Policy

Even the multiple-parameter characterization of LQF in chapter 2 underestimates the stability region. For instance, it excludes some parts of the capacity region that are obviously stabilizable by LQF. To progress further toward complete performance characterization, there is a need to go beyond the current framework of linear transformations on the capacity region. The goal of the third chapter is to establish such a ”non-linear” framework and expand our knowledge about the achievable rate region by LQF. The main contribution is to describe three new achievable rate regions (Ω, Δ_C and Δ_R), which are all larger than the previously-known regions. More precisely, we show that $\Sigma^*(G)\Lambda \subseteq \Omega$ and $\Omega^o \subseteq \Delta_C \subseteq \Delta_R$. Furthermore, the closures of the new regions include all the extreme points of the capacity region and are not convex in general. This is in contrast to previously-known regions of stability, which are all convex and, in general, exclude some extreme points of the capacity region because they each are derived by reducing the capacity region through a linear transformation. We show that the new regions of stability (or their closures) are convex if and only if they are identical to the capacity region itself. The result implies that, when LQF cannot achieve the full capacity region, the largest achievable region, which is yet to be discovered, cannot be convex.

The characterization of the LQF performance has been substantially improved with these new stability regions. For instance, we have found that, for an arbitrarily large $k > 0$, there are cases where an arrival rate vector λ is outside all the previously-known stability regions but $k\lambda$ is in Ω. In other words, the previously-known stability regions can underestimate the performance of LQF by an arbitrarily large factor in certain cases, whereas the new regions can avoid such poor estimates.

The study has also yielded an interesting, counter-intuitive finding that increasing the arrival rates may sometime help to stabilize the network. We have discovered an example where a rate vector achievable by LQF point-wise dominates another rate vector not achievable by LQF. It turns out the former vector is in the stability region Δ_C whereas the latter is not.

We next summarize the key ideas of the chapter 3. Our theory is developed based on considering the *fluid limit* of an unstable network. A typical scenario is that the maximum queue size has an overall trend to grow indefinitely, which requires that, at some time t, a subset of the current longest queues continues to grow. From the set of the longest queues at time t, there is a subset that grows at the fastest rate and remains the longest in the next infinitesimal time interval. Denote this subset by S. Under LQF, the queues in S will be served with priority in the next small time interval, which implies that the average service rate vector, when restricted to S, comes from the convex hull of the maximal schedules with respect to S. This convex hull is denoted by $Co(M_S)$. For the queues in S, the arrival rates must be larger than the service rates. The discussion motivates the definition of a strictly dominating vector for a queue set S, which is a vector λ, when restricted to S, strictly dominating at least one vector in $Co(M_S)$. After removing the union of the strictly dominating vectors, where the union is over all possible subsets of the queues, we get Ω.

Key to the development about Δ_C is a refinement to the notion of strictly dominating vectors, which is called uniformly dominating vectors. For the aforementioned queue set S, the arrival rates not only must be larger than the service rates, but also larger by the same amount, so that the queues in S grow at the same rate. The removal of all the uniformly dominating vectors gives Δ_C. Although the closure of Δ_C contains Ω, there is value in studying and reporting the results about both regions. First, the theory about Ω provides building blocks for proving some of the results about Δ_C. Second, Ω appears to be well connected to the notion of local pooling in [12], thus, providing some continuity in the theoretical development, whereas Δ_C does not appear so.

Throughout, we assume i.i.d. and mutually independent arrival processes among links. As Dimakis and Walrand pointed out, for the same average arrival rate vector, whether the arrival processes have zero or non-zero variances may lead to significantly different stability behavior (the former is the case of deterministic arrivals with constant rates) [12]. They established a queue separation result for the case of non-zero variances and developed a rank condition that leads to queue separation. We generalize the rank condition. Then, we extend Δ_C to a larger stability region Δ_R for the case of non-zero variances. We also show the closures of Δ_C and Δ_R are the same.

Also, we relate the problems of finding stability conditions under LQF to several problems in the fractional graph theory [32]. The latter provide tools for studying the stability regions introduced by the chapter and for characterizing the set σ-local pooling factor given in [23].

1.4 Channel Fading Effect on Wireless Network Scheduling

The study in chapter 2 and 3 on scheduling problem is based on the protocol model or the binary interference graph model, which has been favored and studied extensively [6, 8, 9, 12, 18–20, 22–26, 35, 37, 41, 43, 46, 47], since it is simple while it successfully captures some of the fundamental difficulties in wireless communication. Alternatively, the research community has investigated physical model where a minimum signal-to-interference ratio (SINR) is required for successful data transmission [5, 16, 21]. Several authors have also studied multi-channel and multi-radio wireless network [39, 44], where the channel assignment problem needs to be well addressed.

In practice, the channel states may not be constant and could change over time. This time-varying property of wireless network is termed channel fading effect and has been studied in the wireless scheduling problem [1, 13, 27–29, 33, 34, 40, 42]. The wireless network with channel fading effect is a generalization of static wireless network model studied previously. Solving the scheduling problem in wireless network with channel fading is as difficult or even more difficult compared with static channel model. To reduce the complexity, some simple sub-optimal scheduling algorithms are in demand.

Recently, the performance of LQF under channel fading model has been studied [29]. The variation of channel states fundamentally changes the wireless scheduling problem. Consequently, the previous result on LQF can not be simply applied. In paper [29], the authors considered the case of ON and OFF channel state [42] for each link. Under certain channel state, only those links that are turned ON participate in the process of scheduling. The authors demonstrate both positive and negative effect of channel fading on the performance of LQF. Some network topology which satisfies σ-local pooling condition will no longer achieve the entire capacity under the channel fading model. On the other hand, the performance of LQF improves in some network topology due to the decoupling/precluding effect of channel fading. The authors also define a notion called Fading Local Pooling Factor(F-LPF), which characterize the performance of LQF under channel fading.

Unfortunately, the limitation of channel fading to the case of ON and OFF channel state may not suit for the real wireless network. The underlying interference graph may also change with time. Two links interfere with each other in one time slot may not interfere with each other in another time slot and vice versa.

The third piece of our work is primarily concerned with the performance of LQF in the interference model under the effect of channel fading. We generalize the channel fading model in [29], so that it allows the variation of interference relationship over time. Specifically, the underlying interference graphs depending on the channel states, which change over time, are usually different for different channel states. Hence, the model is applicable for a wider class of network or channel variations, including those caused by the mobility of links.

We show that the one-parameter characterization by F-LPF [29] tends to underestimate the stability region that can be achieved by LQF. Then, we investigate a multiple-parameter characterizations of LQF in wireless network under channel fading. We first define fading σ-local pooling factor for *a link*, denoted by σ_l^* for link l under the channel fading structure π. Based on this, we construct a diagonal matrix $\Sigma_G^* = \text{diag}(\sigma_l^*)_{l \in V}$, where V is the set of links. We then show that the linearly transformed region $\Sigma_G^* \Lambda$ is achieved by LQF under channel fading structure π.

The relationship between the newly-defined fading link σ-local pooling and the original fading σ-local pooling (for the whole network) in [29] is intriguing. We show that $\sigma^*(G) = \min_{l \in V} \sigma_l^*$, for any graph G under fading structure π. In other words, the guaranteed stability region in [29] is derived by linearly transforming the capacity region (with the $\sigma^*(G)\mathbb{I}$ matrix, where $\mathbb{I}$ is the identity matrix) using the smallest diagonal entry in $\Sigma^*(G)$. As a result, using $\sigma^*(G)$ can lead to severe underestimate of the stability region of LQF under channel fading. Hence, our new local pooling concept leads to a more accurate performance characterization for LQF under channel fading.

Throughout the chapter 4, we show that our multiple-parameter refinement of F-LPF is at an appropriate level of generality. We argue that fading link σ-local pooling and the associated *limiting set* are fundamental concepts. We show how the channel fading set σ-local pooling and channel fading over all σ-local pooling are related to channel fading *link* σ-local pooling. Finally, we provide an algorithm for estimating the fading local-pooling factors of links.

CHAPTER 2
A MULTIPLE-PARAMETER BASED PERFORMANCE CHARACTERIZATION OF LONGEST QUEUE FIRST POLICY IN WIRELESS NETWORKS

In this chapter, we study the performance of LQF policy in wireless networks. We demonstrate that the one-parameter performance characterization of LQF proposed by Joo et al. [18] tends to underestimate the stability region of LQF. This motivates us to investigate a multiple-parameter based performance characterization. With the theory builded in this chapter, we successfully enlarge the known stability region of LQF. Meanwhile, we provide a practical algorithm to estimate the performance lower bound of LQF.

This chapter is organized as follows. In Section 2.1, we provide our network model, basic definitions and notations, and describe the link scheduling problem. In Section 2.2, we describe the main conclusion for performance characterization of LQF using the new notion of link σ-local pooling. In Section 2.3, we develop a fuller theory of link and set σ-local pooling that helps to apply these new concepts. In Section 2.4, we provide methods to estimate or bound the link and set σ-local pooling factors. In Section 2.5, we provide additional theoretical results about σ-local pooling. In Section 2.6, we give additional related work. Section 2.7 summarizes the chapter.

2.1 Preliminaries and Network Model

In our model, a wireless network is represented by a directed graph $G = (V, E, \mathcal{I})$, where V is the set of nodes, E is the set of links and $\mathcal{I}$ represents the interference relation among the links. In this work, we inherit the *protocol model* for interference [16] in which two links cannot be activated simultaneously if they interfere with each other.[1] The interference relation for the protocol model can be represented by a symmetric 0-1 matrix of size $|E| \times |E|$ where a value 1 in an entry indicates the existence of

[1] This is in contrast to the *physical* interference model, where the link rate depends on the power levels of the interfering links in the neighborhood.

pairwise interference between the two corresponding links and 0 indicates the absence of such interference. Equivalently, the interference relation can be represented by the *interference graph* (also known as *conflict graph*) in which a node represents a physical link and an edge represents the existence of interference between two physical links (which are two nodes in the interference graph). The interference graph is denoted by G^I, unless specified otherwise.

We represent a schedule by a $|E|$-dimensional 0-1 vector, where a value 1 in an entry indicates the link is active and 0 indicates otherwise. A feasible schedule corresponds to a set of active links that is free from interfering pairs. A feasible schedule is said to be *maximal* if no more links can be activated without violating the interference constraint. Note that a feasible schedule corresponds to an *independent set* in the interference graph and a maximal schedule corresponds to a maximal independent set.[2]

Let M_E be the matrix whose columns are all the maximal schedules. Occasionally, we also view M_E as the set of all maximal schedules. Let $Co(M_E)$ denote the convex hull of the maximal schedules for the whole network.

For a subset of the links $L \subseteq E$, we can consider the interference relation among the links in L: The interference matrix is a submatrix of the original one with only those rows and columns corresponding to the links in L; the interference graph is the node-induced subgraph of the original interference graph. With this, we can talk about feasibility and maximality of schedules restricted to L, which are those defined with respect to the interference submatrix or subgraph. Similarly, we define M_L to be the matrix (set) of

[2] An implicity assumption is that all active links transmit at the same constant rate. As discussed in [12], the main local pooling-related results are extensible to the case where an active link may transmit at different rates. However, we do not explore this extension in the chapter.

maximal schedules restricted to L, where each column of M_L is $|L|$-dimensional 0-1 vector. Let $Co(M_L)$ denote the convex hull of these maximal schedules.

We assume a time-slotted system, where each slot is of a unit length. We assume the traffic arrival processes to different links are independent on each other. For simplicity, we assume that each arrival process to a link is IID over time. This assumption can be relaxed provided the resulting queueing process is Markovian. (See [12] [11] for the reasons.) For each link $l \in E$, the average arrival rate is denoted by λ_l. We assume single-hop traffic: The traffic is transmitted over only one link and leaves the network after the transmission. Extension to the multi-hop traffic situation needs ideas from [7], but will not be further considered in this chapter.

The capacity region Λ of a network is defined to be the set of all arrival rate vectors $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_{|E|})'$ that can be supported by a time sharing of the feasible schedules. It is easy to see that

$$\Lambda = \{\lambda \mid 0 \leq \lambda \leq \mu \text{ for some } \mu \in Co(M_E)\}. \quad (2–1)$$

In the above, $\lambda \leq \mu$ means λ is component-wise less than or equal to μ. We also need the notion of the capacity region for a set of links $L \subseteq E$. This region is defined analogously by replacing $Co(M_E)$ with $Co(M_L)$ in (2–1) and is denoted by Λ_L. We define the *interior* of Λ as follows, which is denoted by Λ^o.[3]

$$\Lambda^o = \{\lambda \mid 0 \leq \lambda < \mu \text{ for some } \mu \in Co(M_E)\}. \quad (2–2)$$

The interior of Λ_L is similarly defined and is denoted by Λ_L^o.

Tassiulas and Ephremides have shown that the interior of the capacity region can be stabilized by using the *maximum weighted schedule* (MWS) in each time slot where

[3] This definition of interior shouldn't be confused with the usual definition of interior of a set in mathematical analysis, which is an open set.

the weights are the queue sizes at the links [41]. In other words, for any arrival process whose average rate vector λ is in Λ^o, the resulting queueing process is stable if the MWS is used. Here, stability means the queueing process is a positive recurrent Markov process.

However, finding the MWS is a difficult problem. For the queue length vector at time t, $Q(t)$, the problem is to find the schedule $m^*(t)$ such that

$$m^*(t) \in \text{argmax}_{m \in M_E} Q(t)' m. \qquad (2\text{–}3)$$

Hence, the problem is to find the maximum weighted independent set in the interference graph, which is NP-hard for the family of all graphs. Even under the more restricted k-hop interference model, the problem is still NP-hard for $k \geq 2$ [36] [35]. Under the 1-hop interference model, the problem becomes finding the maximum weighted matching, which can be solved in $O(|V|^3)$. However, the complexity is still very high.

The LQF schedule can be viewed as an approximation to the MWS. LQF operates as follows at each time slot t. The link with the largest backlog is activated and all links interfering with it are discarded. Next, the same procedure is applied to the remaining links: The link with the largest backlog among the remaining links is found and activated and the interfering links are discarded. The procedure is applied recursively until all links are either activated or discarded. The LQF algorithm requires sorting the backlog values; however, this can be done reasonably efficiently in a distributed manner. Further improvement has been made by Joo [17], whose local greedy scheduling does not require global sorting of the queues. His algorithm is to have the links with the local maximum queue sizes in their neighborhoods to transmit data, subject to the interference constraints. He showed that the local scheduling algorithm achieves the same stability region as LQF. Hence, from the MWS to LQF, a globally optimized decision is replaced by distributed, local decisions.

Notations: The following notations are used. Given a vector u, let u' denote the transpose of u. For two vectors u and v, $u'v$ means the inner product of the two; $u \leq v$ means u is component-wise less than or equal to v; the meanings of $u \geq v$, $u < v$ or $u > v$ are also component-wise. The symbols $\not>$ and $\not\geq$ are the negations of $>$ and $\geq$, respectively. For instance, by $u \not> v$, we mean that u is not component-wise greater than v; that is, there exists a component k such that $u_k \leq v_k$. For a vector $\lambda \in \mathbb{R}^{|E|}$ and for a set of links $L \subseteq E$, we denote the $|L|$-dimensional vector $[\lambda]_L$ as the restriction of λ to those dimensions associated to L.

2.2 σ-Local Pooling and the Performance of LQF

In this section, we introduce our main notion of *link* σ-local pooling and relate it to the original σ-local pooling (for the network graph) by Joo et al. [18]. We will call the latter *network* σ-local pooling. We show how to capture a larger stability region of LQF using this new notion of link σ-local pooling. By stability region, we mean a rate region in Λ for which LQF leads to a stable queueing process. We start by reviewing the σ-local pooling related results in [18].

2.2.1 Review of Network σ-Local Pooling by Joo et al.

Joo et al. investigated a single-parameter performance characterization of a scheduling policy [18]. In this case, the scheduling policy is LQF. They defined the efficiency ratio $\gamma^*(G)$ of a scheduling policy as follows

Definition 1. *The efficiency ratio $\gamma^*(G)$ of a scheduling policy for a given network graph G is:*

$$\gamma^*(G) := \sup\{\gamma | \textit{The network is stable for all arrival rate vectors } \lambda \in \gamma\Lambda^o\}.$$

Joo et al. showed that $\gamma^*(G)$ is equal to the σ-local pooling factor for the network G, i.e.,

$$\gamma^*(G) = \sigma^*(G). \quad (2\text{–}4)$$

$\sigma^*(G)$ depends only on the topological and interference structures of the network and is defined as follows.

Definition 2. *The* local pooling factor $\sigma^*(G)$ *of a network graph* $G = (V, E, \mathcal{I})$ *is:*

$$\begin{aligned}\sigma^*(G) := &\sup\{\sigma | \sigma\mu \ngeq \nu \text{ for all } L \subseteq E \text{ and all} \\ &\mu, \nu \in Co(M_L)\} \qquad (2\text{–}5)\\ := &\inf\{\sigma | \sigma\mu \geq \nu \text{ for some } L \subseteq E \text{ and some} \\ &\mu, \nu \in Co(M_L)\}. \qquad (2\text{–}6)\end{aligned}$$

2.2.2 Motivation of Defining Link σ-Local Pooling

Note that, in the case of $\gamma^*(G) < 1$, $\gamma^*(G)\Lambda$ shrinks all dimensions of the capacity region by the same scaling factor to ensure stability. This can be overly conservative in that some links may be required to reduce the arrival rates more than necessary. This means that using a single factor to characterize the performance will underestimate the achievable rates of these links. Since links are all different in terms of the interference constraints they face, the reduction factors should be non-uniform across the links. Consider the network examples in Fig. 2-1 under the 1-hop interference model. The interference graph for the 6-cycle network is still a 6-cycle, which has been well studied in [8, 12, 18]. It has been shown that for the 6-cycle, $\gamma^*(G) = \sigma^*(G) = 2/3$. This is also the case for the 8-link network, although the interference graph is slightly more complicated.

However, for link 8 of the 8-link network on the right, there is no need to reduce its arrival rate by $2/3$ to achieve stability. To see this, for any arrival rate vector inside the capacity region, we have $\lambda_7 + \lambda_8 < 1$. Under 1-hop interference, exactly one link from link 7 and link 8 must be activated in any maximal schedule, provided queue 8 (i.e., the queue of link 8) is not empty. Hence, the total queue length of the two must decrease when queue 8 is non-empty. The decrease will continue until queue 8 becomes empty. The time taken by this is bounded by the time taken for queue 7 to drift down to the size of queue 8, if queue 7 starts to be larger than queue 8, plus the time taken for both queues to drift down to zero from that point. Hence, if queue 7 is stable, then queue 8 is also stable.

The reason why links have different performance efficiency is that the efficiency ratio is related with the structure of certain "bottleneck" subgraph containing the link in question. The characterization of efficiency is determined by this subgraph but not by the whole graph in general. For different links, such subgraphs are different. For instance, for link 2 in the 8-link network in Fig. 2-1, the subgraph is the six cycle, which leads to its efficiency ratio to be $2/3$. For link 8, the subgraph is link 8 itself, which leads its efficiency ratio to be 1. We will later identify this kind of subgraphs more explicitly.

2.2.3 Definitions of Link σ-Local Pooling

The above discussion motivates us to come up with a performance characterization for each link individually. We next extend the concept of *network* σ-local pooling in Definition 2 to *link* σ-local pooling. This characterization is obtained from the topology and interference structure of a sub-network graph containing link l in question.

Definition 3. *Given a network graph* $G = (V, E, \mathcal{I})$*, the* local pooling factor of a link $l \in E$*, denoted by* σ_l^**, is*

$$\sigma_l^* := \sup\{\sigma | \sigma\mu \not> \nu \textit{ for all } L \subseteq E \textit{ such that } l \in L, \textit{ and all } \mu, \nu \in Co(M_L)\} \tag{2–7}$$

$$:= \inf\{\sigma | \sigma\mu > \nu \textit{ for some } L \subseteq E \textit{ such that } l \in L, \textit{ and some } \mu, \nu \in Co(M_L)\}. \tag{2–8}$$

Compared with Definition 2, the subset L in Definition 11 is required to contain link l. Moreover, in (2–8), the inequality is strict as opposed to the non-strict inequality in (2–6). This in turn leads to the difference ($\not>$ as opposed to $\not\geq$) between the two supremum-based definitions. We will show later that the strict inequality is a tighter condition for the stability proof. However, the σ_l^* value is unchanged whether the strict or non-strict inequality is used.

The following lemma relates the original network σ-local pooling with the new link σ-local pooling. The proof is obvious after we replace the strict inequality with a non-strict inequality in (2–8).

Lemma 1. *For a network graph $G = (V, E, \mathcal{I})$, the following holds.*

$$\sigma^*(G) = \min_{l \in E} \sigma_l^*. \tag{2–9}$$

2.2.4 Link-Based Performance Guarantee of LQF

Lemma 1 makes it clear that the original network σ-local pooling factor in [18] is equal to the smallest of the σ-local pooling factors for the links. In [18], the LQF performance is bounded by the lowest link σ-local pooling factor. In contrast, we can show the following improved performance bound for LQF, which takes into account all σ_l^*. Let $\Sigma^*(G)$ be the $|E| \times |E|$ diagonal matrix whose diagonal entries are σ_l^* for $l \in E$. That is, $\Sigma^*(G) = \text{diag}(\sigma_l^*)_{l \in E}$. The next theorem is one of the main results in the paper.

Theorem 2.1. *Given a network graph $G = (V, E, \mathcal{I})$, if an arrival rate vector λ satisfies $\lambda \in \Sigma^*(G)\Lambda^o$, then, the network is stable under the LQF policy.*

Proof. We will consider the fluid limit of the queue process, denoted by $\{q_l(t)\}_{t \geq 0}$, for all $l \in E$. (See [11] [12] [18] for more details on this approach.) Consider a fixed time instance t. Let L be the set of those longest queues (with equal length) whose time derivatives at t, $\dot{q}_l(t)$, are the largest (also identical) under the given LQF policy (that is, an instance of the LQF policy that is being used.). The queues in L will remain the longest with identical length in the next infinitesimally small time interval.

Let $l \in \text{argmax}_{k \in L} \sigma_k^*$. Since $\lambda \in \Sigma^*(G)\Lambda^o$, there exists $\mu \in Co(M_E)$ such that $\lambda < \Sigma^*(G)\mu$. This implies $[\lambda]_L < \Sigma_L^*[\mu]_L$, where Σ_L^* denotes the restriction of $\Sigma^*(G)$ to L, i.e., the diagonal submatrix of $\Sigma^*(G)$ with only the rows and columns corresponding to the set L. Hence, $[\lambda]_L < \sigma_l^*[\mu]_L$.

It is easy to see that $[\mu]_L \in \Lambda_L$. Hence, there exists $\mu_L \in Co(M_L)$ such that $[\mu]_L \leq \mu_L$. Then, $[\lambda]_L < \sigma_l^* \mu_L$. Given λ, let us suppose the way of picking such a μ_L is well-defined. Let $\epsilon_L = \min_{k \in L}(\sigma_l^* \mu_L(k) - \lambda(k))$.[4] We have $\epsilon_L > 0$.

For such fixed λ and μ_L, consider any arbitrary $\nu_L \in Co(M_L)$. We must have $\sigma_l^* \mu_L \not> \nu_L$ by the definition of link σ-local pooling. Hence, there exists a link $k \in L$ such that $\sigma_l^* \mu_L(k) \leq \nu_L(k)$. For such a k, since $\lambda(k) < \sigma_l^* \mu_L(k) \leq \nu_L(k)$, we have $\nu_L(k) - \lambda(k) \geq \epsilon_L$. Hence, $\max_{k \in L}(\nu_L(k) - \lambda(k)) \geq \epsilon_L$. Note that ϵ_L is independent of ν_L.

Note that the service rate vector, when restricted to L, must belong to the set $Co(M_L)$. Roughly, this is because L contains all the queues that are among the longest and remain longest in the near future, and hence, every LQF schedule being used must be a maximal schedule when restricted to L. (See [11] [12] for a more rigorous argument for this.)

Now imagine ν_L is the service rate vector at the current time t. We have just shown that, for some $k \in L$, $\nu_L(k) - \lambda(k)$ is at least ϵ_L. Hence, the queue at link k decreases at a rate no less than ϵ_L. Since the queues in the set L change at the same rate, they all decrease at a rate no less than ϵ_L. Hence, each of the longest queues decreases its size at a rate no less than ϵ_L. Let $\epsilon = \min\{\epsilon_L | L \subseteq E\}$. Since the number of possible subsets of E is finite, we have $\epsilon > 0$. Hence, at any time instance, each of the longest queues decreases at a positive rate no less than ϵ. By [11], this is sufficient to conclude that the original queueing process is a positive recurrent Markov process, which means the queues are stable by definition. $\square$

To summarize, Joo et al. showed in [18] that the region $\sigma^*(G)\Lambda^o$ is stable under LQF. Our Theorem 2.1 shows that the region $\Sigma^*(G)\Lambda^o$, which contains $\sigma^*(G)\Lambda^o$, is

[4] Given a vector ν, we write its component corresponding to link k by ν_k or $\nu(k)$ interchangeably.

stable under LQF. For the example of the 8-link network in Fig. 2-1, $\sigma^*(G) = 2/3$ whereas $\Sigma^*(G) = \text{diag}(2/3, 2/3, 2/3, 2/3, 2/3, 2/3, 1, 1)$.[5]

2.3 Theory of Link σ-Local Pooling

In this section, we show some important properties of the newly defined concept of link σ-local pooling and the related concept of *limiting set*. In general, these concepts are difficult to work with since their definitions involve combinatorial enumerations. Our objective is to provide tools for using or applying these concepts. As will be shown, some of the theories developed in this section can help to estimate the link σ-local pooling factors. We will also argue that link σ-local pooling and limiting set are fundamental concepts. The understanding of them may help to reveal deeper structures and key insights about wireless link scheduling.

2.3.1 σ-Local Pooling of a Set

In order to develop the intended theory, it is convenient to first define *set* σ-local pooling. This concept can be used as a building block for performance characterization of the LQF policy for a set of links in the network graph. Since links in a wireless network may exert influence on each other, it is natural to study a set of links as a whole.

Suppose $L \subseteq E$ and L is non-empty. For convenience, let

$$\Theta_L = \{\sigma \mid \sigma\mu_L \ngtr \nu_L, \text{for all } \mu_L, \nu_L \in Co(M_L)\ \}. \tag{2–10}$$

The compliment of Θ_L is

$$\Theta_L^c = \{\sigma \mid \sigma\mu_L > \nu_L, \text{for some } \mu_L, \nu_L \in Co(M_L)\ \}. \tag{2–11}$$

[5] This may not be obvious now, but can be shown easily by applying the theory to be developed subsequently.

Definition 4. *Given a non-empty set $L \subseteq E$, we say L has a* set σ-local pooling factor σ_L^* if the following holds.

$$\sigma_L^* := \sup\{\sigma \mid \sigma \in \Theta_L\} \tag{2–12}$$

$$:= \inf\{\sigma \mid \sigma \in \Theta_L^c\}. \tag{2–13}$$

Note that, unlike the definition of network σ-local pooling in Definition 2, the definition of set σ-local pooling for a set L does not involve subsets of L.

The following are some elementary facts. Since $0 \notin Co(M_L)$, by (2–11), $\sigma_L^* > 0$. By considering $\mu_L = \nu_L$ in (2–10), we see that $\sigma_L^* \leq 1$. If $\sigma \in \Theta_L^c$, then $(\sigma - \epsilon)\mu_L > \nu_L$ for small enough $\epsilon > 0$, where L, μ_L and ν_L are as in the definition of Θ_L^c. Hence, Θ_L^c is an open set on $\mathbb{R}$. In fact, $\Theta_L^c = (\sigma_L^*, \infty)$; $\Theta_L = [0, \sigma_L^*]$.

The following lemma says that σ_L^* can be found by a well-defined optimization problem where the constraint region is a closed set. The constraint region can be thought as being compact when taking into account the fact that σ_L^* is bounded from above by 1. The fact in the lemma needs to be explicitly stated since the infimum-based definition in (2–13) is not over a closed set in variables (σ, μ_L, ν_L).

Lemma 2. *For any non-empty $L \subseteq E$, σ_L^* is the optimal value of the following optimization problem.*

$$\textbf{\textit{(I)}} \min_{\sigma, \mu_L, \nu_L} \sigma \tag{2–14}$$

$$\textit{subject to } \sigma\mu_L \geq \nu_L \tag{2–15}$$

$$\mu_L, \nu_L \in Co(M_L). \tag{2–16}$$

Proof. By (2–11) and (2–13), σ_L^* is the optimal value of the following problem.

$$\textbf{(II)} \inf_{\sigma,\mu_L,\nu_L} \sigma \tag{2–17}$$

$$\text{subject to } \sigma\mu_L > \nu_L \tag{2–18}$$

$$\mu_L, \nu_L \in Co(M_L). \tag{2–19}$$

Let the constraint sets of the optimization problem (I) and (II) be denoted by S_1 and S_2 ($S_2 = \Theta_L^c$), respectively, both of which lie in $\mathbb{R} \times \mathbb{R}^{|L|} \times \mathbb{R}^{|L|}$. We will show that S_1 is the closure of S_2. For this, we need to show that every point in $S_1 \backslash S_2$ is a limit point of S_2. Let a point $(\sigma, \mu_L, \nu_L) \in S_1 \backslash S_2$, which is characterized by $\sigma\mu_L \geq \nu_L$ with $\sigma\mu_k = \nu_k$ for some $k \in L$. If $\mu_L > 0$, we only need to increase σ by a little bit to find a point in S_2. To handle the general case where $\mu_k = 0$ for some $k \in L$, note that $\nu_k = 0$. Since M_L includes all maximal vectors, there exists $\omega_L \in Co(M_L)$ such that $\omega_L > 0$. Then, a vector $\hat{\mu}_L = (1 - \epsilon_1)\mu_L + \epsilon_1\omega_L$, where $0 < \epsilon_1 \leq 1$, has the property that $\hat{\mu}_L > 0$. Note that $\hat{\mu}_L \in Co(M_L)$. We can choose ϵ_1 small enough and choose $\epsilon_2 > 0$ accordingly such that $(\sigma + \epsilon_2)\hat{\mu}_L > \nu_L$ and also $(\sigma + \epsilon_2, \hat{\mu}_L, \nu_L)$ is in the ϵ-open ball around (σ, μ_L, ν_L). Hence, S_1 is the closure of S_2.

Next, $(\sigma, \mu_L, \nu_L) \mapsto \sigma$ is a continuous function on $\mathbb{R} \times \mathbb{R}^{|L|} \times \mathbb{R}^{|L|}$. Hence, the two problems have the same optimal value and the optimum is attained for problem (I). $\square$

Since the optimization problem (I) has a continuous objective function and the constraint set is closed, the optimum is attained in its constraint set. The optimization problem (II) is not attained. The following lemma makes this more precise.

Lemma 3. *For any non-empty set $L \in E$, the optimum solution to the optimization problem (I) satisfies $\sigma_L^*\mu_L^* \geq \nu_L^*$ with $\sigma_L^*\mu_k = \nu_k$ for some $k \in L$, where $\mu_L^*, \nu_L^* \in Co(M_L)$ and σ_L^* is the σ-local pooling factor for set L. Furthermore, such k is not unique.*

Proof. Suppose we have $\sigma_L^*\mu_L^* > \nu_L^*$. Then, $(\sigma_L^* - \epsilon)\mu_L^* \geq \nu_L^*$ for small enough $\epsilon > 0$, and σ_L^* cannot be optimal.

We next show such k is not unique; in other words, there are at least two components achieving equality in an optimal solution. The proof is by contradiction. Suppose there is only one k such that $\sigma_L^* \mu_k^* = \nu_k^*$, and $\sigma_L^* \mu_i^* > \nu_i^*$ for all $i \neq k, i \in L$. Consider two cases. The first case is $\sigma_L^* \mu_k^* = \nu_k^* > 0$. Note that there is a vector $\hat{\nu} \in Co(M_L)$ such that $\hat{\nu}_k = 0$. By choosing a small enough $\epsilon > 0$, we can obtain a new vector $\tilde{\nu} = (1 - \epsilon)\nu^* + \epsilon \hat{\nu}$ such that $\sigma_L^* \mu_i^* > \tilde{\nu}_i$ for all $i \in L$. Consider the second case where $\sigma_L^* \mu_k^* = \nu_k^* = 0$. Note that there must be a vector $\hat{\mu} \in Co(M_L)$ such that $\hat{\mu}_k > 0$. By choosing a small enough $\epsilon > 0$, we can obtain a new vector $\tilde{\mu} = (1 - \epsilon)\mu^* + \epsilon \hat{\mu}$ such that $\sigma_L^* \tilde{\mu}_i > \nu_i^*$ for all $i \in L$. In either case, the conclusion contradicts the definition of σ_L^* (or equivalently, the first part of this lemma). □

The optimization characterization of σ_L^* is useful since one can apply the duality theory to derive important results and insights. The problem (I) has an alternative form, which is a linear program. From the linear program, we can obtain the following dual problem. Suppose M_L has $c(L)$ columns. Let e_n be the vector $(1, 1, \dots, 1)'$ with n 1's.

Lemma 4. *σ_L^* is the optimal value of the following optimization problem.*

$$
\begin{aligned}
\textbf{\textit{(Dual)}} \quad & \max_{x \geq 0, w} \; w \\
\textit{subject to} \quad & x' M_L \leq e'_{c(L)} \\
& x' M_L \geq w e'_{c(L)}.
\end{aligned}
$$

Proof. The problem (Dual) is the dual problem of a linear version of the problem (I). First, rewrite the problem (I).

$$\min_{\alpha,\beta,\sigma} \; \sigma$$

$$\text{subject to} \;\; \sigma M_L \alpha \geq M_L \beta$$

$$\alpha' e_{c(L)} = 1$$

$$\beta' e_{c(L)} = 1$$

$$\alpha, \beta \geq 0.$$

Let $\gamma = \sigma\alpha$. The problem can be written as,

$$\min_{\gamma,\beta,\sigma} \; \sigma \tag{2–20}$$

$$\text{subject to} \;\; M_L \gamma \geq M_L \beta \tag{2–21}$$

$$\gamma' e_{c(L)} = \sigma \tag{2–22}$$

$$\beta' e_{c(L)} = 1 \tag{2–23}$$

$$\gamma, \beta \geq 0. \tag{2–24}$$

Let x, y, z to be dual variables associated with (2–21), (2–22) and (2–23), respectively. Then, the dual problem is

$$\max_{x \geq 0, z} \; -z$$

$$\text{subject to} \;\; x' M_L + z e'_{c(L)} \geq 0$$

$$y e'_{c(L)} - x' M_L \geq 0$$

$$y = 1.$$

Let $w = -z$, we get the optimization problem in the lemma. $\square$

Remark 1: From Lemma 4, a set $L \subseteq E$ is σ_L^*-local pooling if and only if σ_L^* is the largest number for which $\sigma_L^* e'_{c(L)} \leq x' M_L \leq e'_{c(L)}$ holds for some $x \geq 0$. In the special case of

$\sigma_L^* = 1$, we see that L is local pooling if and only if there exists some $x \geq 0$ such that $x'M_L = e'_{c(L)}$, or equivalently, there exists some nonzero $x \geq 0$ such that the components of the vector $x'M_L$ are all identical. The latter statement coincides with the original definition of local pooling in [12].

Remark 2: The optimization problem (2–20)-(2–24) can be rewritten as follows.

$$\min_{\gamma,\beta} \sum_{i=1}^{c(L)} \gamma_i$$

$$\text{subject to} \quad M_L\gamma \geq M_L\beta$$

$$\beta' e_{c(L)} = 1$$

$$\gamma, \beta \geq 0.$$

Consider an optimal solution (γ^*, β^*) and $\sigma^* = \sum_{i=1}^{c(L)} \gamma_i^*$. We can interpret $\nu_L^* = M_L\beta^*$ as achieving the service rates ν_L^* by time sharing of the maximal schedules with the time shares β_i^*. Then, $M_L\gamma^*$ is an alternative way of time sharing the maximal schedules that achieves at least link rates ν_L^*, but with the least amount of time, σ^*. That is, γ^* is the most compact schedule in terms of time. Therefore, σ^* is the largest degree (smallest number) at which any time-sharing schedule (i.e., one in $Co(M_L)$) can be packed.

2.3.2 Relation between Set and Link σ-Local Pooling

The development of set σ-local pooling serves as a basis for better understanding of link σ-local pooling. The performance limitation of a link is related to all subsets of links containing the link itself. Therefore, some of the results about set σ-local pooling can be applied here. The performance of the links is generally not uniform due to the fact that each link is associated with a different collection of subsets.

Lemma 5. *For a link $l \in E$, σ_l^* is the smallest σ_L^* for all $L \subseteq E$ that contains l, i.e.,*

$$\sigma_l^* = \min_{\{L \subseteq E \mid l \in L\}} \sigma_L^* \tag{2–25}$$

Proof. The proof is by definition of σ_l^* and σ_L^*. $\square$

We have the following lemma indicating the relationship between the local pooling factor for sets and the local pooling factor for links.

Corollary 1. *Let $L \subseteq E$ be an arbitrary non-empty set. For all $l \in L$, $\sigma_L^* \geq \sigma_l^*$.*

2.3.3 Limiting Set

We next study the situation where the set σ-local pooling factor is equal to the link σ-local pooling factor for a link in the set. When equality holds, we see that the efficiency ratio, i.e., the link σ-local pooling factor, is limited by the set.

Loosely speaking, a limiting set for a link l is a subset of the links, $L \subseteq E$ with $l \in L$, that "achieves" σ_l^* (for instance, see the infimum definition of σ_l^* in (2–8)). The significance of a limiting set is that it is the set of links whose interference with l prevents σ_l^* from becoming larger, hence, the term *limiting*. Therefore, it is the limiting set for a link, instead of the complete network, that represents structural constraints for the link. While the network can be large, the limiting set for a link may contain a much smaller number of links. Hence, finding the limiting set and understanding its properties have both theoretical and practical significance.

Definition 5. *For any link $l \in E$, a set $L \subseteq E$ is called a* limiting set for link l *if $l \in L$ and there exist $\mu_L, \nu_L \in Co(M_L)$ such that $\sigma_l^* \mu_L \geq \nu_L$.*

Lemma 6. *For any link l, a limiting set for l exists.*

Proof. The proof is omitted for brevity. □

Note that the limiting set for a link is not necessarily unique.

Lemma 7. *A set $L \subseteq E$ containing link l is a limiting set for l if and only if $\sigma_L^* = \sigma_l^*$.*

Proof. Suppose L is a limiting set for l. Then, there exist $\mu_L, \nu_L \in Co(M_L)$ such that $\sigma_l^* \mu_L \geq \nu_L$. By Lemma 2, $\sigma_L^* \leq \sigma_l^*$. Combining this with Corollary 1, we have $\sigma_L^* = \sigma_l^*$.

Conversely, suppose $\sigma_L^* = \sigma_l^*$. Then, by Lemma 2, there exist $\mu_L, \nu_L \in Co(M_L)$ such that $\sigma_l^* \mu_L \geq \nu_L$. By the definition, L is a limiting set for l. □

Corollary 2. *Given a non-empty set of links $L \subseteq E$, if $\sigma_L^* = \max_{l \in L} \sigma_l^*$, then L is a limiting set for each link in the set argmax$_{l \in L} \sigma_l^*$.*

For a link l with $\sigma_l^* = 1$, any set L containing l is a limiting set for l, since we can choose $\mu_L = \nu_L$ in Definition 5. Hence, when $\sigma_l^* = 1$, the notion of limiting set is trivial, and the corresponding limiting sets are called *trivial*. Only when $\sigma_l^* < 1$, the notion is consequential.

Lemma 8. *Consider a link $l \in E$ with $\sigma_l^* < 1$, and let L be a limiting set for l. In the interference graph G^l, every node (link in the network graph G) in L has a (interference) degree of at least 2 with respect to L. Hence, the subgraph of G^l induced by L contains at least one cycle.*

Proof. The proof is by contradiction. Consider the interference graph, G^l. Suppose in the limiting set L for l, some node in L, say p, has a degree either 0 or 1 in L. Then, we can construct a vector x as follows. When p has a degree 0 in L, the entry of x corresponding to p is set to 1 and all other entries are set to 0; when p's degree in L is 1, let the entries corresponding to p and p's only neighbor in L be equal to 1, and set all other entries to 0. Then, the problem (Dual) in Lemma 4 has the optimal value 1, which implies $\sigma_L^* = 1$. By Corollary 1, $\sigma_l^* \geq \sigma_L^* = 1$. Hence, we have $\sigma_l^* = 1$, which contradicts the assumption of the current lemma. Therefore, every node in the limiting set L should have a degree at least 2 in L. Then, there must exist a cycle in the subgraph of G^l induced by the node set L. □

Lemma 8 gives a necessary condition for any non-trivial limiting set. When we want to find a link σ-local pooling factor or the limiting set for a link, we can apply this lemma to reach conclusions or prune the search space. For instance, in the 8-link network of Fig. 2-1, link 8 has $\sigma_8^* = 1$, since in the interference graph, the node corresponding to link 8 has a degree 1. More generally, any link that interferes with no more than one other link must have a link σ-local pooling factor equal to 1.

Lemma 9. *For any link $l \in E$, one of its limiting sets induces a connected subgraph in the interference graph for the network.*

Proof. Let G^I denote the interference graph. Suppose L is an arbitrary limiting set for l. If L induces a connected subgraph of G^I, then there is nothing to prove. Otherwise, let $L' \subseteq L$ with $l \in L'$ be the largest subset of L that contains l and induces a connected subgraph of G^I. We now only need to show that L' is also a limiting set for l.

Since L is a limiting set for link l, by Lemma 7, we must have $\sigma_l^* = \sigma_L^*$. From Lemma 2, we know there are two vectors $\mu, \nu \in Co(M_L)$ such that $\sigma_L^* \mu \geq \nu$. Since L' is the largest subset of L containing l that induces a connected subgraph of G^I, any (physical) link in L' does not interfere with any link in $L \setminus L'$. Thus, $[\mu]_{L'}, [\nu]_{L'} \in Co(M_{L'})$ and $\sigma_l^*[\mu]_{L'} \geq [\nu]_{L'}$. Therefore, by Lemma 2, $\sigma_l^* \geq \sigma_{L'}^*$. Also, since $l \in L'$, we have $\sigma_l^* \leq \sigma_{L'}^*$ by Corollary 1. Hence, $\sigma_l^* = \sigma_{L'}^*$. According to Lemma 7, L' is a limiting set for l. □

Remark: Lemma 9 shows that for any link l in the graph, only those subsets of the links (containing l) that induce connected subgraphs of the interference graph may further limit link l's performance under LQF. When we calculate the local pooling factor for link l, we only need to inspect these subsets.

2.3.4 Performance Guarantees of LQF - A Revisit

With the development of set σ-local pooling, we can state the following sufficient condition for stability under LQF.

Theorem 2.2. *Given a network graph $G = (V, E, \mathcal{I})$, suppose the arrival rate vector λ satisfies the condition that, for every non-empty $L \subseteq E$, $[\lambda]_L \in \sigma_L^* \Lambda_L^o$. Then, the network is stable under the LQF policy.*

Proof. Since the proof is similar to that for Theorem 2.1, we will be brief and omit some arguments, which can be found in the proof for Theorem 2.1. We will consider the fluid limit of the queue process, denoted by $\{q_l(t)\}_{t \geq 0}$, for all $l \in E$. Consider a fixed time instance t. Let L be the set of those longest queues (with equal length) whose time derivatives at t, $\dot{q}_l(t)$, are the largest (also identical) under the particular LQF policy being used.

By the assumption of the theorem, there exists $\mu_L \in Co(M_L)$ such that $[\lambda]_L < \sigma_L^* \mu_L$. For this μ_L and any other $\nu_L \in Co(M_L)$, $\sigma_L^* \mu_L \not> \nu_L$ by the definition of σ_L^*. Hence, there exists a link $k \in L$ such that $\sigma_L^* \mu_k \leq \nu_k$. Then, $\lambda_k < \nu_k$. If ν_L is the service rate vector (in the fluid limit) for the queues in L, the queue at link k decreases at the rate $\nu_k - \lambda_k$. Since all queues in the set L change at the same rate, they all decrease at the rate $\nu_k - \lambda_k$, which is positive. □

With the relationship between set and link σ-local pooling, we can show Theorem 2.1 is implied by Theorem 2.2. Hence, the condition of Theorem 2.2 for stability under LQF is more general than that of Theorem 2.1. This shows one of the utilities provided by our theoretical development of set and link σ-local pooling.

Proof. (Alternative Proof of Theorem 2.1) Consider any link set $L \subseteq E$. Let $l \in \text{argmax}_{k \in L} \sigma_k^*$. Since $\lambda \in \Sigma^*(G)\Lambda^o$, there exists $\mu \in Co(M_E)$ such that $\lambda < \Sigma^*(G)\mu$. This implies $[\lambda]_L < \Sigma_L^*[\mu]_L$, where Σ_L^* denotes the restriction of $\Sigma^*(G)$ to L, i.e., the diagonal submatrix of $\Sigma^*(G)$ with only the rows and columns corresponding to the set L. Hence, $[\lambda]_L < \sigma_l^*[\mu]_L \leq \sigma_L^*[\mu]_L$, where we have used Corollary 1 in the second inequality. It is easy to see that there exists $\hat{\mu}_L \in Co(M_L)$ such that $[\mu]_L \leq \hat{\mu}_L$. Hence, $[\lambda]_L \in \sigma_L^* \Lambda_L^o$. By Theorem 2.2, the queues are stable under LQF. □

2.4 Estimating $\Sigma^*(G)$ Matrix

2.4.1 Estimating σ-Local Pooling Factor for Set

In Section 2.3.1, we introduced a linear programming formulation (LP), (2–20)-(2–24), for calculating the σ-local pooling factor for a set of links. Although linear programs can be solved in polynomial time in terms of the problem size, our formulation contains exponentially many decision variables and is computationally intractable for large networks. This section concentrates on providing methods to estimate σ_L^*. We find defining the problem on the interference graph to be simpler. Accordingly, the following observations are made primarily on the interference graph. Recall that a node in the interference graph corresponds to a link in the original network. As a result, a maximal schedule corresponds to a maximal independent set in the interference graph. Unless mentioned otherwise, the interference graph in this subsection refers to the subgraph of G^I induced by the set L.

Consider the dual problem in Lemma 4. We observe that the dual LP is a weight assignment problem on the nodes in the interference graph. Consider a fixed set $L \subseteq E$. Let $\{s^1, s^2, s^3, \ldots, s^t\}$ represent all the maximal schedules with respect to set L, i.e., each s^i is the i^{th} column of the matrix M_L. Consequently, the dual problem can be rewritten as follows.

$$\max_{x \geq 0, w} \; w \tag{2–26}$$

$$\text{subject to} \quad \max_i x' s^i \leq 1 \tag{2–27}$$

$$\min_i x' s^i \geq w. \tag{2–28}$$

Entries of the x vector in the dual problem can be interpreted as the weights assigned to the nodes in the interference graph. We define *the weight of a schedule* to be the sum of the weights of all active nodes in the schedule. The dual problem strives to balance the weights of the maximal schedules. It is easy to see that, in an optimal solution to the dual problem, denoted by (w^*, x^*), equality is achieved in both (2–27) and (2–28) by some schedules. Otherwise, the objective value can be further improved. Hence, in an optimal solution, the weight of any maximum-weight schedule is forced to 1. This can be interpreted as normalizing the weight assignment according to the maximum-weight schedule. The weight of any minimum-weight schedule is w^*. With some thought, the weight assignment problem can be reformulated as finding node weights to maximize the ratio between the minimum and the maximum schedule weights. That is,[6]

$$\sigma_L^* = w^* = \max_{x \geq 0} \frac{\min_i x's^i}{\max_i x's^i}. \tag{2–29}$$

The new formulation in (2–29) provides a simple way to derive a lower bound for σ_L^*, which is by assigning some particular weights to the nodes in the interference graph and calculating the ratio between the minimum and the maximum schedule weights. Next, we will use this idea to derive lower-bounds on σ_L^*. We denote the component sum of a vector s by $||s||_1$, which is the 1-norm of s. Then, we have the following.

Lemma 10. *For a non-empty set $L \subseteq E$, suppose the maximal schedules are $M_L = (s^1, s^2, s^3, \dots, s^t)$ where s^i is a vector corresponding to the i^{th} maximal schedule with respect to L. Then,*

$$\sigma_L^* \geq \frac{\min_i ||s^i||_1}{\max_i ||s^i||_1}. \tag{2–30}$$

Proof. In (2–29), we assign identical weights to all nodes in L, i.e, $x_j = 1$ for all $j \in L$. □

Remark: A similar result was also given in [4] and [22].

For an interference graph that forms a single cycle, the lower bound in Lemma 10 is in fact achieved.

Lemma 11. *Suppose the interference graph corresponding to L forms a cycle. Then,*

$$\sigma_L^* = \frac{\min_i ||s^i||_1}{\max_i ||s^i||_1}. \tag{2–31}$$

Proof. See Corollary 3 later. □

Next, we give a lower bound of σ_L^* for interference graphs that are cycles.

Lemma 12. *Suppose the interference graph corresponding to L forms a cycle. Then, we have $\sigma_L^* \geq 2/3$.*

[6] We assume the convention $\frac{0}{0} = 0$ so that $x = 0$ is not optimal.

Proof. In any maximal schedule for a cycle interference graph, there must be at least one node active among any three consecutive nodes. So, for any schedule s^i, $||s^i||_1 \geq |L|/3$. On the other hand, since any two consecutive nodes cannot be activated simultaneously, there are at most $|L|/2$ active nodes in s^i. Therefore, $||s^i||_1 \leq |L|/2$. By applying Lemma 11, we have $\sigma_L^* \geq 2/3$. □

Based on the proof of Lemma 12, as the number of nodes in the cycle increases, σ_L^* eventually approaches $2/3$. Hence, only cycles with a small number of nodes (but greater than 6) can have σ_L^* significantly different from $2/3$.

The bounding approach in Lemma 10 works well for cycles. However, the following example illustrates that this approach can produce arbitrarily small lower bounds for some network topologies. Consider the interference graph in Fig. 2-2 with 9 nodes, which is an instance of the star graph S_k for $k = 9$. If we assign identical weights to all nodes in the graph and compute the ratio in Lemma 10, we will end up getting the ratio between the largest and smallest cardinality of the maximal schedules, which is $1/8$. As $k \to \infty$, the ratio approaches 0. However, σ_L^* of the network is 1 by Lemma 8, since the star interference graph contains no cycles.

To improve the lower bound, we extend our weight assignment approach to subsets of L. For a set $L' \subseteq L$, let $||[s^i]_{L'}||_1$ represent the 1-norm of the vector $[s^i]_{L'}$, which is s^i restricted to the set L'.

Lemma 13. *For a non-empty set $L \subseteq E$, suppose the maximal schedules are $M_L = (s^1, s^2, s^3, \dots, s^t)$ where s^i is a vector corresponding to the i^{th} maximal schedule with respect to L. Then,*

$$\sigma_L^* \geq \max_{L' \subseteq L} \frac{\min_i ||[s^i]_{L'}||_1}{\max_i ||[s^i]_{L'}||_1}. \tag{2–32}$$

Proof. Assign $x_j = 1$ for all nodes $j \in L'$ and $x_j = 0$ otherwise. □

Regardless of the weight assignment scheme on nodes, enumerating all the maximal schedules is intractable for large sets. As a result, it is difficult to find the schedules with the maximum or the minimum weights. This observation motivates us to find a simpler way to lower-bound σ_L^*. The following lemma states that it is possible to derive a lower bound for σ_L^* using the interference degree. The interference degree of a node is defined as the maximum number of nodes that can be scheduled simultaneously in the node's single-hop neighborhood, where the hop count is measured in the interference graph [18]. Since the interference graph here is restricted to L, the interference degree of a node $i \in L$ is denoted by $d_L(i)$.

Lemma 14. *Let* $d^* = \min_{i \in L} d_L(i)$*. Then, we have* $\sigma_L^* \geq 1/d^*$*.*

Proof. Let $l^* \in \text{argmin}_{i \in L} d_L(i)$. Let $L' \subseteq L$ be the set of neighbors of l^* in the interference graph (restricted to L), i.e., the largest subset such that each $l \in L'$ interferes with l^*. Here, we use the convention that $l^* \in L'$.

The maximum number of nodes in L' that can be activated simultaneously is d^*. There is at least one node that must be activated in L' for any maximal schedule. By applying Lemma 13, we have $\sigma_L^* \geq 1/d^*$. □

Remark: Lemma 17 can be used to derive Proposition 3 in [18].

2.4.2 Estimating σ-Local Pooling Factor for Link

By Lemma 7, the σ-local pooling factor for a link is equal to the σ-local pooling factor for its limiting set; by Lemma 9, there is a connected limiting set in the interference graph. In this part, we concentrate on deriving lower bounds for link performance based on these facts.

We present an algorithm (Algorithm 1), which provides a lower bound for the local pooling factor for each link based on its interference degree. Unlike the algorithm in [18], which aims at finding a single performance bound for the whole graph, our algorithm finds a separate performance bound for each link.

Theorem 2.3. *Given* $G = (V, E, \mathcal{I})$*, let* $(\sigma_l)_{l \in E}$ *be the values returned by Algorithm 1. Then,* $\sigma_l^* \geq \sigma_l$ *for each* l*.*

Algorithm 1 σ-Local Pooling for Each Link

INPUT: A graph $G = (V, E, \mathcal{I})$
OUTPUT: σ-local pooling factors for all links, $(\sigma_l)_{l \in E}$
Initialization: $L_1 \leftarrow E$, $d \leftarrow 1$
for all $1 \le i \le |E|$ **do**
  Choose a link l from L_i with the minimum interference degree restricted to L_i.
  if $d_{L_i}(l) \le d$ **then**
    $\sigma_l = 1/d$
    $L_{i+1} \leftarrow L_i \setminus l$
    $i \leftarrow i + 1$
  else
    $d \leftarrow d + 1$
  end if
end for
Return $(\sigma_l)_{l \in E}$.

Proof. Consider the first round where L_1 contains all links. Suppose there is a link l satisfying $d_{L_1}(l) \le d$. Then, if the link has a limiting set L, we have $d_L(l) \le d$ since $L \subseteq L_1$. By applying Lemma 7 and Lemma 14, $\sigma_l^* = \sigma_L^* \ge 1/d_L(l) \ge 1/d = \sigma_l$.

For any later round, when a link l is chosen to be removed, there are two possibilities: (1) Link l's limiting set L contains a previously removed link; or (2) link l's limiting set L does not contain any previously removed link. If case (1) is true, we can assume L contains a previous removed link k. By Corollary 1, $\sigma_l^* = \sigma_L^* \ge \sigma_k^*$. Due to the monotonicity of d, $d_k \le d_l$ where d_k is the d value when k is removed and d_l is the d value when l is removed. Hence, $\sigma_l^* = \sigma_L^* \ge \sigma_k^* \ge 1/d_k \ge 1/d_l$. Since $\sigma_l = 1/d_l$, we have $\sigma_l^* \ge \sigma_l$. In case (2), since the limiting set L for l does not contain any previously removed links, we must have $L \subseteq L_i$, and hence, $d_L(l) \le d_{L_i}(l)$. Thus, $\sigma_l^* = \sigma_L^* \ge 1/d_{L_i}(l) \ge 1/d_l = \sigma_l$. Therefore, in either case, $\sigma_l^* \ge \sigma_l$ holds. $\square$

Remark: Algorithm 1 can be further compared with the similar algorithm in [18]. Both algorithms generate a sequence of links $l_1, l_2, \ldots, l_{|E|}$ as they progress (but not the same sequence in general). Our algorithm produces a sequence of values $\{d_{l_i}\}_{1 \le i \le |E|}$, where d_{l_i} is the d value before link l_i is removed from the graph. The algorithm in [18] also implicitly generates a sequence of values $\{\hat{d}_{l_i}\}_{1 \le i \le |E|}$. In the end, our algorithm returns $(\sigma_{l_i})_{1\leq i\leq|E|}$, where each σ_{l_i} is a lower bound of $\sigma^*_{l_i}$ and $\sigma_{l_i} = 1/d_{l_i}$ for each l_i. The algorithm in [18] returns $1/d_e$ as a lower bound of $\sigma^*(G)$, where $d_e = \max_i \hat{d}_{l_i}$. The following fact can be shown.

Lemma 15. *Every σ_l returned by Algorithm 1 is greater than or equal to the returned value $1/d_e$ by the algorithm of [18].*

Proof. The proof is by contradiction. Suppose there exists a link p with $\sigma_p < 1/d_e$. According to Algorithm 1, $\sigma_p = 1/d_p$, where d_p is the d value before p is removed from the graph. Then, $1/d_p = \sigma_p < 1/d_e$, which implies that $d_p > d_e$ or $d_p - 1 \geq d_e$. In Algorithm 1, d is increased by 1 only when in some round i, there is no link l in L_i such that $d_{L_i}(l) \leq d$. Hence, there exists a round i such that every link l in L_i satisfies that $d_{L_i}(l) > d_p - 1 \geq d_e$.

Now, consider the algorithm of [18] and consider any sequence of link removals in that algorithm. Suppose k is the first link removed from set L_i by that algorithm. Assume that just before removing k, the remaining set of links is $\tilde{L}$. Then $L_i \subseteq \tilde{L}$. This implies that $d_{L_i}(k) \leq d_{\tilde{L}}(k) \leq d_e$. This leads to a contradiction, because $d_{L_i}(k) > d_p - 1 \geq d_e$. Therefore, the lemma holds. $\square$

Example: Consider the interference graph in Fig. 2-3, which contains a 6-cycle connected with a tree. For this graph, Algorithm 1 works as follows.

1. Initially, $d = 1$ and $L_1 = E$.

2. Every leaf node, say l, satisfies $d_{L_1}(l) \leq d = 1$. Pick one leaf node l_1 and assign $\sigma_{l_1} = 1/d = 1$. Then, remove l_1 from L_1 to get L_2.

3. Continue to remove leaf nodes from the tree until only the cycle is left. Suppose this takes $k-1$ steps. Then, the node set of the cycle is L_k. For a cycle, we cannot find any node l with $d_{L_k}(l) \leq d = 1$. Hence, we increase d to 2.

4. Find node l_k in the cycle with $d_{L_k}(l_k) \leq d = 2$. Assign $\sigma_{l_k} = 1/d = 1/2$. Then, remove l_k from L_k to get L_{k+1}.

5. From now on, we can always find a node l_i from the remaining graph L_i, satisfying $d_{L_i}(l_i) \leq d = 2$. We obtain $\sigma_{l_i} = 1/2$ for all remaining nodes.

For the same graph, the algorithm of [18] can obtain a lower bound for $\sigma^*(G)$ no greater than $1/2$ for the following reason. Since there is a six-cycle in the graph, we know $\sigma^*(G) \leq 2/3$; but, $1/2$ is the best value the algorithm can obtain other than 1. Let $\sigma(G)$ denote the lower bound returned by the algorithm ($\sigma(G) = 1/d_e$). Comparing the two algorithms, we see that $\sigma_l \geq \sigma(G)$ for all l. Moreover, our algorithm obtains the exact σ_l^* for every node l on the tree; the algorithm of [18] underestimates the performance of those nodes on the tree by using a lower bound of σ_p^* for a node p in the cycle. Imagine the tree has many nodes. We see that a small part of the network can limit the performance characterization of the entire network.

2.5 Optimality of Equal Weight Assignment for Set σ-Local Pooling in Special Classes of Graphs

This section presents some theoretical results about σ-local pooling that can be useful for further research on this subject.

2.5.1 Optimality of Equal Weight Assignment for Set σ-Local Pooling in Special Classes of Graphs

In this section, we study the set σ-local pooling factor for some special classes of graphs. We use the word *schedule* to mean a *maximal* independent set of a graph.

Remark: In this section, unless mentioned other wise, all the graphs are understood as interference graphs or induced subgraphs of interference graphs for networks.

Given an interference graph (or induced subgraph) $G = (V, E)$, the set σ-local pooling factor is really a property of the graph itself. For clarity and simplicity, we will denote this factor by σ_G^* in this section (which should not be confused with $\sigma^*(G)$ introduced earlier). Let $w : V \rightarrow \mathbb{R}^+$ be a weight assignment on the nodes. For a schedule s, we use $\chi_w(s)$ to denote the total weight of the schedule s under w.

2.5.1.1 Weight-balanced graphs

Definition 6. *In a graph $G = (V, E)$, let w be an arbitrary weight assignment on the nodes. We say the graph is* weight-balanced*, if the following inequalities hold:*

$$\bar{w}|M| \leq \chi_w(M) \tag{2–33}$$

$$\chi_w(m) \leq \bar{w}|m|, \tag{2–34}$$

where m and M are schedules with the minimum and maximum weight, respectively, and $\bar{w}$ is the average weight of all the nodes in G.

Lemma 16. *A cycle graph is weight-balanced.*

Proof. Let $G = (V, E)$ be a cycle graph. Let us index the nodes sequentially around the cycle from 1 to $|V|$. Given a schedule s, we let s^i denote the i^{th} rotation of s, for $0 \leq i \leq |V| - 1$. That is, if we denote a schedule by the set of selected nodes, then $s^i = \{j | j = (k + i) \bmod |V|, \text{ where } k \in s\}$ (assuming we equate node 0 with node $|V|$). Note that $s^0 = s$.

Note that each rotation s^i of s is a schedule as well. Let $w = (w_1, w_2, \dots, w_{|V|})$ be an arbitrary weight vector (weight assignment). By summing the weights of all these rotations, we get,

$$\sum_{i=0}^{|V|-1} \chi_w(s^i) = \sum_{i=1}^{|V|} w_i |s|. \tag{2–35}$$

Let M be a schedule with the maximum weight. It follows that $\chi_w(s^i) \leq \chi_w(M)$. Thus, we have the following.

$$\sum_{i=1}^{|V|} w_i |s| \leq |V| \chi_w(M)$$

$$\bar{w}|s| \leq \chi_w(M),$$

where $\bar{w}$ is the average node weight. This proves (2–33). The proof for (2–34) follows a similar argument. $\square$

We will consider computing the set σ-local pooling factor σ_G^* according to (2–29), which we re-write next using the new notations. Let M_G be the set of schedules for the graph G.

$$\sigma_G^* = \max_{x \geq 0} \frac{\min_{s \in M_G} \chi_x(s)}{\max_{s \in M_G} \chi_x(s)}. \tag{2–36}$$

The following lemma shows a special property if the graph G is weight-balanced.

Lemma 17. *If the graph G is weight-balanced, then an equal weight assignment is optimal with respect to the optimization problem (2–36).*

Proof. We consider two cases. First, consider the case where $\sigma_G^* = 1$. Suppose w is an optimal weight assignment. The inequalities (2–33) and (2–34) hold under w. Since $\sigma_G^* = 1$, all schedules have the same weight. Hence, if m and M are schedules with the minimum and maximum weight, respectively, we get $\chi_w(M) = \chi_w(m)$. Then, $\bar{w}|M| \leq \bar{w}|m|$ must hold. Using this inequality and the fact that any schedule can be considered as either a maximum-weight schedule or a minimum-weight schedule, we conclude that all schedules must have the same cardinality. Therefore, assigning equal weight to every node achieves $\sigma_G^* = 1$.

Consider the second case where $\sigma_G^* < 1$. Suppose there are n nodes in G and they are indexed from 1 to n. Take an arbitrary optimal weight assignment and suppose the average of the node weights is c, where c is some constant. For any weight vector $x = \{x_1, x_2, ..., x_n\}$, let $\phi(x) = \max_{i=1}^n x_i - \min_{i=1}^n x_i$, and let the average of x be denoted by $\bar{x}$.

Among all the optimal weight assignments with the average node weight equal to c, we pick one that minimizes ϕ and denote this assignment by x^*. That is, $x^* \in \arg\min\{\phi(x) \mid x \text{ is optimal and } \bar{x} = c\}$, with ties broken arbitrarily.

Notice that if $\phi(x^*) = 0$, the lemma holds. Next, we assume $\phi(x^*) > 0$ and we will show this leads to a contradiction. In particular, we will show it is always possible to construct a new optimal weight vector y with the same average node weight, c, such that $\phi(x^*) > \phi(y) > 0$.

We define $y = \{y_i | y_i = x_i^* - \epsilon(x_i^* - c)\}$. Note that, for any ϵ, $\bar{y} = \frac{1}{n}\sum y_i = \frac{1}{n}\sum x_i^* = c$. Also, there exists $\epsilon_1 > 0$ such that for all ϵ on $(0, \epsilon_1]$, we have $\phi(x^*) > \phi(y) > 0$. We will show that we can choose small enough ϵ on $(0, \epsilon_1]$ such that y is also an optimal weight assignment.

Since $\sigma_G^* < 1$, the maximum schedule weight is strictly greater than the minimum schedule weight under x^*. As a result, there is a gap between the maximum schedule weight and the second largest weight of all schedule weights. Furthermore, under the weight vector y, the weight of each schedule is a linear function of ϵ. Hence, if we choose small enough ϵ on $(0, \epsilon_1]$, we can make sure that some maximum-weight schedule under the weight assignment x^* remains a maximum-weight schedule under y. More precisely, there exists ϵ_2 on $(0, \epsilon_1]$ such that, for all $\epsilon \in (0, \epsilon_2]$, there exists a schedule M (independent of ϵ) that has the maximum weight under both x^* and y.

For a weight-balanced graph, we next show the maximum schedule weight is not increased when the weight assignment changes from x^* to y.

$$\begin{aligned}\chi_y(M) &= \sum_{i \in M} y_i \\ &= \sum_{i \in M}(x_i^* - \epsilon(x_i^* - c)) \\ &= \chi_{x^*}(M) - \epsilon(\chi_{x^*}(M) - |M|c).\end{aligned}$$

By the definition of a weight-balanced graph, $\chi_{x^*}(M) - |M|c \geq 0$. Thus, we conclude $\chi_y(M) \leq \chi_{x^*}(M)$.

Similarly, there exists ϵ_3 on $(0, \epsilon_1]$ such that, for all $\epsilon \in (0, \epsilon_3]$, there exists a schedule m (independent of ϵ) that has the minimum weight under both x^* and y; and furthermore, $\chi_y(m) \geq \chi_{x^*}(m)$. By choosing ϵ on $(0, \min(\epsilon_2, \epsilon_3)]$, we get $\sigma_G^* = \chi_{x^*}(m)/\chi_{x^*}(M) \leq \chi_y(m)/\chi_y(M)$. Hence, y is also an optimal weight assignment for the problem in (2–36). Considering $\phi(x^*) > \phi(y) > 0$ and the assumption that $x^* \in \arg\min\{\phi(x) \mid x \text{ is optimal and } \bar{x} = c\}$, we have reached a contradiction. $\square$

Lemma 17 implies that the inequality in Lemma 13 can be changed to equality in a weight-balanced graph. The next corollary follows immediately from Lemma 16 and 17.

Corollary 3. *If the graph G is a cycle, then the optimal σ_G^* is achieved by assigning identical weights to the nodes.*

2.5.1.2 Vertex-transitive graphs

We next consider another class of graphs for which an equal weight assignment is optimal for (2–36). We first need to introduce some definitions (see [45]). We consider undirected graphs with no loops and no more than one edge between any two different nodes, i.e., the *simple graphs*.

Definition 7. *An isomorphism from a graph $G = (V_G, E_G)$ to a graph $H = (V_H, E_H)$ is a bijection $f : V_G \to V_H$ such that $(u, v) \in E_G$ if and only if $(f(u), f(v)) \in E_H$.*

Definition 8. *An automorphism of a graph G is an isomorphism from G to G.*

Definition 9. *A graph $G = (V, E)$ is vertex-transitive if for every pair $u, v \in V$ there is an automorphism that maps u to v.*

Lemma 18. *If a graph G is vertex-transitive, then the optimization problem (2–36) is achieved by an equal weight assignment on the nodes.*

Proof. As in the proof of Lemma 17, we assume the nodes in G are indexed from 1 to n; for any weight vector x, we define $\phi(x) = \max_{i=1}^n x_i - \min_{i=1}^n x_i$. Note that $\phi(x) = 0$ if and only if the node weights are all identical. Let $\bar{x}$ denote the average node weight.

We restrict our attention to those weight assignments whose average node weight is equal to c, where c is some constant, e.g., $c = 1$. This “normalization” is without loss of generality. Among all the optimal weight assignments that have the average node weight equal to c, we pick one that minimizes ϕ and denote this assignment by x^*. That is, $x^* \in \arg\min\{\phi(x) \mid x \text{ is optimal and } \bar{x} = c\}$. Suppose $\phi(x^*) > 0$.

Let $a = \min_i x_i^*$ and $b = \max_i x_i^*$. Let $S_a = \{i | x_i^* = a\}$ and $S_b = \{i | x_i^* = b\}$. We observe that $S_a, S_b \neq \emptyset$ and $\phi(x^*) = b - a > 0$. Now, we pick a node $u \in S_b$ and a node $v \in S_a$. By the definition of vertex-transitivity, there is an automorphism f that maps u to v. We construct a new weight vector x' by $x'_i = 1/2(x_i^* + x_{f(i)}^*)$ for all i. Note that the weight vector y with $y_i = x_{f(i)}^*$ for all i is also optimal. Since x' is a convex combination of two optimal solutions, x' is also optimal. Also, the new weight vector x' has the same average weight as x^*. Notice that $\max_i x'_i \leq b$, $\min_i x'_i \geq a$ and $\phi(x') \leq \phi(x)$. Since $x_u^* > x_v^*$, it follows that $x'_v > x_v^* = a$. Hence, the set of nodes whose weight is equal to a has fewer elements under x', and we again call this set S_a. If S_a is empty, then $\phi(x') < \phi(x^*)$, which is a contradiction. For a similar reason, the set of nodes whose weight is equal to b has fewer elements under x' and we call this set S_b. If S_b is empty, then $\phi(x') < \phi(x^*)$, which is a contradiction.

If neither S_a nor S_b is empty, we can repeat the above procedure to construct a new weight vector from x' and update S_a and S_b. Eventually, either S_a or S_b first becomes empty under some new weight vector, say $\hat{x}$. Then $\phi(\hat{x}) < \phi(x^*)$, which is a contradiction. $\square$

Remark: A similar result was found independently in [4] [3]. It is easy to show that cycles are vertex-transitive. Therefore, Lemma 18 also implies Corollary 3.

2.6 Additional Related Work

In this section, we cover some additional related work. The references cited by [18] are mostly related, but are not all repeated here. In [25], Lin et al. provided a distributed algorithm using schedules that correspond to *maximal* matchings (in the interference graph) for the 1-hop interference model. In each such schedule, no more links can be added to it without violating the interference constraint. They showed that the algorithm can achieve a stability region $\Lambda/2$ under the 1-hop model. For more general interference models, they showed that if one can find an approximation algorithm with the approximation ratio γ for a maximum weighted independent set subproblem, then one can achieve a stability region $\gamma\Lambda$.

In [2], the authors considered the 2-hop interference model that can successfully capture the IEEE 802.11 network and provided an algorithm to find an upper-bound for the network capacity. In [36] [35], the 2-hop interference is generalized to k-hop interference and the problem of finding a maximum-weight schedule was shown to be NP-Hard, for $k \geq 2$. The authors of [9] [38] provided lower bounds on the performance of the *maximal matching* algorithms for the cases of arbitrary and geometric graphs. More specifically, for geometric graphs, [9] showed that the efficiency ratio $\gamma^*(G) \geq 1/8$ in the 2-hop interference model; [35] showed $\gamma^*(G) \geq 1/49$ in the k-hop interference model where $k \geq 2$.

In addition to [17], several other papers also introduced algorithms of local scheduling that have performance guarantee. Lin and Rasool [24] introduced random scheduling schemes under the 1-hop and 2-hop interference models. There is a slight loss of efficiency in their schemes compared with what is achievable by the so-called distributed greedy scheduling algorithm by Wu et al. [46], which is also a local greedy algorithm. Joo and Shroff [19] and Gupta et al. [15] both discovered related random scheduling schemes that improve upon Lin and Rasool's schemes.

2.7 Summary

In this chapter, we provide a refined framework on performance characterization of the LQF policy, based on the idea of local pooling introduced in [12] [18]. In particular, we introduce the concept of link σ-local pooling, which allows heterogeneous characterization of individual link performance, as opposed to treating all links the same. We define the $\Sigma^*(G)$ diagonal matrix, which contains the link σ-local pooling factors in the diagonal entries, as a generalization of the network σ-local pooling, $\sigma^*(G)$, in [18]. The matrix $\Sigma^*(G)$ provides a refined performance characterization for LQF. We show that our performance characterization captures a larger region of stability than previous results.

We then introduce a set of theory that helps to apply the new idea of link σ-local pooling. The core of this theory involves the concepts of σ-local pooling for a set of links and the limiting set for a link. We show how these concepts are related to link σ-local pooling, and how to calculate or bound both set and link σ-local pooling factors. Based on the developed theory, we derive new estimation methods for set and link σ-local pooling factors.

There are still open issues that may be addressed by further research. The following are three examples. First, the computational complexity of calculating $\Sigma^*(G)$ or σ_L^* is still unknown. Second, in light of the newly discovered fact that Joo's local greedy scheduling also achieves the stability region $\sigma^*(G)\Lambda^o$ [17], it would be interesting to investigate whether the enlarged stability region of LQF, $\Sigma^*(G)\Lambda^o$, can be preserved by similar local greedy algorithms. Third, there are other, possibly nonlinear, transformations of the capacity region Λ to $\phi(\Lambda)$ where LQF stabilizes the network. Further investigation on these transformations may have important theoretical and practical values.

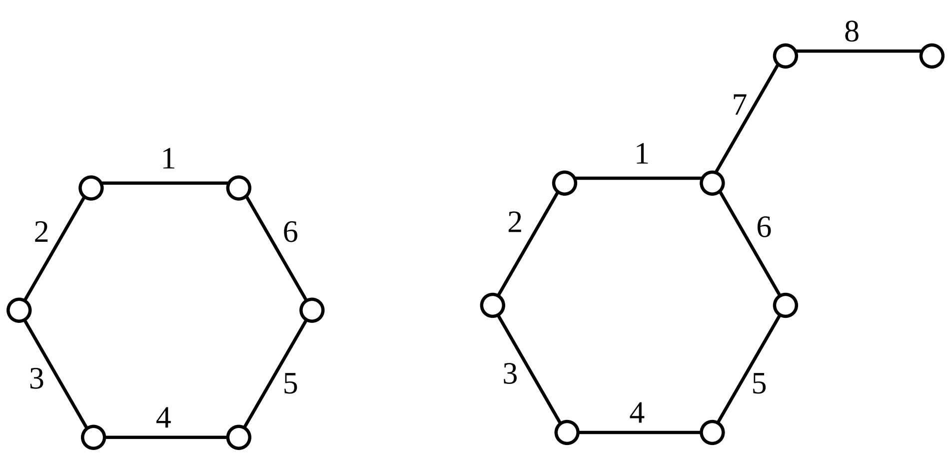

Figure 2-1. Two networks with different performance behavior from the link perspective. Left: uniform; Right: heterogeneous.

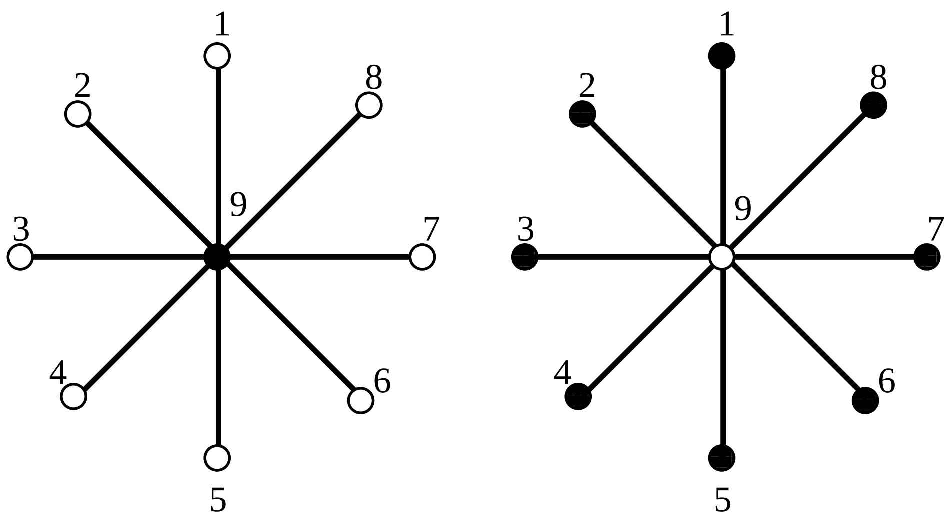

Figure 2-2. A 9-node interference graph under equal node-weight assignment.
Left: the minimum-weight schedule; Right: the maximum-weight schedules.

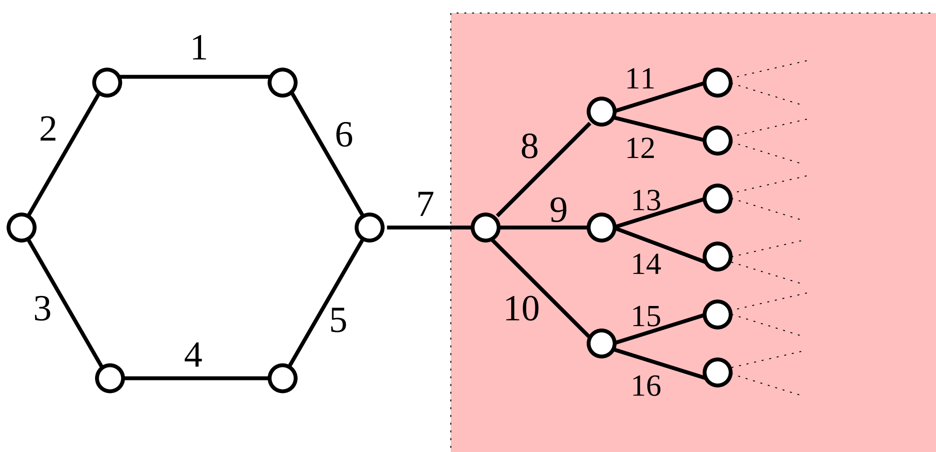

Figure 2-3. An interference graph with a six cycle connected with a tree.

CHAPTER 3
A NON-CONVEX PERFORMANCE GUARANTEE UNDER LONGEST QUEUE FIRST SCHEDULE IN WIRELESS NETWORKS

In this chapter, we further study the performance guarantee of the LQF algorithm. It is known that the LQF algorithm achieves the full capacity region, Λ, when the interference graph satisfies the so-called local pooling condition. For a general graph G, LQF achieves (i.e., stabilizes) a part of the capacity region, $\sigma^*(G)\Lambda$, where $\sigma^*(G)$ is the overall local pooling factor of the interference graph G and $\sigma^*(G) \leq 1$. In chapter 2, it has been shown that LQF achieves a larger rate region, $\Sigma^*(G)\Lambda$, where $\Sigma^*(G)$ is a diagonal matrix. The contribution of this chapter is to describe three new achievable rate regions, which are larger than the previously-known regions. In particular, the new regions include all the extreme points of the capacity region and are not convex in general. We also discover a counter-intuitive phenomenon in which increasing the arrival rate may sometime help to stabilize the network. This phenomenon can be well explained using the theory developed in this chapter.

The chapter is organized as follows. In Section 3.1, we specify the models and notations. In Sections 3.2 and 3.3, we introduce the Ω and Δ (Δ_C and Δ_R) regions, respectively. In Section 3.4, we introduce the fractional coloring and related problems that are relevant to the study of the stability regions. In Section 3.5, we give some simulation results to confirm aspects of the theory. The summarization is given in Section 3.6.

3.1 Preliminaries and Network Model

In our model, a wireless network is represented by an undirected interference (or conflict) graph $G = (V, E)$, where the node set V represents the set of physical, wireless links in the network and the edge set E represents the interference relation among the physical links. Two nodes in G are connected with an edge whenever the physical links they represent interfere with each other.[1] We assume the node set V is arbitrarily indexed from 1 to $|V|$, and hence, V can be written as $V = \{1, \dots, |V|\}$.

Given a subset of nodes $S \subseteq V$, we denote $G_S = (S, L)$ to be the subgraph of G induced by the nodes in S. In other words, an edge (u, v) belongs to L if and only if $u, v \in S$ and $(u, v) \in E$.

We assume a time-slotted system. The capacity of each wireless link is normalized to 1 per time slot. There is a queue associated with each wireless link at the transmitter. We assume single-hop traffic. Traffic arrives at the transmitter side of a link, joining the queue and waiting for transmission; after transmission, it leaves the network. We assume i.i.d. and mutually independent arrival processes to the queues. It is easy to see that, under the LQF schedule with either deterministic or typical random tie-breaking rules, the joint queue process is Markovian. By stability, we mean the Markov process is positive recurrent[2] .

A schedule is denoted by a $|V|$-dimensional 0-1 vector, where a value 1 in an entry indicates the corresponding link is active and 0 otherwise. A schedule is feasible if and only if the links that are active do not interfere with each other. A feasible schedule is said to be *maximal* if no additional links can be activated without violating the interference constraints. Therefore, every feasible schedule is an independent set of G and every maximal schedule is a maximal independent set of G.

For the graph $G = (V, E)$, let M_V denote the set of all the maximal schedules and let $Co(M_V)$ denote the convex hull of all the maximal schedules. When relevant, we also consider M_V to be the matrix whose columns are all the maximal schedules, with arbitrary indexing of the schedules. Similarly, for a node-induced subgraph $G_S = (S, L)$,

[1] All the graphs in this paper are interference graphs, unless specified otherwise.

[2] Without loss of generality, we assume the Markov Chain is irreducible. See [41] for general cases.

let M_S be the set (or matrix) representing all the maximal schedules of G_S and let $Co(M_S)$ be the convex hull of all the maximal schedules in M_S.

The capacity region Λ of a network is defined as the set of arrival rate vectors that are supportable by time sharing of the feasible schedules. Equivalently,

$$\Lambda = \{\lambda \mid 0 \leq \lambda \leq \mu \text{ for some } \mu \in Co(M_V)\}. \quad (3–1)$$

For a non-empty subset of nodes $S \subseteq V$, the capacity region is defined analogously by replacing $Co(M_V)$ with $Co(M_S)$ in (3–1) and is denoted by Λ_S. In the above, $\lambda \leq \mu$ means that vector λ is component-wise less than or equal to vector μ. The *interior* of the capacity region can be written as

$$\Lambda^o = \{\lambda \mid 0 \leq \lambda < \mu \text{ for some } \mu \in Co(M_V)\}. \quad (3–2)$$

The interior of the capacity region thus defined can be stabilized by the MWIS schedule and any rate vector outside the capacity region cannot be stabilized by any schedule [41].

Given a $|V|$-dimensional vector λ, the $|S|$-dimensional vector $[\lambda]_S$ represents the restriction of λ to the set $S \subseteq V$. That is, $[\lambda]_S$ contains only those components of λ which correspond to the nodes in S.

For a vector μ defined for a node set, let μ_l or $\mu(l)$ denote the component associated with $l \in V$. Note that, if $\mu \in \mathbb{R}_+^{|V|}$, then the notation indicates the lth component of μ. However, if $\mu \in \mathbb{R}_+^{|S|}$ for some non-empty $S \subseteq V$, then for $l \in S$, μ_l or $\mu(l)$ is not necessarily the lth component of μ. If μ is any other type of vector, μ_i denotes the ith component of μ. We use e to represent the vector $(1, 1, \ldots, 1)'$. The dimension of the vector e depends on the context.

The capacity region for the whole graph and the capacity region for the subset $S \subseteq V$ has the following relationship.

Lemma 19. *An arrival rate vector $\lambda \in \Lambda$ if and only if for all non-empty $S \subseteq V$, $[\lambda]_S \in \Lambda_S$. Likewise, $\lambda \in \Lambda^o$ if and only if for all non-empty $S \subseteq V$, $[\lambda]_S \in \Lambda_S^o$.*

Proof. Suppose $\lambda \in \Lambda$. Then, $\lambda \leq \mu$ for some $\mu \in Co(M_V)$. It is easy to see that, for any non-empty subset $S \subseteq V$, there must exist a vector $\nu \in Co(M_S)$ such that $[\mu]_S \leq \nu$. Then, $[\lambda]_S \leq \nu$. Hence, $[\lambda]_S \in \Lambda_S$. The other direction is true by taking $S = V$. The last statement of the lemma can be proved similarly. □

Throughout, in the statements about rate regions that involve topological concepts such as open/close sets and the interior of a set, the space is assumed to be the set of non-negative real vectors, i.e., $\mathbb{R}_+^{|V|}$. Also, in the set-complement operation for any rate region, the whole set is understood to be the non-negative real vectors. For a set $Y \subseteq \mathbb{R}_+^{|V|}$, we let Y^o and Y^c denote the interior and complement of Y, respectively.

In the LQF schedule, the links with longer queues are activated at a higher priority than those with shorter queues, subject to the interference constraints. The following may be considered as a reference implementation of this schedule. First, one of the links with the longest queue is selected to be in the schedule; ties are broken with either an arbitrary deterministic rule or randomly. All links with which the selected link interferes are removed from further consideration. Then, the same selection process repeats over the remaining links yet to be considered until no links remain to be considered.

Remark: The following is the key mathematical property about LQF that is used throughout. Suppose, at time t, a non-empty set $S \subseteq V$ dominates $V - S$ in the sense that, for any $i \in S$ and any $j \in V - S$, the queue size of i is greater than that of j. Then, the schedule used at t must be maximal when restricted to S (i.e., with respect to G_S).

3.2 Stability Region Ω under LQF

In this section, we introduce a notion of strictly dominating vectors and construct a region denoted by Ω based on this notion. The Ω region is larger than $\sigma^*(G)\Lambda$ and $\Sigma^*(G)\Lambda$, which have previously been shown to be stabilizable by the LQF policy. Unlike those previously-discovered regions of stability, the Ω region includes all the extreme points of Λ and it is not convex in general.

3.2.1 Review of Set, Link and Overall σ-local Pooling

Set σ-local pooling has been studied in [23]. It has many interesting properties and is related to (overall) σ-local pooling defined in [18].

Definition 10. *Given a non-empty set of nodes $S \subseteq V$, the* set σ-local pooling factor for S, *denoted by σ_S^*, is given by*

$$\sigma_S^* = \sup\{\sigma \mid \sigma\mu \not> \nu, \textit{ for all } \mu, \nu \in Co(M_S)\ \} \tag{3–3}$$

$$= \inf\{\sigma \mid \sigma\mu > \nu, \textit{ for some } \mu, \nu \in Co(M_S)\ \}. \tag{3–4}$$

It has been shown that the set σ-local pooling factor is equal to the optimal value of the following problem.

$$\sigma_S^* = \min_{\sigma,\mu,\nu} \sigma, \quad \text{subject to } \sigma\mu \geq \nu,\ \mu, \nu \in Co(M_S). \tag{3–5}$$

The link σ-local pooling factor is defined as follows.

Definition 11. *The* local pooling factor of a link $l \in V$, *denoted by σ_l^*, is given by*

$$\sigma_l^* = \sup\{\sigma|\sigma\mu \not> \nu \textit{ for all } S \subseteq V \textit{ such that } l \in S, \textit{ and all } \mu, \nu \in Co(M_S)\} \tag{3–6}$$

$$= \inf\{\sigma|\sigma\mu > \nu \textit{ for some } S \subseteq V \textit{ such that } l \in S, \textit{ and some } \mu, \nu \in Co(M_S)\}. \tag{3–7}$$

Comparing the definitions of σ_S^* and σ_l^*, we have

$$\sigma_l^* = \min_{\{S\subseteq V \mid l\in S\}} \sigma_S^*. \tag{3–8}$$

The overall σ-local pooling factor of the graph $G = (V, E)$ is

$$\sigma^*(G) = \min_{l\in V} \sigma_l^*.$$

Let the diagonal matrix $\Sigma^*(G)$ be defined by $\Sigma^*(G) = \text{diag}\,(\sigma_l^*)_{l\in V}$. It has been shown that $\sigma^*(G)\Lambda$ and $\Sigma^*(G)\Lambda$ are both regions of stability under LQF [18] [23], with the latter containing the former.

3.2.2 Strictly Dominating Vectors and Ω Region

We first discuss some intuition that leads to the construction of the Ω region. When the network is unstable, a typical situation is that the size of the longest queues has an overall trend of increase, if one ignores the short-time fluctuations. This would not have occurred if, for any subset $S \subseteq V$, the arrival rate is strictly less than the service rate at each node in S. Here, we imagine S is the set of nodes with the longest queues for an extended period of time. Then, over that period of time, the schedule on each time slot must be maximal when restricted to S and, by time sharing of such maximal schedules, the (average) service rate must be in $Co(M_S)$. The discussion motivates us to define the notion of strictly dominating vectors for a subset of the nodes.

Definition 12. *Given a non-empty node set $S \subseteq V$, a vector $\lambda \in \mathbb{R}_+^{|V|}$ is a strictly dominating vector of S if $[\lambda]_S > \nu$ for* **some** *$\nu \in Co(M_S)$. The region composed with all the strictly dominating vectors of S is called the strictly dominating region of S and is denoted by Π_S. That is,*

$$\Pi_S = \{\lambda \in \mathbb{R}_+^{|V|} \mid [\lambda]_S > \nu,\ \textit{for some}\ \nu \in Co(M_S)\}.$$

For convenience, if $S = \emptyset$, we assume $\Pi_S = \emptyset$.

We are often interested in the complement of Π_S:

$$\begin{aligned}\Pi_S^c &= \{\lambda \in \mathbb{R}_+^{|V|} \mid [\lambda]_S \not> \nu,\ \text{for all}\ \nu \in Co(M_S)\} \\ &= \{\lambda \in \mathbb{R}_+^{|V|} \mid \text{for every}\ \nu \in Co(M_S),\ \text{there exists}\ l \in S\ \text{such that}\ \lambda(l) \le \nu(l)\}.\end{aligned}$$

Definition 13. *The Ω region is defined by*

$$\Omega = \bigcap_{S\subseteq V} \Pi_S^c.$$

Remark. A vector λ is outside Ω if and only if $\lambda \in \Pi_S$ for some non-empty node set S. Also, when restricted to the components corresponding to the nodes in S, Π_S is an open set (it is a union of open sets). Hence, Ω is a closed set. It can also be helpful to think $\Omega = (\bigcup_{S \subseteq V} \Pi_S)^c$.

Lemma 20. *Suppose an arrival rate vector λ satisfies $\lambda \in \Omega^o$. Then, for any non-empty subset $S \subseteq V$ and any $\nu \in Co(M_S)$, there exists $l \in S$ such that $\lambda(l) + \epsilon_o < \nu(l)$, where $\epsilon_o > 0$ is a constant independent of S, ν and l.*

Proof. Since $\lambda \in \Omega^o$, we have $\lambda + \hat{\epsilon} e \in \Omega$ for some small enough $\hat{\epsilon} > 0$. Suppose the conclusion of the lemma is not true. That is, suppose for any $\epsilon > 0$, there exists a non-empty subset $S \subseteq V$ and $\nu \in Co(M_S)$ such that $\lambda(l) + \epsilon \geq \nu(l)$ for all $l \in S$. We can choose ϵ satisfying $0 < \epsilon < \hat{\epsilon}$. Then, $[\lambda + \hat{\epsilon} e]_S > [\lambda + \epsilon e]_S \geq \nu$. Hence, $\lambda + \hat{\epsilon} e \in \Pi_S$, which implies $\lambda + \hat{\epsilon} e \notin \Omega$ by Definition 13, leading to a contradiction. $\square$

3.2.3 Performance Guarantee of LQF in Ω Region

Theorem 3.1. *If an arrival rate vector λ satisfies $\lambda \in \Omega^o$, then, the network is stable under the LQF policy.*

The full proof requires replicating most of the arguments in [12]. In the following, we only highlight the part of the argument that needs modification.

Sketch of Proof. Consider the fluid limit of the queue processes, denoted by $\{q_l(t)\}_{t \geq 0}$ for all $l \in V$. For a fixed (and regular) time instance t, let S be the set of those longest queues whose time derivatives at t, $\dot{q}_l(t)$, are the largest under a given LQF policy instance. The queues in S will remain the longest with identical length in the next infinitesimally small time interval.

The service rate vector, when restricted to S, must belong to the set $Co(M_S)$. Roughly, this is because S contains all the queues that are the longest and remain the longest in the near future, and hence, as remarked earlier, every LQF schedule being used must be a maximal schedule when restricted to S.

Now imagine ν is the service rate vector for the nodes in S at time t. Since $\lambda \in \Omega^o$, by Lemma 20, there exists a link $l \in S$ such that $\lambda(l) + \epsilon_o < \nu(l)$ for some constant $\epsilon_o > 0$. Then, $\nu(l) - \lambda(l) > \epsilon_o$. Hence, at any time instance, each of the longest queues decreases at a positive rate no less than ϵ_o. This is sufficient to conclude that the original queueing process is a positive recurrent Markov process (see [11]), which means the queues are stable. □

Next, we show Ω contains the previously-known regions of stability for LQF.

Lemma 21. *The following holds:* $\Sigma^*(G)\Lambda \subseteq \Omega$.

Proof. Consider any vector $\lambda \in \Sigma^*(G)\Lambda$. Let $S \subseteq V$ be an arbitrary non-empty node set. Let $n \in \arg\max_{k \in S} \sigma^*_k$. Since $\lambda \subseteq \Sigma^*(G)\Lambda$, by Lemma 19 and (3–8), $[\lambda]_S \in \sigma^*_n \Lambda_S \subseteq \sigma^*_S \Lambda_S$. Then, there exists a vector $\mu \in \Lambda_S$ such that $[\lambda]_S \leq \sigma^*_S \mu$. According to Definition 10, $\sigma^*_S \mu \not> \nu$ for any $\nu \in Co(M_S)$. Hence, $[\lambda]_S \not> \nu$ for any $\nu \in Co(M_S)$, implying $\lambda \in \Pi^c_S$. Since S is chosen arbitrarily, $\lambda \in \bigcap_{S \subseteq V} \Pi^c_S = \Omega$ by Definition 13. □

3.2.4 Shape of Ω Region

The previously-known regions of stability under LQF, such as $\Sigma^*(G)\Lambda$, are derived by reducing the capacity region through a linear transformation. Since the capacity region Λ is convex, each of these derived stability regions is also convex. In contrast, we will show that the shape of the Ω region is not convex in general. Furthermore, when the previously-known regions are not identical to Λ, they exclude many, if not most, of the extreme points of Λ. We will show that Ω contains all the extreme points of Λ.

Lemma 22. *The set of all the independent sets of the interference graph G, i.e., the set of all the feasible schedules, has a bijection to the set of all the extreme points of Λ.*

Note that we consider the empty schedule where no link is activated a trivial independent set. Lemma 22 establishes a connection between the graph topology and the geometry of Λ in a vector space.

Proof. Suppose ν is an independent set of G, represented by a 0-1 vector. Clearly, $\nu \in \Lambda$. Let us write $\nu = a\nu^1 + (1-a)\nu^2$ for some $0 \le a \le 1$, and $\nu^1, \nu^2 \in \Lambda$. Note that $\mu \le e$ for every $\mu \in Co(M_V)$. Thus, $0 \le \nu^1 \le e$ and $0 \le \nu^2 \le e$. For any index i, if $\nu_i = 0$, we must have $\nu_i^1 = 0$ and $\nu_i^2 = 0$. Similarly, if $\nu_i = 1$, we have $\nu_i^1 = \nu_i^2 = 1$. Therefore, $\nu = \nu^1 = \nu^2$ and ν is an extreme point of Λ.

Conversely, take any extreme point ν of Λ. Then, $\nu \le \mu$ for some $\mu \in Co(M_V)$. For an index i, if $\nu_i > 0$, we claim that $\nu_i = \mu_i$. Otherwise, we let $t = \mu_i - \nu_i > 0$. Then, we can create $\bar{\nu}$ and $\tilde{\nu}$ such that $\bar{\nu}_j = \tilde{\nu}_j = \nu_j$ for $j \neq i$. We can find an $\epsilon > 0$ such that $\bar{\nu}_i \triangleq \nu_i - \epsilon t \ge 0$, and we let $\tilde{\nu}_i = \mu_i > 0$. Since $0 \le \bar{\nu} \le \tilde{\nu} \le \mu$, we have $\bar{\nu}, \tilde{\nu} \in \Lambda$. It is easy to see $\nu = \frac{1}{1+\epsilon}\bar{\nu} + \frac{\epsilon}{1+\epsilon}\tilde{\nu}$, which implies ν is not an extreme point. Hence, either $\nu_i = 0$ or $\nu_i = \mu_i$, for all i.

We now show that $\nu_i = 0$ or $\nu_i = \mu_i = 1$ for all i. Write μ as $\mu = \sum_{j=1}^{k} a_j \mu^j$, where each $\mu^j \in M_V$, each $a_j > 0$, and $\sum_{j=1}^{k} a_j = 1$. Let $\nu_i^j = 0$ if $\nu_i = 0$; $\nu_i^j = \mu_i^j$ otherwise, for all $0 \le j \le k$. Because $\mu^j \in M_V$, we have $\nu_i^j = 0$ or $\nu_i^j = 1$. It is easy to check that $\nu = \sum_{j=1}^{k} a_j \nu^j$. Since $\nu^j \le \mu^j$, we have $\nu^j \in \Lambda$ for each j. Since ν is an extreme point of Λ, $\nu^1 = \nu^2 = \cdots = \nu^k$. Thus, $\nu_i = 1$ or $\nu_i = 0$ for all i.

It is easy to see that any 0-1 vector in $Co(M_V)$ must be a feasible schedule, i.e., an independent set of G. Since $\mu \in Co(M_V)$, the set of nodes, S, for which $\mu_i = 1$ forms an independent set. Let S' be the set of nodes for which $\nu_i = 1$. We have $S' \subseteq S$. Therefore, ν corresponds to an independent set. $\square$

Lemma 23. *Suppose λ is a vector corresponding to an independent set of the interference graph G. Then, $\lambda \in \Omega$.*

Proof. $[\lambda]_S$ is an independent set of the node-induced subgraph G_S for any non-empty $S \subseteq V$. Then, $[\lambda]_S \not> \nu$ for any $\nu \in Co(M_S)$, which implies $\lambda \notin \Pi_S$. Since S is arbitrary, we must have $\lambda \in \Omega$. $\square$

Corollary 4. *All the extreme points of the capacity region Λ belong to Ω.*

Proof. This is a result of Lemma 22 and Lemma 23. □

As an example, let G be the six cycle graph in Fig. 3-1. The arrival rate vector $\lambda = (1, 0, 1, 0, 1, 0)'$ corresponds to an independent set, and hence, $\lambda \in \Omega$. However, we know that $\Sigma^*(G) = \text{diag}(2/3, 2/3, 2/3, 2/3, 2/3, 2/3)$. As a result, $\lambda \notin \Sigma^*(G)\Lambda$. The example shows that Ω can be strictly larger than $\Sigma^*(G)\Lambda$. In the example, $\Omega - \Sigma^*(G)\Lambda$ contains not only the extreme points. For instance, one can check that, for $\lambda = (7/10, 1/10, 7/10, 1/10, 7/10, 1/10)'$, $\lambda \in \Omega$ but $\lambda \notin \Sigma^*(G)\Lambda$.

The next example shows that, the previously-discovered stability regions $\sigma^*(G)\Lambda$ and $\Sigma^*(G)\Lambda$ can underestimate the performance of LQF by an arbitrarily large factor in certain directions and in certain cases, whereas Ω can avoid such poor estimates.

Lemma 24. *For any $k > 0$, there exists an interference graph $G = (V, E)$ and an arrival rate vector λ such that $\lambda \notin \Sigma^*(G)\Lambda$, but $k\lambda \in \Omega$.*

Proof. Consider the bipartite graph in Fig. 3-2 with N pairs of nodes, where $N = 4$ in this particular case. It is almost a complete bipartite graph except that every corresponding pair of nodes (such as nodes 1 and 2) does not have an edge between them. It is easy to check that $\Sigma^*(G) = \text{diag}(2/N, 2/N, ..., 2/N)$. Therefore, the rate vector $\lambda = (2/N + \epsilon, 0, 0, ..., 0)'$, where $\epsilon > 0$, is not in $\Sigma^*(G)\Lambda$. For any $k > 0$, we can find a large enough N and a small enough ϵ such that $k(2/N + \epsilon) \leq 1$. Then, we have $k\lambda = (k(2/N + \epsilon), 0, 0, ..., 0)'$ in Ω. □

Though we cannot draw various regions in a high-dimensional vector space, it may still be helpful to make a highly simplified illustration with Fig. 3-3. The whole capacity region Λ is convex. The region $\Sigma^*(G)\Lambda$ is derived by scaling down the capacity region Λ using the diagonal matrix $\Sigma^*(G)$. This sort of scaling usually cuts off many or most extreme points of Λ. The newly defined stability region Ω is a superset of $\Sigma^*(G)\Lambda$ and Ω contains all the extreme points of Λ. The figure makes the point that Ω is not convex in general. We next show Ω is convex if and only if it is equal to Λ.

Lemma 25. *The following statements are equivalent.*

1. *Ω is a convex.*
2. *G is an overall local pooling graph.*
3. *$\Omega = \Lambda$.*

Proof. First, we prove that statement 1 implies statement 2. Suppose G is not overall local pooling. We claim that there must exist a non-empty set $S \subseteq V$ and $\mu, \nu \in Co(M_S)$ such that $\mu > \nu$. Since G is not overall local pooling, there exists a non-empty set $S \subseteq V$ such that $\sigma_S^* < 1$, which implies that there exist $\mu, \nu \in Co(M_S)$ and $\sigma_S^* \mu \geq \nu$, according to (3–5). If $\nu > 0$, we have the required set S, and $\mu, \nu \in Co(M_S)$ with $\mu > \nu$. If not, let $H = \{l \in S | \nu(l) > 0\}$. Because $\nu \in Co(M_S)$ and $[\nu]_{S-H} = 0$, it is easy to show $[\nu]_H \in Co(M_H)$[3] . Because $\mu \in Co(M_S)$ and $H \subseteq S$, there must exist $\tilde{\mu} \in Co(M_H)$ such that $\tilde{\mu} \geq [\mu]_H$. Then, $\sigma_S^* \tilde{\mu} \geq \sigma_S^* [\mu]_H \geq [\nu]_H > 0$. Thus, $\tilde{\mu} > [\nu]_H$ and $\tilde{\mu}, [\nu]_H \in Co(M_H)$. By renaming H to be S, $\tilde{\mu}$ to be μ and $[\nu]_H$ to be ν, we have the required set S and $\mu, \nu \in Co(M_S)$ with $\mu > \nu$.

Let $\lambda \in \mathbb{R}_+^{|V|}$ be an extended vector from μ such that $[\lambda]_S = \mu$ and $[\lambda]_{V-S} = 0$. According to Definition 12 and 13, $\lambda \notin \Omega$. Since $\mu \in Co(M_S)$, we can write $\mu = \sum_{i=1}^{|M_S|} \alpha_i m^i$, where $\sum_i \alpha_i = 1$ and $\alpha_i \geq 0$ for all i, and m^i for $i = 1, \dots, |M_S|$ are all the maximal schedules with respect to S. For each i, let $\tilde{m}^i$ be a $|V|$-dimensional vector extended from m^i, such that $\tilde{m}^i(j) = m^i(j)$ when $j \in S$ and $\tilde{m}^i(j) = 0$ when $j \notin S$. Clearly, each $\tilde{m}^i$ corresponds to an independent set of G. Hence, by Lemma 23, $\tilde{m}^i \in \Omega$ for all i. Since $\lambda = \sum_{i=1}^{|M_S|} \alpha_i \tilde{m}^i$ and $\lambda \notin \Omega$, we conclude that Ω is not convex.

[3] Suppose we write $M_S = (m^i)_{i=1}^{|M_S|}$, where each m^i is a maximal schedule with respect to S. We can represent ν as $\nu = \sum_{i=1}^{|M_S|} \alpha_i m^i$, where $\sum_i \alpha_i = 1$ and $\alpha_i \geq 0$ for all i. Since $[\nu]_{S-H} = 0$, we have $[m^i]_{S-H} = 0$ for each i. It is clear that $[m^i]_H$ corresponds to an independent set of G_H, the subgraph of G induced by H. Moreover, by the maximality of m^i with respect to S, if $m^i(j) = 0$ for some $j \in H$, it must be that $m^i(k) = 1$ for some $k \in H$ and j and k interfere with each other, i.e., $(j, k) \in E$. Therefore, $[m^i]_H$ must be maximal with respect to H. Hence, by $[\nu]_H = \sum_{i=1}^{|M_S|} \alpha_i [m^i]_H$, we get $[\nu]_H \in Co(M_H)$.

Next, we show that statement 2 implies statement 3. Since G is an overall local pooling graph, $\Sigma^*(G) = I$ (the identity matrix). By Lemma 21, $\Lambda \subseteq \Omega$. Hence, $\Omega = \Lambda$.

Finally, statement 3 implies statement 1 since Λ is convex. □

Remark. Suppose, for a given interference graph, the LQF algorithm does not achieve the full interior of the capacity region. Lemma 25 implies that Ω is not convex. Furthermore, since the closure of the full stability region of LQF (which is unknown) contains Ω, it contains all the extreme points of the capacity region Λ. Hence, the closure of the full stability region of LQF cannot be convex either, and it cannot be characterized by any linear transformation of the capacity region.

3.3 Stability Region Δ under LQF

In this section, we develop a notion termed as uniformly dominating vectors. It leads to a stability region Δ_C, which is a superset of Ω^o. When the arrival processes are not constant, i.e., when the variances of the i.i.d. arrival processes are non-zero, we obtain a stability region Δ_R, which contains Δ_C.

3.3.1 Motivating Examples

Example 1: We will first give an example to show that an arrival rate vector $\lambda \notin \Omega$ can sometime be stabilized by LQF. Hence, there is a region larger than Ω that captures the performance of LQF more precisely. The example also contains hints about how such a region can be defined.

Consider the six cycle graph G in Fig. 3-1. There are exactly five maximal schedules: $s^1 = (1,0,1,0,1,0)'$, $s^2 = (0,1,0,1,0,1)'$, $s^3 = (1,0,0,1,0,0)'$, $s^4 = (0,1,0,0,1,0)'$, $s^5 = (0,0,1,0,0,1)'$. Suppose the arrival rate vector is $\lambda = (5/12+\epsilon, 1/3+\epsilon, 1/3+\epsilon, 1/3+\epsilon, 1/3+\epsilon, 1/3+\epsilon)'$, where $\epsilon > 0$ is some small enough constant. Let $e = (1,1,1,1,1,1)'$, $\mu = \frac{1}{2}e$ and $\nu = \frac{1}{3}e$. Then, one can check that $\mu = \frac{1}{2}s^1 + \frac{1}{2}s^2$ and $\nu = \frac{1}{3}s^3 + \frac{1}{3}s^4 + \frac{1}{3}s^5$, which implies that $\mu, \nu \in Co(M_V)$. For $0 < \epsilon < 1/12$, $\nu < \lambda < \mu$. Hence, $\lambda \in \Lambda^o$ and $\lambda \notin \Omega$ by Definition 12 and 13.

Consider the fluid limit of the queue processes under LQF, denoted by $\{q_l(t)\}_{t\geq 0}$, for all $l \in V$. For a fixed (regular) time instance t, let S be the set of those longest queues whose time derivatives at t, $\dot{q}_l(t)$, are the largest. The queues in S will remain the longest with identical length in the next infinitesimally small time interval. Since $\lambda \in \Lambda^o$, $[\lambda]_S \in \Lambda_S^o$ by Lemma 19. If $S \neq V$, it is a fact that the node-induced subgraph G_S satisfies the local pooling condition [23]. An argument similar to that in the proof of Theorem 3.1 shows that the queues in S all have a negative drift.

The case of $S = V$ is more subtle. Since only the maximal schedules of G are used during the aforementioned infinitesimally small time interval, we can assume that the service rate vector is $\gamma = \sum_{i=1}^{5} \alpha_i s^i$, where $\sum_{i=1}^{5} \alpha_i = 1$ and $\alpha_i \geq 0$ for all i. In the fluid limit, $\dot{q}_l(t) = \lambda_l - \gamma_l$ for $l \in V$. By assumption, $\dot{q}_l(t)$ should be identical for all nodes $l \in V$. However, one can check that it is impossible to find such γ for the given λ. Therefore, the case of $S = V$ would not have occurred, and only the case of $S \neq V$ needs to be considered. Hence, G is stable under LQF for the given λ, according to the discussion for the $S \neq V$ case.

Example 2: Let $\lambda^1 = 0.7(1/2 - \epsilon, 1/2 - \epsilon, 1/2 - \epsilon, 1/2 - \epsilon, 1/2 - \epsilon, 1/2 - \epsilon)'$ and $\lambda^2 = (1/2 - \epsilon, 1/2 - 2\epsilon, 1/2 - 2\epsilon, 1/2 - 2\epsilon, 1/2 - 2\epsilon, 1/2 - 2\epsilon)'$ and $\epsilon = 10^{-3}$. Both λ^1 and λ^2 are outside Ω. Interestingly, although $\lambda^1 < \lambda^2$, λ^1 cannot be stabilized by LQF while λ^2 can. This has been verified by simulation experiments under constant arrivals. We will next develop a theory that provides a larger stability region and also can explain this counter-intuitive example.

3.3.2 Uniformly Dominating Vector and Δ_C Region

Definition 14. *Given a non-empty node set $S \subseteq V$, a vector $\lambda \in \mathbb{R}_+^{|V|}$ is said to be a uniformly dominating vector of S if $[\lambda]_S = \nu + de$ for some $\nu \in Co(M_S)$ and scaler $d \geq 0$. The region composed with all the uniformly dominating vectors of S is called the* *uniformly dominating region of S and is denoted by Γ_S. That is,*

$$\Gamma_S = \{\lambda \in \mathbb{R}_+^{|V|} \mid [\lambda]_S = \nu + de, \text{ for some } \nu \in Co(M_S) \text{ and some scalar } d \geq 0\}.$$

By convention, if $S = \emptyset$, we assume $\Gamma_S = \emptyset$.

Definition 15. *The Δ_C region is defined by*

$$\Delta_C = \bigcap_{S \subseteq V} \Gamma_S^c.$$

Remark. Note that a vector λ is outside Δ_C if and only if $\lambda \in \Gamma_S$ for some non-empty node set S.

Lemma 26. *For any non-empty $S \subseteq V$, Γ_S is closed. Hence, Δ_C is open.*

Proof. Let $B = \{de | d \geq 0\}$, where e is $|S|$ dimensional, and let $C = Co(M_S)$. It is easy to see C is compact and B is closed. From Definition 14, Γ_S is $B + C$ extended to the $|V|$-dimensional space. It can be shown that $B + C$ is closed, and hence, Γ_S is closed. Then, $\Delta_C = \bigcap_{S \subseteq V} \Gamma_S^c$ is open (with respect to the metric space $\mathbb{R}_+^{|V|}$). $\square$

Lemma 27. *Suppose $\lambda \in \Delta_C$ and suppose $S \subseteq V$ is a non-empty node set. If $\nu - [\lambda]_S = de$ for some $\nu \in Co(M_S)$, then $d > \epsilon_o$, for some $\epsilon_o > 0$ independent of S and ν.*

Proof. Suppose $\nu - [\lambda]_S = de$ (here, e is of $|S|$-dimension) for some $\nu \in Co(M_S)$. Since $\lambda \in \Delta_C$ and Δ_C is open, $\lambda + \epsilon_o e \in \Delta_C$ (here, e is of $|V|$-dimension) for some small enough $\epsilon_o > 0$ independent of S and ν. Then $\nu - [\lambda + \epsilon_o e]_S = (d - \epsilon_o)e$ or $[\lambda + \epsilon_o e]_S = \nu + (\epsilon_o - d)e$. Since $\lambda + \epsilon_o e \in \Delta_C$, $\lambda + \epsilon_o e \notin \Gamma_S$. Hence, $\epsilon_o - d < 0$ or $d > \epsilon^o$. $\square$

The constant ϵ_o will serve as a bound for the rate of the Lyapunov drift in the performance analysis.

3.3.3 Performance Guarantee of LQF in Δ_C Region

Theorem 3.2. *If an arrival rate vector λ satisfies $\lambda \in \Delta_C$, then, the network is stable under the LQF policy.*

Sketch of Proof. Again, consider the fluid limit of the queue process and apply a similar argument as in the proof of Theorem 3.1. Let $S \subseteq V$ be the set of nodes whose queues are the longest at time t and will remain the longest for the next infinitesimally small time interval. Let ν_S be the service rate vector for the nodes in S at time t. Under LQF, $\nu_S \in Co(M_S)$ and $\nu_S - [\lambda]_S = \epsilon e$ for some ϵ. Since $\lambda \in \Delta_C$, by Lemma 27, we have $\epsilon > \epsilon_o$ for some $\epsilon_o > 0$ independent of S and ν. Hence, at any time instance, each of the longest queues decreases at a positive rate no less than ϵ_o. This is sufficient to conclude that the original queueing process is a positive recurrent Markov process, which means the queues are stable. $\square$

Lemma 28. $\Omega^o \subseteq \Delta_C$.

Proof. Suppose $\Omega^o \nsubseteq \Delta_C$. Then, there exists a vector $\lambda \in \Omega^o$ and $\lambda \notin \Delta_C$. Hence, $[\lambda]_S = \nu + de$ for some non-empty $S \subseteq V$, $\nu \in Co(M_S)$ and $d \geq 0$. Since $\lambda \in \Omega^o$, $\lambda + \epsilon e \in \Omega$ for some small enough $\epsilon > 0$. From $[\lambda + \epsilon e]_S = \nu + (d + \epsilon)e$ and $d + \epsilon > d \geq 0$, we have $[\lambda + \epsilon e]_S > \nu$. Hence, $\lambda + \epsilon e \notin \Omega$, leading to a contradiction. $\square$

Consider Example 2 in Section 3.3.1. With the linear programming tools introduced in Section 3.4, one can check that $\lambda^2 \in \Delta_C$ but $\lambda^1 \notin \Delta_C$. This explains why λ^2 can be stabilized by LQF while λ^1 cannot, even though $\lambda^2 > \lambda^1$.

3.3.4 Rank Condition and Δ_R Region

For the same average arrival rate vector, whether the i.i.d. arrival processes have zero or non-zero variances leads to significantly different stability behavior (in the former case, the arrival processes are deterministic with constant rates). This issue has been discussed in [12] where the authors develop a queue separation result related to a rank condition about the matrices of the maximal independent sets. We next generalize the rank condition. Then, we extend Δ_C to a larger stability region Δ_R. We will show Δ_R can be stabilized under LQF when the arrival processes all have non-zero variances.

Definition 16. *Let $S \subseteq V$ be a non-empty set. We call the matrix (M_S, e) the extended schedule matrix for S (or graph G_S). Let $R(M_S, e)$ denote the rank of the extended schedule matrix, i.e., the number of linearly independent columns in the matrix (M_S, e). We say S (or graph G_S) has a high rank if $R(M_S, e) = |S|$. Otherwise, we say S (or G_S) has a low rank.*

Suppose $S \subseteq V$ is the set of nodes with the longest queues at some time instance. When the arrival has non-zero variances, the queue separation result suggests (Lemma 1 and Lemma 3 of [12]): If the rank $R(M_S) \leq |S| - 2$, then, with probability 1, the queue sizes of S will not stay identical in the next infinitesimal time interval. We find that the condition $R(M_S) \leq |S| - 2$ can be relaxed to $R(M_S, e) \leq |S| - 1$, i.e., the low rank condition in Definition 16. The queue separation lemma (Lemma 1 of [12]) uses the assumption $R(M_S) \leq |S| - 2$ to obtain a vector ν such that $\nu' e = 0$ and $\nu' M_S = 0$. Such a vector ν still exists when the low rank condition in Definition 16 is satisfied. Then, every subsequent step in the proof of the queue separation lemma still holds. The low rank condition is a generalization since $R(M_S) \leq |S| - 2$ implies $R(M_S, e) \leq |S| - 1$.

Roughly speaking, when the variances are non-zero, the randomness in the arrival processes pressures the queues in S to move around in an $|S|$-dimensional space. This means that the $|S|$ queues cannot be simultaneously the longest queues for a sustained period of time (in which case, the queue trajectory moves along a line), unless the service can fully compensate the pressure from the arrival processes. But, full compensation is not possible in the low-rank case since the service rate vector lives in a lower-dimensional space. What will happen is that some subset of the queues in S with a high rank will dominate the rest. This is known as queue separation. The implication is that, in the case of non-zero variances, there is no need to consider the low-rank subsets of V when evaluating the performance degradation of LQF. The discussion motivates the following definition of Δ_R.

Definition 17. *The Δ_R region is defined by*

$$\Delta_R = \bigcap_{S \subseteq V, S \text{ with high rank}} \Gamma_S^c.$$

In words, a vector λ is outside Δ_R if and only if $\lambda \in \Gamma_S$ for some node set S that has a high rank.

By comparing the definitions of Δ_C and Δ_R, we have the following lemma.

Lemma 29. $\Delta_C \subseteq \Delta_R$.

In addition to the i.i.d and mutually independent assumptions, the following assumption on the arrival processes is needed for technical reasons (see [12] for their relevance).

A1: (The large deviation bound on the arrival processes) Let $A_l(n)$ be the cumulative arrivals at queue l (at node $l \in V$) up to time n, and let λ_l be the average arrival rate at queue l. For each $\epsilon > 0$,

$$P(|\frac{A_l(n)}{n} - \lambda_l| > \epsilon) \leq \beta \exp(-n\gamma(\epsilon)) \text{ for all } n \geq 1, \text{ for some } \gamma(\epsilon) > 0 \text{ and } \beta > 0.$$

Theorem 3.3. *Assume the condition in A1 holds and assume the variance of the i.i.d arrival process to each node is non-zero but finite. If an arrival rate vector λ satisfies $\lambda \in \Delta_R$, then, the network is stable under the LQF policy.*

Sketch of Proof. Again, consider the fluid limit of the queue process and apply a similar argument as in the proof of Theorem 3.1. Let $S \subseteq V$ be the set of nodes whose queues are the longest at time t and will remain the longest for the next infinitesimally small time interval. By replicating most of the arguments in the queue separation lemmas (Lemma 1 and Lemma 3 in [12]), it can be shown that S must have a high rank[4] . Otherwise,

[4] The only change is to Lemma 3 in [12]. Instead of saying for any low-rank set, there must be a subset that satisfies local pooling, we say for any low-rank set, there must be a subset that is of high rank. This is so because a set with a single node is of high rank.

the queue sizes of the nodes in S will be separated and they cannot all remain the longest. Hence, we can apply the same argument as that in Theorem 3.2, but only to the high-rank node sets. $\square$

Some graph examples are given in Fig. 3-4, regarding their set σ-local pooling factors and ranks. Note that, the shaded region includes those subsets S which either satisfy $\sigma_S^* = 1$, i.e., set local pooling (SLoP), or have low rank. Those subsets need not to be considered for the performance of LQF in case of non-zero variances.

3.3.5 Further Properties of Regions Δ_C and Δ_R

It has been demonstrated that $\Delta_C \subseteq \Delta_R$. We now continue to study the properties of the two regions and their relationship.

Theorem 3.4. *The closures of Δ_C and Δ_R are the same, i.e. $\overline{\Delta_C} = \overline{\Delta_R}$.*

Proof. Since $\Delta_C \subseteq \Delta_R$, we have $\overline{\Delta_C} \subseteq \overline{\Delta_R}$. We will next show $\overline{\Delta_R} \subseteq \overline{\Delta_C}$. Since $\Delta_R = \Delta_C \bigcup (\Delta_R - \Delta_C)$ and $\overline{\Delta_R} = \overline{\Delta_C} \bigcup \overline{(\Delta_R - \Delta_C)}$, we only need to show $\overline{(\Delta_R - \Delta_C)} \subseteq \overline{\Delta_C}$.

Given any vector $\tilde{\lambda} \in \Delta_R - \Delta_C$, by comparing Definition 15 and 17, we have

$$\tilde{\lambda} \in \bigcap_{\substack{S \subseteq V \\ S \text{ with high rank}}} \Gamma_S^c \bigcap \Big(\bigcup_{\substack{S \subseteq V \\ S \text{ with low rank}}} \Gamma_S \Big).$$

By Lemma 26, Γ_S is a closed set and Γ_S^c is open. Hence, Δ_R is open. Therefore, there exists $\delta > 0$ such that $\gamma \in \Delta_R$ whenever $\gamma \geq 0$ and the distance between the two vectors $d(\tilde{\lambda}, \gamma) < \delta$.

Let $0 < \epsilon < \frac{1}{2\sqrt{|V|}}\delta$ and $\lambda = \tilde{\lambda} + \epsilon e$. Then, the distance between λ and $\tilde{\lambda}$ is $d(\lambda, \tilde{\lambda}) = |\epsilon e| < \frac{1}{2\sqrt{|V|}}\delta |e| = \frac{1}{2}\delta$. Then, $\lambda \in \Delta_R$ and $\lambda \geq \epsilon e$.

Now, let $Q(\lambda) = \{S | S \subseteq V, S \text{ with low rank}, \lambda \in \Gamma_S\}$. We will next construct a sequence of low-rank node sets, S_i, for $i = 1, 2, \cdots$. Since each of them has a low

The modification is needed in the proof of Lemma 3 in [12]. The statement of Lemma 3 also needs to be modified accordingly.

rank, there exists an $|S_i|$-dimensional vector $g^i \neq 0$ with $||g^i|| = 1$ such that $(g^i)'e = 0$ and $(g^i)'M_{S_i} = 0$. We then extend each g^i to a $|V|$-dimensional vector by setting the values of the new components to be zero. With a little abuse of notation, we call this $|V|$-dimensional vector g^i as well.

We now construct the sequence of S_i. If $Q(\lambda) \neq \emptyset$, pick any subset $S_1 \in Q(\lambda)$. Let $\lambda^1 = \lambda + 1/2\epsilon g^1$. Next, if $Q(\lambda^1) \neq \emptyset$, pick any $S_2 \in Q(\lambda^1)$ and let $\lambda^2 = \lambda^1 + 1/2^2\epsilon g^2$. In step j, if $Q(\lambda^{j-1}) \neq \emptyset$, we will pick any $S^j \in Q(\lambda^{j-1})$ and let $\lambda^j = \lambda^{j-1} + 1/2^j\epsilon g^j$. This procedure will go on until $Q(\lambda^j)$ becomes empty for some j. We can check that the ith component of λ^j is $\lambda^j(i) = (\lambda + 1/2\epsilon g^1 + 1/2^2\epsilon g^2 + \cdots + 1/2^j\epsilon g^j)(i) \geq \epsilon - 1/2\epsilon - 1/2^2\epsilon - \cdots - 1/2^j\epsilon \geq 0$. This ensures that λ^j is always a non-negative vector for all j.

Now, we will show that there exists an integer $K \geq 0$ such that $Q(\lambda^K)$ becomes empty for the first time (hence, the sequence of S_j ends at S_{K-1}, or contains no elements if $K = 0$). For convenience, let $\lambda^0 = \lambda$.

We will show that $S_j \notin Q(\lambda^k)$ for $k \geq j$, where $S_j \neq \emptyset$. Suppose $S_j \in Q(\lambda^k)$ for some $k \geq j$. Then, $\lambda^k \in \Gamma_{S_j}$, which implies that $[\lambda^k]_{S_j} = d^1 e + \nu^1$ for some $d^1 \geq 0$ and $\nu^1 \in Co(M_{S_j})$. From the construction procedure, we know that $S_j \in Q(\lambda^{j-1})$, which implies that $[\lambda^{j-1}]_{S_j} = d^2 e + \nu^2$ for some $d^2 \geq 0$ and $\nu^2 \in Co(M_{S_j})$. Since

$$\lambda^k = \lambda^{j-1} + 1/2^j\epsilon g^j + 1/2^{j+1}\epsilon g^{j+1} + \cdots + 1/2^k\epsilon g^k,$$

we have

$$[\lambda^k]_{S_j} = d^2 e + \nu^2 + [1/2^j\epsilon g^j + 1/2^{j+1}\epsilon g^{j+1} + \cdots + 1/2^k\epsilon g^k]_{S_j}.$$

Then,

$$\begin{aligned}(g^j)'[\lambda^k]_{S_j} =& (g^j)'(d^2 e + \nu^2 + [1/2^j\epsilon g^j + 1/2^{j+1}\epsilon g^{j+1} + \cdots + 1/2^k\epsilon g^k]_{S_j}) \\ =& 1/2^j\epsilon||g^j||^2 + 1/2^{j+1}\epsilon(g^j)'[g^{j+1}]_{S_j} + \cdots + 1/2^k\epsilon(g^j)'[g^k]_{S_j} \\ \geq& 1/2^j\epsilon - 1/2^{j+1}\epsilon - \cdots - 1/2^k\epsilon > 0.\end{aligned}$$

However, since $[\lambda^k]_{S_j} = d^1 e + \nu^1$, we have $(g^j)'[\lambda^k]_{S_j} = (g^j)'(d^1 e + \nu^1) = 0$, leading to a contradiction. Hence, $S_j \notin Q(\lambda^k)$ for $k \geq j$.

In summary, each non-empty S_j in the constructed sequence is in $Q(\lambda^{j-1})$ but not in $Q(\lambda^k)$ for $k \geq j$. Hence, each S_j is distinct. Since there is a finite number of non-empty node sets $S \subseteq V$, there exists an integer $K \geq 0$ such that $Q(\lambda^K)$ becomes empty for the first time.

Then, $\lambda^K \notin \Gamma_S$ for any node set S with a low rank. Hence, $\lambda^K \in \bigcap_{S\subseteq V, S \text{ with low rank}} \Gamma_S^c$. The distance between λ and λ^K is $d(\lambda, \lambda^K) \leq \epsilon(1/2 + 1/2^2 + ... + 1/2^K) < \epsilon$. Then, the distance between $\tilde{\lambda}$ and λ^K is $d(\tilde{\lambda}, \lambda^K) \leq d(\tilde{\lambda}, \lambda) + d(\lambda, \lambda^K) \leq \epsilon\sqrt{|V|} + \epsilon < \delta$. Hence, $\lambda^K \in \Delta_R$. It follows $\lambda^K \in \bigcap_{S\subseteq V} \Gamma_S^c = \Delta_C$.

Since ϵ can be chosen arbitrarily small, $\tilde{\lambda}$ is a limit point of Δ_C. Thus, $\tilde{\lambda} \in \overline{\Delta_C}$, implying $(\Delta_R - \Delta_C) \subseteq \overline{\Delta_C}$. Hence, $\overline{(\Delta_R - \Delta_C)} \subseteq \overline{\Delta_C}$. □

The following is an intermediary lemma.

Lemma 30. *If a non-empty set $S \subseteq V$ satisfies $\sigma_S^* = 1$, then $\Gamma_S \bigcap \Lambda^o = \emptyset$.*

Proof. Suppose there exists a vector $\lambda \in \Gamma_S \bigcap \Lambda^o$. By Definition 14, $[\lambda]_S = \nu + de$ for some $d \geq 0$ and $\nu \in Co(M_S)$. Since $\lambda \in \Lambda^o$, by Lemma 19, $[\lambda]_S + \epsilon e \leq \mu$ for some $\mu \in Co(M_S)$ and a small enough $\epsilon > 0$. Hence, $\mu \geq \nu + (d+\epsilon)e \geq \nu(1 + d + \epsilon)$. Thus, $\sigma_S^* < 1$ and we arrive at a contradiction. □

Lemma 31. *If every high-rank node set $S \subseteq V$ satisfies $\sigma_S^* = 1$, then, $\overline{\Delta_C} = \overline{\Delta_R} = \Lambda$ and $\Delta_R = \Lambda^o$.*

Proof. According to Definition 17, we have $\Delta_R = \bigcap_{S\subseteq V, S \text{ with high rank}} \Gamma_S^c$. For any high-rank node set S, since $\sigma_S^* = 1$, we have $\Gamma_S \bigcap \Lambda^o = \emptyset$ by Lemma 30, which implies $\Gamma_S^c \bigcap \Lambda^o = \Lambda^o$. Hence, $\Delta_R \bigcap \Lambda^o = \bigcap_{S\subseteq V, S \text{ with high rank}} \Gamma_S^c \bigcap \Lambda^o = \Lambda^o$. Combining this with Theorem 3.4, we get $\overline{\Delta_C} = \overline{\Delta_R} = \overline{\Lambda^o} = \Lambda$. Also, the fact that $\Delta_R \bigcap \Lambda^o = \Lambda^o$ implies $\Lambda^o \subseteq \Delta_R$. Since Δ_R is an open set in Λ and Λ^o is the largest open set in Λ, it must be that $\Delta_R = \Lambda^o$. □

Remark. From Lemma 31, we know that when all the subsets $S \subseteq V$ satisfy either set local pooling (i.e., $\sigma_S^* = 1$) or the rank of S is low, then $\Delta_R = \Lambda^o$. That is, the entire Λ^o is achievable by LQF, assuming the arrival processes have non-zero variances. This is the same statement as Theorem 1 of [12]. Thus, the newly developed theory here is able to reproduce the main result of [12].

Lemma 32. *$\overline{\Delta_C} = \Lambda$ if and only if $\overline{\Delta_C}$ is convex. Similarly, $\overline{\Delta_R} = \Lambda$ if and only if $\overline{\Delta_R}$ is convex.*

Proof. It is obvious that $\overline{\Delta_C} = \Lambda$ implies $\overline{\Delta_C}$ is convex. We will next show the converse. Since $\Delta_C \subseteq \Lambda$ and Λ is a closed set, we have $\overline{\Delta_C} \subseteq \Lambda$. Because $\Omega^o \subseteq \Delta_C$, we have $\Omega \subseteq \overline{\Delta_C}$. Since Ω contains all the extreme points of Λ (Corollary 4), $\overline{\Delta_C}$ also contains all of them. Since Λ is the convex combination of all its extreme points and $\overline{\Delta_C}$ is convex, we must have $\Lambda \subseteq \overline{\Delta_C}$.

The second statement can be proved similarly. □

3.4 Graph Coloring and LQF scheduling

The scheduling problem in this paper is deeply connected with graph coloring and its related problems. In this section, we will introduce fractional coloring, and more generally, aspects of the fractional graph theory that can provide useful tools for studying the stability regions discussed in the previous sections.

3.4.1 Fractional Coloring and Capacity Region

The chromatic number of a graph G, denoted by $\chi(G)$, is the minimum number of colors needed to paint the nodes so that the connected nodes do not share the same color. When we relax the integrality constraints of the chromatic number problem and introduce a parameter $\lambda \in \mathbb{R}_+^{|V|}$, we obtain the following linear programming (LP) problem.

Definition 18. *Given a graph* $G = (V, E)$ *and* $\lambda \in \mathbb{R}_+^{|V|}$*, the* weighted fractional coloring *problem with the weight vector* λ *is:*

$$\chi_f(G, \lambda) \triangleq \min\ e'\alpha, \text{ subject to } M_V\alpha \geq \lambda, \alpha \geq 0. \tag{3–9}$$

The optimal value of the above problem, $\chi_f(G, \lambda)$, is called the weighted fractional chromatic number, which is known to be related to the capacity region as follows (see [6]):

$$\Lambda = \{\lambda \in \mathbb{R}_+^{|V|} \mid \chi_f(G, \lambda) \leq 1\}. \tag{3–10}$$

Based on (3–9), $\chi_f(G, \lambda)$ can be interpreted as the fastest way of serving queued data when the queue sizes are proportional to the weights λ. Based on (3–10), $\chi_f(G, \lambda)$ can be interpreted as the 'traffic load' to the network.

The relevance and usefulness of this problem to the study of wireless scheduling have been amply demonstrated in [6]. The characterization of the capacity region by (3–10) suggests that the fractional chromatic number can serve as an oracle for judging whether an arrival rate vector is in the capacity region or not. With this observation and with known complexity results about the fractional coloring problem, the authors of [6] have derived results about the inherent complexity of the wireless scheduling problem.

3.4.2 Weighted Fractional Matching Number and Ω Region

We next discuss the problem of finding the weighted fractional matching number of a graph [32]. This problem can help to decide whether a vector is in Ω.

Definition 19. *Given a graph* $G = (V, E)$ *and* $\lambda \in \mathbb{R}_+^{|V|}$*, the* weighted fractional matching number *problem with the weight vector* λ *is:*

$$\phi_f(G, \lambda) \triangleq \max\ e'\beta, \text{ subject to } M_V\beta \leq \lambda, \beta \geq 0. \tag{3–11}$$

The above problem is the Lagrangian dual of the weighted fractional transversal number problem, which is the hypergraph dual problem of the weighted fractional coloring problem [32]. Here, the ith component of β can be interpreted as the amount of time for which the ith maximal schedule is used. The weighted fractional matching number, $\phi_f(G, \lambda)$, can be interpreted as the slowest way of serving the queued data (in the amount λ) using only the maximal schedules, subject to the additional constraint that a schedule should not be selected if it activates a link associated with an empty queue.

Lemma 33. *The Ω region satisfies the following:*

$$\Omega = \{\lambda \in \mathbb{R}_+^{|V|} \mid \phi_f(G_S, [\lambda]_S) \leq 1, \forall S \subseteq V, S \neq \emptyset\}.$$

Proof. Consider any vector $\lambda \in \Omega$ and an arbitrary non-empty node set $S \subseteq V$. Suppose $\phi_f(G_S, [\lambda]_S) > 1$. Then, by Definition 19, we have $[\lambda]_S \geq k\nu$ for some $\nu \in Co(M_S)$ and $k > 1$. Let Z be the largest subset in S such that $[\nu]_Z = 0$. Note that $Z \neq S$. Then, the vector $[\nu]_{S-Z} \in Co(M_{S-Z})$ and $[\nu]_{S-Z} > 0$. Hence, $[\lambda]_{S-Z} \geq k[\nu]_{S-Z} > [\nu]_{S-Z}$, which implies that $\lambda \in \Pi_{S-Z}$. According to Definition 12 and 13, $\lambda \notin \Omega$.

Conversely, suppose a vector λ satisfies $\phi_f(G_S, [\lambda]_S) \leq 1$ for every non-empty $S \subseteq V$. Then, $[\lambda]_S \not> \nu$ for any $\nu \in Co(M_S)$. Otherwise, there would exist $k > 1$ such that $[\lambda]_S \geq k\nu$ for some $\nu \in Co(M_S)$, which implies that $\phi_f(G_S, [\lambda]_S) > 1$. Thus, $\lambda \in \Omega$. □

3.4.3 Hypergraph Duality and Set σ-local Pooling

We next relate the ratio of $\chi_f(G, \lambda)$ and $\phi_f(G, \lambda)$ to the set σ-local pooling factor. First, we have the following lemma.

Lemma 34. *For any $k \geq 0$, $\chi_f(G, k\lambda) = k\chi_f(G, \lambda)$ and $\phi_f(G, k\lambda) = k\phi_f(G, \lambda)$.*

Proof. The case of $k = 0$ is trivial. We only focus on the case of $k > 0$. Suppose β^* is an optimal solution to the problem in (3–11) for finding $\phi_f(G, \lambda)$. Then, $k\beta^*$ is feasible to the problem for finding $\phi_f(G, k\lambda)$. Since $e'(k\beta^*) = ke'\beta^*$, $\phi_f(G, k\lambda) \geq k\phi_f(G, \lambda)$.

Conversely, suppose $\tilde{\beta}$ is an optimal solution to the problem for finding $\phi_f(G, k\lambda)$. Then, $\tilde{\beta}/k$ is feasible to the problem for finding $\phi_f(G, \lambda)$. Since $e'(\tilde{\beta}/k) = e'\tilde{\beta}/k$, $\phi_f(G, \lambda) \geq \phi_f(G, k\lambda)/k$. Therefore, $\phi_f(G, k\lambda) = k\phi_f(G, \lambda)$, for $k > 0$. A similar argument can be used to show $\chi_f(G, k\lambda) = k\chi_f(G, \lambda)$. □

Theorem 3.5. *Given a non-empty node set $S \subseteq V$, the set σ-local pooling factor of S satisfies the following*[5] *:*

$$\sigma_S^* = \min_{\lambda \geq 0} \frac{\chi_f(G_S, \lambda)}{\phi_f(G_S, \lambda)}. \tag{3–12}$$

Proof. Suppose $(\sigma_S^*, \mu_S^*, \nu_S^*)$ is an optimal solution to the problem in (3–5). Then, we choose $\lambda = \nu_S^*$. Since $\sigma_S^* \mu_S^* \geq \nu_S^*$, we have

$$\begin{aligned} \chi_f(G_S, \lambda) &= \chi_f(G_S, \nu_S^*) \\ &\leq \chi_f(G_S, \sigma_S^* \mu_S^*) = \sigma_S^* \chi_f(G_S, \mu_S^*) \leq \sigma_S^*. \end{aligned}$$

The last inequality above uses the fact $\chi_f(G_S, \mu_S^*) \leq 1$, which follows from (3–10) (since $\mu_S^* \in \Lambda_S$).

Because $\nu_S^* \in Co(M_S)$, there exists a non-negative vector β such that $M_S \beta = \nu_S^*$ and $e'\beta = 1$. Such β is feasible to (3–11) for finding $\phi_f(G_S, \lambda)$. Hence, $\phi_f(G_S, \lambda) \geq 1$. Therefore, $\min_{\lambda \geq 0} \chi_f(G_S, \lambda)/\phi_f(G_S, \lambda) \leq \sigma_S^*$.

Next, suppose λ^* is an optimal solution for the problem $\min_{\lambda \geq 0} \chi_f(G_S, \lambda)/\phi_f(G_S, \lambda)$. Suppose α^* and β^* are optimal solutions for the problems of finding $\chi_f(G_S, \lambda^*)$ and $\phi_f(G_S, \lambda^*)$, respectively. Then, we have $\min_{\lambda \geq 0} \chi_f(G_S, \lambda)/\phi_f(G_S, \lambda) = \sum_i \alpha_i^* / \sum_i \beta_i^*$. Now, let $\mu = M_S \alpha^* / \sum_i \alpha_i^*$ and $\nu = M_S \beta^* / \sum_i \beta_i^*$. Then, $\mu, \nu \in Co(M_S)$ and

$$\frac{\chi_f(G_S, \lambda^*)}{\phi_f(G_S, \lambda^*)} \mu = \frac{\sum_i \alpha_i^*}{\sum_i \beta_i^*} \frac{M_S \alpha^*}{\sum_i \alpha_i^*} = \frac{M_S \alpha^*}{\sum_i \beta_i^*}.$$

[5] We take the convention $a/0 = \infty$ for any scalar $a \geq 0$. Note that, if any component of λ is equal to zero, then $\phi_f(G_S, \lambda) = 0$. As a result, the optimal solution λ^* to (3–12) must satisfy $\lambda^* > 0$.

By the feasibility of α^* and β^* to (3–9) and (3–11), respectively, $M_S\alpha^* \geq \lambda^* \geq M_S\beta^*$. Hence,

$$\frac{M_S\alpha^*}{\sum_i \beta_i^*} \geq \frac{\lambda^*}{\sum_i \beta_i^*} \geq \frac{M_S\beta^*}{\sum_i \beta_i^*} = \nu.$$

Thus, $\chi_f(G_S, \lambda^*)/\phi_f(G_S, \lambda^*)$ is feasible to the problem in (3–5). Therefore, we have the following $\min_{\lambda \geq 0} \chi_f(G_S, \lambda)/\phi_f(G_S, \lambda) \geq \sigma_S^*$. □

The theorem above shows that the set σ-local pooling factor is the same as the minimum of the hypergraph duality ratios over all different weights.

3.4.4 Weighted Fractional Domination Number and Δ_C Region

Definition 20. *Given a graph $G = (V, E)$ and $\lambda \in \mathbb{R}_+^{|V|}$, the* weighted fractional domination number *problem with the weight vector λ is:*

$$\tau_f(G, \lambda) \triangleq \max d, \text{ subject to } de + \nu = \lambda, \nu \in Co(M_V).$$

For convenience, let $\tau_f(G, \lambda) = -\infty$ when the problem is infeasible.

By Definition 14, 15 and 20, we have the following lemma.

Lemma 35. *The following relations hold:*

$$\Gamma_S = \{\lambda \in \mathbb{R}_+^{|V|} \mid \tau_f(G_S, [\lambda]_S) \geq 0, \text{ for } S \subseteq V, S \neq \emptyset\},$$

$$\Delta_C = \{\lambda \in \mathbb{R}_+^{|V|} \mid \tau_f(G_S, [\lambda]_S) < 0, \forall S \subseteq V, S \neq \emptyset\}.$$

3.5 Experimental Examples

In this section, we show some simulation results. The main purpose is to confirm some of the less intuitive theoretical results. We first show the performance of LQF on the six-cycle graph, denoted by C_6, for arrival rate vectors in different sections of the capacity region. For C_6, LQF can achieve the entire interior of the capacity region for arrivals satisfying assumption A1 and with non-zero variances. On the other hand, for constant arrivals, experiments have shown that some rate vectors in the interior of the capacity region are not achievable by LQF.

In the experiments with constant arrivals, we use a *load* parameter to scale the arrival rate vectors. Each experiment runs for 10^6 iterations with initial queue sizes of 10^3. The following arrival rate vectors are used for the results in Fig. 3-5:

$$\begin{aligned}
\lambda^1 &= (\frac{1}{2}-\epsilon, \frac{1}{2}-\epsilon, \frac{1}{2}-\epsilon, \frac{1}{2}-\epsilon, \frac{1}{2}-\epsilon, \frac{1}{2}-\epsilon)' \\
\lambda^2 &= (\frac{1}{2}-\epsilon, \frac{1}{2}-2\epsilon, \frac{1}{2}-3\epsilon, \frac{1}{2}-\epsilon, \frac{1}{2}-2\epsilon, \frac{1}{2}-3\epsilon)' \\
\lambda^3 &= (\frac{1}{2}-\epsilon, \frac{1}{2}-2\epsilon, \frac{1}{2}-2\epsilon, \frac{1}{2}-2\epsilon, \frac{1}{2}-2\epsilon, \frac{1}{2}-2\epsilon)',
\end{aligned}$$

where $\epsilon = 10^{-3}$. Note that $0.7\lambda^1 < 0.95\lambda^2 < \lambda^3$. However, judging by the queue sizes in Fig. 3-5, the arrival rate vectors $0.7\lambda^1$ and $0.95\lambda^2$ seem to be not stabilizable, whereas λ^3 seems to be stabilizable. The theory allows this counter-intuitive phenomenon. Readers can verify that $0.7\lambda^1, 0.95\lambda^2 \notin \Delta_C$ while $\lambda^3 \in \Delta_C$.

In Section 3.3, we generalize the definitions of high or low-rank graphs. In Fig. 3-6, we provide an interference graph that is not set local pooling ($\sigma_S^* < 1$) and is of high rank according to the original definition in [12]. However, in the new definition, the graph is of low rank. We ran simulations with initial queue sizes of 10^3 using the Bernoulli arrivals with an identical arrival rate of 0.499. Fig. 3-7 shows the evolution of the average queue size for nodes 1-8 and 9-13 over 10^6 iterations. The queues for nodes 1-8 appear to be unstable and the queues for nodes 9-13 appear to be stable. Our refinement of the rank condition rules out the possibility that the queues of all nodes are simultaneously the longest and remain longest, whereas the previous rank condition does not rule that out.

3.6 Summary

In this chapter, we investigate the performance guarantee of the LQF scheduling policy in wireless networks. The objective is to discover new stability regions of LQF that are larger than those previously known, and to improve our knowledge about the largest possible stability region of LQF. We show that it is necessary to go beyond the existing framework of linear reduction of the capacity region, and move to a non-linear framework.

We introduce the concepts of strictly dominating vectors and uniformly dominating vectors; the former leads to the new stability region of LQF, Ω, and the latter leads the stability regions Δ_C and Δ_R. We show that Ω contains $\Sigma^*(G)\Lambda$, which is the stability region given in [23]. We also show $\Omega^o \subseteq \Delta_C \subseteq \Delta_R$. Hence, the new stability regions all capture the performance of LQF better. Contrary to the previously-known regions of stability, the closures of these new stability regions contain all the extreme points of the capacity region Λ, but they are not convex in general. The only case where they are convex is when they are equal to the capacity region itself, which occurs only for selected interference graphs. The general lack of convexity is not a defect of the theory. We show that, when LQF cannot achieve the full capacity region, the largest achievable region cannot be convex.

The study reveals a counter-intuitive situation where increasing the arrival rates helps LQF to stabilize the network. It turns out, in this case, the original rate vector is outside Δ_C, and after the rate increase, the new rate vector is inside Δ_C. We also generalize the rank condition studied in [12], and with this generalization, refine the stability results for non-deterministic arrivals. We can show that if a set of nodes satisfies the new low-rank condition, the queue sizes of these nodes will be separated. Based on this result, we can enlarge Δ_C to Δ_R, which is achievable by LQF under non-deterministic arrivals. Interestingly, we show that the closures of Δ_C and Δ_R are the same. Finally, we introduce several linear programming problems encountered in the fractional graph theory, which can provide tools for studying the newly developed stability regions. We show that a ratio between the weighted fractional coloring number and the weighted fractional matching number is related to the set σ-local pooling factor introduced in [23].

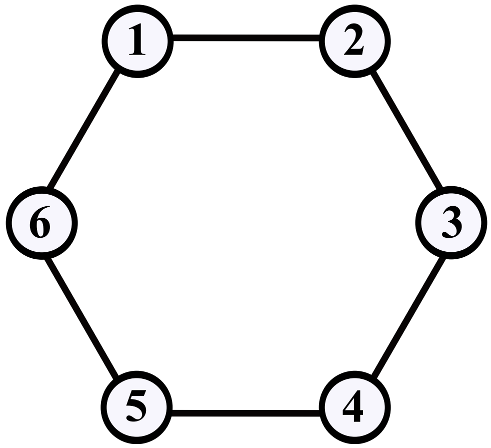

Figure 3-1. The six-cycle graph, C_6.

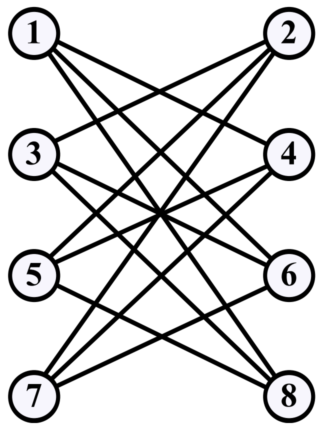

Figure 3-2. A bipartite graph.

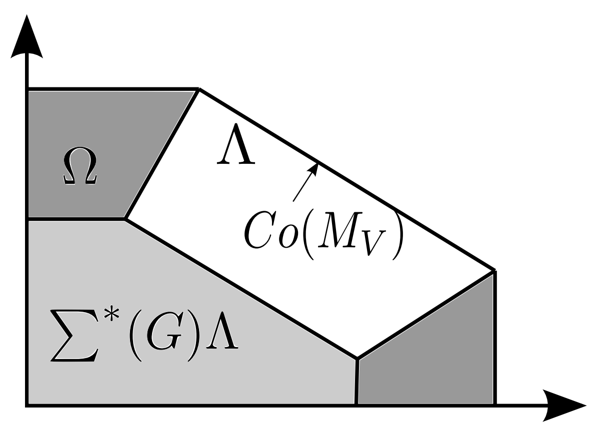

Figure 3-3. The Ω region and other relevant regions. The largest convex polytope is Λ. The entire shaded region is Ω, which is not a convex set.

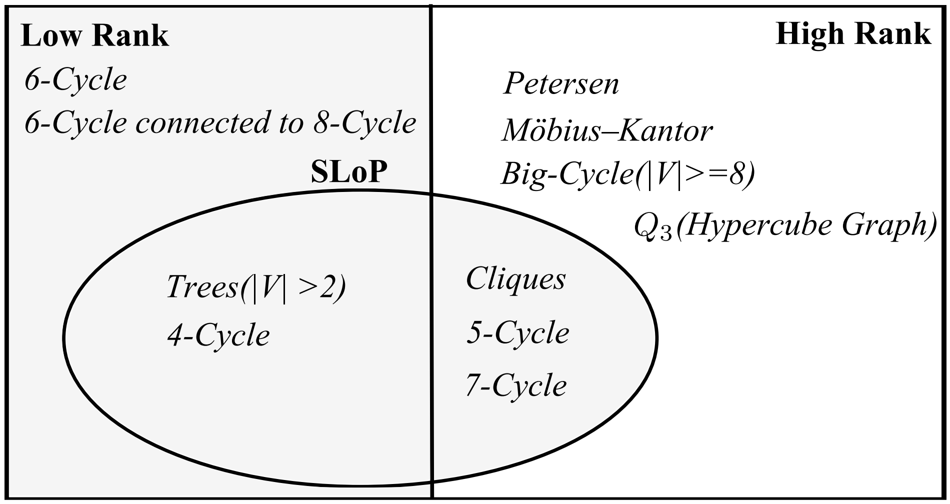

Figure 3-4. Graph examples and classification by the set σ-local pooling factor and rank condition. For graphs $G_S = (S, L)$ inside the oval, $\sigma_S^* = 1$; outside the oval, $\sigma_S^* < 1$. The graph labeled '6-cycle connected to 8-cycle' is shown in Fig. 3-6.

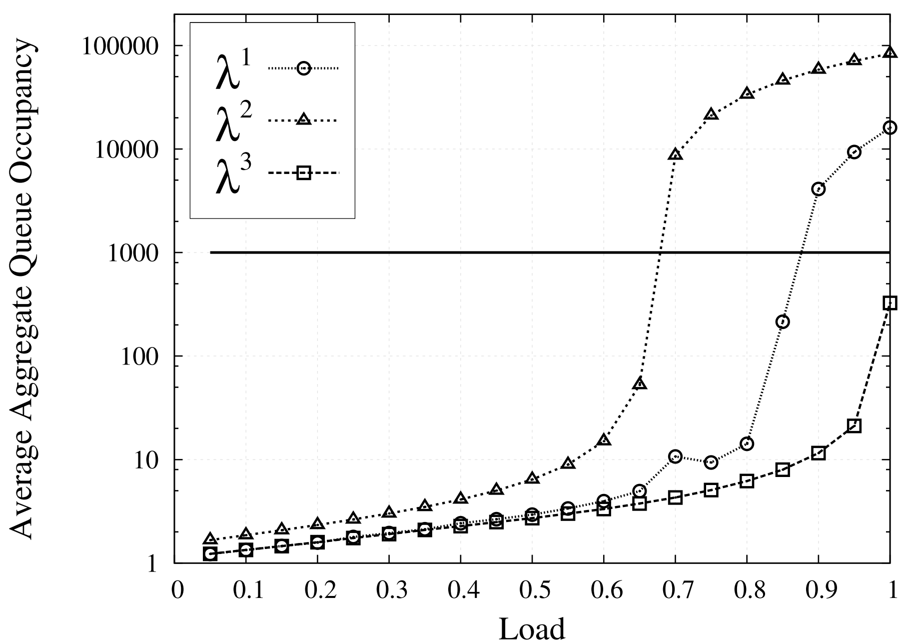

Figure 3-5. Constant arrivals in C_6

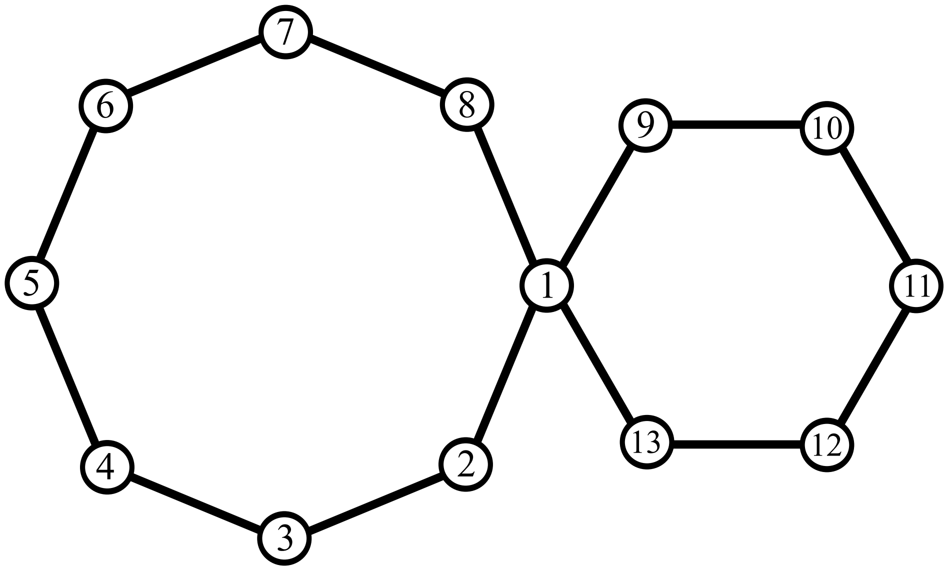

Figure 3-6. An interference graph with C_6 connected to C_8. For this graph, the ranks are $R(M_V) = R(M_V, e) = 12$.

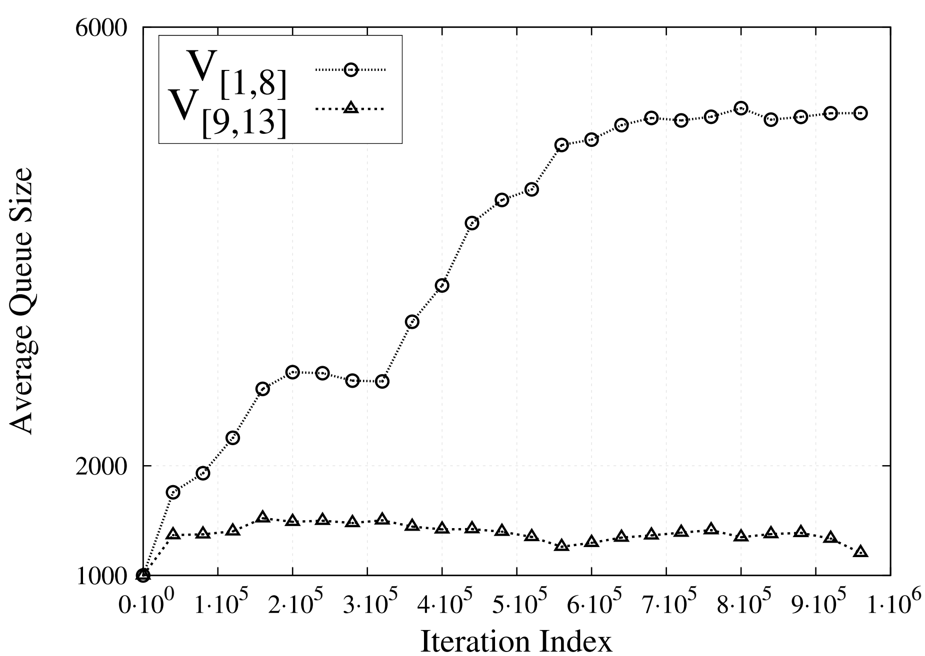

Figure 3-7. Average queue sizes for nodes 1-8 on C_8 (labelled as $V_{[1,8]}$) and for the nodes 9-13 on C_6 ($V_{[9,12]}$).

CHAPTER 4
PERFORMANCE OF LONGEST QUEUE FIRST POLICY IN WIRELESS NETWORK WITH CHANNEL FADING

In this chapter, we investigate the performance of the LQF algorithm in wireless network with channel fading. The time-varying channel states fundamentally change the wireless scheduling problem and the previous result on LQF may not simply apply. We generalize the channel fading model in [29] so that the underlying interference relationship could change over time. We show that the performance characterization of LQF with single parameter F-LPF [29] tends to underestimate the stability region of LQF under channel fading. Hence, we study a multiple-parameter based performance characterization of LQF under channel fading. With the theory builded in this chapter, we capture a larger stability region of LQF under channel fading. Moreover, we introduce a simple algorithm to estimate the performance lower bound of LQF under any given channel fading structure.

The chapter is organized as follows. In Section 4.1, we provide our system model, basic definitions and notations, and describe the link scheduling problem under channel fading. In Section 4.2, we describe the main conclusion for performance characterization of LQF using the new notion of channel fading link σ-local pooling. In Section 4.3, we develop a fuller theory of channel fading link and set σ-local pooling that helps to apply these new concepts. In Section 4.4, we provide methods to estimate or bound the channel fading link and set σ-local pooling factors. Section 4.5 summarizes the chapter.

4.1 Preliminaries and System Model

We assume a time-slotted system, where each slot has a unit length. Let V be the set of all the wireless links in the network. We regard the elements of V as nodes of interference graphs.

In our channel fading model, the network experiences time-varying (global) channel states where each state is represented by an interference graph. Let $\mathcal{S}$ be the set of all possible channel states. We assume $\mathcal{S}$ is a finite set. For each t, let $S(t)$ denote the channel state at time t. We assume $\{S(t)\}_{t\geq 1}$ is i.i.d. with a probability distribution π. The probability that the channel state is S is denoted by $\pi(S)$. Without loss of generality, we assume $\pi(S) > 0$ for all $S \in \mathcal{S}$.

For each $S \in \mathcal{S}$, the interference graph is denoted by $G_S = (V_S, E_S)$, where the node set V_S is a subset of V and the edge set E_S represents the interference relationship under the channel state S. An edge $(u, v) \in E_S$, where $u, v \in V_S$, if and only if the links u and v interference with each other when the channel state is S. In state S, a feasible transmission schedule corresponds to a subset of V_S that is free of interfering pairs. Alternatively speaking, a feasible schedule must be an independent set of the graph G_S. In particular, any link that is not in V_S cannot be activated for transmission. A feasible schedule is said to be *maximal* under channel state S if no more links from V_S can be activated without violating the interference constraints.

To make the scheduling problem well-posed, we assume $\bigcup_{S\in\mathcal{S}} V_S = V$. This way, every link has a chance to be activated in some channel state.

Let a schedule be represented as a $|V|$-dimensional 0-1 vector, where a value 1 in an entry indicates the link is active and 0 indicates otherwise.

We represent all the maximal schedules under the channel state S by a matrix M_V^S. Each column in M_V^S is a vector corresponding to a maximal schedule with respect to G_S. For convenience, we also view M_V^S as the set of all maximal schedules under the channel state S. Let $Co(M_V^S)$ denote the convex hull of the maximal schedules in M_V^S under the channel state S.

For a subset of the links $L \subseteq V$, we define M_L^S to be the matrix (set) of maximal schedules restricted to L under the channel state S, where each column of M_L^S is an $|L|$-dimensional 0-1 vector. Let $Co(M_L^S)$ denote the convex hull of the schedules in M_L^S.

We assume single-hop traffic: The traffic that arrives at a link is transmitted only by that link and it leaves the network after the transmission. We assume the traffic arrival processes to different wireless links are independent to each other. For simplicity, we assume that, for each link, the arrival process is i.i.d. over time. This assumption can be relaxed provided the resulting queue process is Markovian. (See [12] [11] for the reasons.) For each link $l \in V$, the average arrival rate is denoted by λ_l. We also assume the variation of channel state is i.i.d. over time and the average time fraction for each channel state S is $\pi(S)$. According to the strong law of large number, we have the following:

$$\frac{1}{T}\sum_{k=0}^{T}\hat{A}_l(k) \rightarrow \lambda_l \text{ as } T \rightarrow \infty, \tag{4–1}$$

$$\frac{1}{T}\sum_{k=0}^{T}\mathbf{1}_{GS(k)=S} \rightarrow \pi(S) \text{ as } T \rightarrow \infty. \tag{4–2}$$

Here, $\hat{A}_l(k)$ is the instantaneous arrival on link l at slot k and $\mathbf{1}_{GS(k)=S}$ is the identity variable indicates whether the channel state at slot k is S.

4.1.1 Capacity Region

The capacity region Λ of the network under channel fading is defined as,

$$\Lambda = \{\lambda | 0 \leq \lambda \leq \sum_S \pi(S)\mu_S, \text{ where } \mu_S \in Co(M_V^S)\}. \tag{4–3}$$

Roughly speaking, it contains all the arrival rate vectors that can be supported by some scheduling algorithm [41] [14].

We also need the notion of the capacity region for a subset of the links $L \subseteq V$. This region is defined analogously by replacing $Co(M_V^S)$ with $Co(M_L^S)$ in (4–3) and is denoted by Λ_L. We define the *interior* of Λ as follows, which is denoted by Λ^o.[1]

$$\Lambda^o = \{\lambda | 0 \leq \lambda < \sum_S \pi(S)\mu_S, \text{ where } \mu_S \in Co(M_V^S)\}. \tag{4–4}$$

The interior of Λ_L is similarly defined and is denoted by Λ_L^o.

[1] This definition of the interior coincides with the topological notion if the whole space is $\mathbb{R}_+^{|V|}$.

The following fact is well known (see [14]).

Lemma 36. *The capacity region Λ is convex.*

Given a vector $\lambda \in \mathbb{R}^{|V|}$ and a set $L \subseteq V$, the notation $[\lambda]_L$ denotes the $|L|$-dimensional vector given by λ restricted to the entries in L.

For a vector α, the transpose of α is written as α'.

4.1.2 Review of Set, Link and Overall σ-local Pooling

We first review the conventional notions of set σ-local pooling, link σ-local pooling and overall σ-local pooling, which are developed for the standard protocol model [18] [23]. These notions are crucial for understanding the stability region of the LQF algorithm in a wireless network. We will later extend these notions to the channel fading case.

In the standard protocol model (i.e., the non-fading case), there is only one channel state or one interference graph. We therefore can omit the channel state designators in the notations. For instance, the notations M_L and $Co(M_L)$ take the obvious meanings.

Definition 21. *Given a non-empty set of links $L \subseteq V$, the* **set σ-local pooling factor for** *L, denoted by σ^*_L, is given by*

$$\sigma^*_L = \sup\{\sigma \mid \sigma\mu \not> \nu, \text{ for all } \mu, \nu \in Co(M_L)\ \} \tag{4–5}$$

$$= \inf\{\sigma \mid \sigma\mu > \nu, \text{ for some } \mu, \nu \in Co(M_L)\ \}. \tag{4–6}$$

It has been shown that the optimal value of the following problem is equal to the set σ-local pooling factor [23].

$$\min_{\sigma,\mu,\nu} \sigma \tag{4–7}$$

$$\text{subject to } \ \sigma\mu \geq \nu \tag{4–8}$$

$$\mu, \nu \in Co(M_L). \tag{4–9}$$

The link σ-local pooling factor is defined as follows.

Definition 22. *The* σ**-local pooling factor of a link** $l \in V$*, denoted by* σ_l^**, is given by*

$$\begin{aligned}\sigma_l^* =& \sup\{\sigma | \sigma\mu \ngtr \nu \text{ for all } L \subseteq V \text{ such that } l \in L, \text{ and all} \\ & \mu, \nu \in Co(M_L)\} \qquad (4\text{–}10)\\ =& \inf\{\sigma | \sigma\mu > \nu \text{ for some } L \subseteq V \text{ such that } l \in L, \text{ and some} \\ & \mu, \nu \in Co(M_L)\}. \qquad (4\text{–}11)\end{aligned}$$

Comparing the definitions of σ_L^* and σ_l^*, we have

$$\sigma_l^* = \min_{\{L \subseteq V \mid l \in L\}} \sigma_L^*. \qquad (4\text{–}12)$$

Definition 23. *For a network whose interference graph is* $G = (V, E)$*, the* **overall** σ**-local pooling factor***, denoted by* $\sigma^*(G)$*, is defined as*

$$\sigma^*(G) = \min_{L \subseteq V, L \neq \emptyset} \sigma_L^*. \qquad (4\text{–}13)$$

The following lemma can be shown by comparing the definitions involved.

Lemma 37.

$$\sigma^*(G) = \min_{l \in V} \sigma_l^*. \qquad (4\text{–}14)$$

Let the diagonal matrix $\Sigma^*(G)$ be defined by $\Sigma^*(G) = \text{diag}\,(\sigma_l^*)_{l \in V}$. It has been shown that $\sigma^*(G)\Lambda$ and $\Sigma^*(G)\Lambda$ are both regions of stability under LQF [18] [23], with the latter containing the former.

A model for fading channels that is similar to ours is considered in [29]. In that model, there is a single interference graph for all the channel states. However, under each channel state, a different subset of the links are capable of transmission. Hence, a channel state can be represented by the subset $J \subseteq V$ containing those links that can transmit. Again, let $\mathcal{S}$ be the set of all channel states. Given $L \subseteq V$ and $J \in \mathcal{S}$, the maximal schedules under the channel state J restricted to the set L is denoted by $M_{J,L}$, which is a submatrix of M_V by selecting only the rows corresponding to the intersection of J and L and then removing repeated columns. The authors extend the set σ-local pooling factor to their fading model. For $L \subseteq V$,

$$\sigma_L^* = \inf\{\sigma \mid \exists \mu, \nu \in \Phi(L) \text{ such that } \sigma\mu \geq \nu\}, \tag{4–15}$$

where,

$$\Phi(L) = \{\sum_{J \in \mathcal{S}} \pi(J)\xi_J \mid \xi_J \in Co(M_{J,L}) \text{ for all } J \in \mathcal{S}\}. \tag{4–16}$$

In this fading model, the overall σ-local pooling factor of the network with the interference graph $G = (V, E)$ is defined as

$$\sigma^*(G) = \min_{L \in \mathcal{S}} \sigma_L^*. \tag{4–17}$$

4.2 Link σ Local Pooling and the Performance of LQF

In this section, we introduce the notion of *link* σ-local pooling under channel fading. We show how to capture a larger stability region of LQF using this new notion.

4.2.1 Motivation for Defining Link σ-Local Pooling Factor

In the case of $\sigma^*(G) < 1$, the region $\sigma^*(G)\Lambda$ is the result of shrinking the capacity region by the same scaling factor on all the dimensions. This can be overly conservative when the links face heterogenous interference constraints; in that case, using a single factor to characterize the performance will underestimate the achievable rates at some links. To better characterize the stability region of LQF, the reduction factors should be non-uniform across the links.

Consider the example in Fig.4-1 under the 1-hop interference model. There are three channel states: S_1, S_2 and S_3, with the probabilities $\pi(S_1) = \pi(S_2) = \pi(S_3) = 1/3$. Fig.4-1 shows the three interference graphs G_{S_1}, G_{S_2} and G_{S_3}. The graph G is the union of the three interference graphs. Let the node set L be $L = \{1, 2, 3\}$. The three interference graphs, when restricted to L, are identical to an example in [29]. The set σ-local pooling fact $\sigma_L^* = 3/4$. Hence, we have $\sigma(G)^* \leq \sigma_L^* = 3/4$. In this example, we can see that although the interference graph under any channel state satisfies the so-called local pooling condition, LQF cannot achieve the full capacity region due to the effect of channel fading.

However, for claiming stability, it is unnecessary to reduce the dimension corresponding to node 4 by 3/4. To see this, note that for any arrival rate vector λ inside the capacity region, we have $\lambda_2+\lambda_4 < 1$. This is so because, in every channel state, node 2 and node 4 interference with each other and at most one of them can be activated in any maximal schedule. Furthermore, if queue 4 (i.e., the queue at node 4) is non-empty, then exactly one node from nodes 2 and 4 is activated, and hence, the total queue length of the two nodes must decrease. This decrease will continue until queue 4 becomes empty. Hence, the condition $\lambda_2 + \lambda_4 < 1$ ensures queue 4 is stable. We conclude that for any $\lambda \in \Lambda^o$, queue 4 is stable; reducing λ_4 by multiplying a factor of $3/4$ will understate the stability region.

The reason why links have different performance is that the efficiency ratio is related with the distribution of channel states and the corresponding topologies. They together will determine certain “bottleneck”, for each link, formed by subset of links containing the link in question. For different link, such subset of links are different. For instance, the ”bottleneck” subset for link 2 in Fig. 4-1 contains 3 links (including link 1, 2, and 3), which leads its efficiency ratio to be $3/4$. The ”bottleneck” subset for link 4 is link 4 itself, which leads its efficiency ratio to be 1. We will later identify this kind of subgraphs more explicitly.

4.2.2 Definitions of Link σ-Local Pooling under Channel Fading

The above discussion motivates us to come up with a performance characterization for each link individually.

Definition 24. *For any set $L \subseteq V$, the* **convex hull of maximal schedules associated with the set** *L under channel fading is:*

$$\Phi(L) = \{\sum_S \pi(S)\xi_S \mid \xi_S \in Co(M_L^S) \text{ for all } S \in \mathcal{S}\}.$$

Definition 25. *Under channel fading, the* **σ-local pooling factor of a link** *$l \in V$, denoted by σ_l^*, is*

$$\begin{aligned}\sigma_l^* = &\sup\{\sigma | \sigma\mu \not> \nu \text{ for all } L \subseteq V \text{ such that } l \in L, \\ &\text{and all } \mu, \nu \in \Phi(L)\} \qquad (4\text{–}18)\\ = &\inf\{\sigma | \sigma\mu > \nu \text{ for some } L \subseteq V \text{ such that } l \in L, \\ &\text{and some } \mu, \nu \in \Phi(L)\}. \qquad (4\text{–}19)\end{aligned}$$

Definition 26. *Given a non-empty set of links $L \subseteq V$, the* **set σ-local pooling factor for** *L under channel fading, denoted by σ_L^*, is given by*

$$\sigma_L^* = \sup\{\sigma \mid \sigma\mu \not> \nu, \text{ for all } \mu, \nu \in \Phi(L)\ \} \qquad (4\text{–}20)$$

$$= \inf\{\sigma \mid \sigma\mu > \nu, \text{ for some } \mu, \nu \in \Phi(L)\ \}. \qquad (4\text{–}21)$$

Under channel fading, the *overall σ-local pooling factor*, $\sigma^*(G)$, is defined in the same way as in Definition 23.

The definitions of set and overall σ-local pooling factors under our fading model naturally extend those in [29]. Furthermore, Lemma 37 still holds.

4.2.3 Link-Based Performance Guarantee of LQF under Fading

In [29], LQF is found to stabilize the region $\sigma^*(G)\Lambda$ under a special class of channel fading. Lemma 37 makes it clear that the overall σ-local pooling factor $\sigma^*(G)$ is equal to the smallest of all the link σ-local pooling factors. We now define a $|V| \times |V|$ diagonal matrix, $\Sigma^*(G)$, whose diagonal entries are σ_l^* for $l \in V$. Clearly, $\sigma^*(G)\Lambda \subseteq \Sigma^*(G)\Lambda$ The next theorem says that LQF can stabilize $\Sigma^*(G)\Lambda$. Hence, we have found better performance guarantee for LQF under more general channel fading.

Theorem 4.1. *For any arrival rate vector $\lambda \in \Sigma^*(G)\Lambda$, the network is stable under the LQF policy.*

Proof. We will consider the fluid limit of the queue process, denoted by $\{q_l(t)\}_{t\geq 0}$, for all $l \in V$. (See [11] [12] [18] [29] for more details on this approach.) Consider a fixed time instance t. Let L be the set of those longest queues (with equal length) whose time derivatives at t, $\dot{q}_l(t)$, are the largest (also identical) under the given LQF policy (that is, an instance of the LQF policy that is being used.). The queues in L will remain the longest with identical length in the next infinitesimally small time interval.

Let $l \in \text{argmax}_{k\in L}\sigma_k^*$. Since $\lambda \in \Sigma^*(G)\Lambda^o$, there exists $\mu \in \Lambda^o$ such that $\lambda < \Sigma^*(G)\mu$. This implies $\lambda_k < \sigma_k^*\mu_k$, for all $k \in L$. Hence, $[\lambda]_L < \sigma_l^*[\mu]_L$.

It is easy to see that $[\mu]_L \in \Lambda_L$. Hence, there exists $\mu_L \in \Phi(L)$ such that $[\mu]_L \leq \mu_L$. Then, $[\lambda]_L < \sigma_l^*\mu_L$. Given λ, let us suppose the way of picking such a μ_L is well-defined. Let $\epsilon_L = \min_{k\in L}(\sigma_l^*\mu_L(k) - \lambda(k))$.[2] We have $\epsilon_L > 0$.

For such fixed λ and μ_L, consider any arbitrary $\nu_L \in \Phi(L)$. We must have $\sigma_l^*\mu_L \not> \nu_L$ by the definition of the link σ-local pooling factor. Hence, there exists a link $k \in L$ such that $\sigma_l^*\mu_L(k) \leq \nu_L(k)$. For such a k, since $\lambda(k) < \sigma_l^*\mu_L(k) \leq \nu_L(k)$, we have $\nu_L(k) - \lambda(k) \geq \epsilon_L$. Hence, $\max_{k\in L}(\nu_L(k) - \lambda(k)) \geq \epsilon_L$. Note that ϵ_L is independent of ν_L.

The service rate vector being used, when restricted to L, must belong to the set $\Phi(L)$. Roughly, this is because L contains all the queues that are among the longest and remain longest in the near future, and hence, every LQF schedule being used must be a maximal schedule when restricted to L. (See [11] [12] for a more rigorous argument for this.)

Now imagine ν_L is the service rate vector at the current time t. We have just shown that, for some $k \in L$, $\nu_L(k) - \lambda(k)$ is at least ϵ_L. Hence, the queue at link k decreases

[2] Given a vector ν, we write its component corresponding to link k by ν_k or $\nu(k)$ interchangeably.

at a rate no less than ϵ_L. Since the queues in the set L change at the same rate, they all decrease at a rate no less than ϵ_L. Hence, each of the longest queues decreases its size at a rate no less than ϵ_L. Let $\epsilon = \min\{\epsilon_L | L \subseteq V\}$. Since the number of possible subsets of V is finite, we have $\epsilon > 0$. Hence, at any time instance, each of the longest queues decreases at a positive rate no less than ϵ. By [11], this is sufficient to conclude that the original queueing process is a positive recurrent Markov process, which means the queues are stable by definition. □

As an example, for the fading situation in Fig. 4-1, $\sigma^*(G) = 3/4$ whereas $\Sigma^*(G) = \text{diag}(3/4, 3/4, 3/4, 1)$.[3]

4.3 Theory of Link σ-Local Pooling

In this section, we show some important properties of the newly defined concepts of link and set σ-local pooling and the related concept of *limiting set*. In general, these concepts are difficult to work with since their definitions involve combinatorial enumerations. Our objective is to provide tools for using or applying these concepts. As will be shown later, some of the theories developed in this section can help in the estimation of the link σ-local pooling factors.

4.3.1 Set σ-Local Pooling Factor

The concept of the *set* σ-local pooling factor (Definition 26) is a building block for performance characterization of the LQF policy. We will first study it in some detail.

Suppose $L \subseteq V$ and L is non-empty. For convenience, let

$$\Theta_L = \{\sigma \mid \sigma\mu \ngeq \nu, \text{for all } \mu, \nu \in \Phi(L)\ \}, \tag{4–22}$$

[3] This may not be obvious now, but can be shown easily by applying the theory to be developed subsequently.

and let compliment of Θ_L

$$\Theta_L^c = \{\sigma \mid \sigma\mu > \nu, \text{for some } \mu, \nu \in \Phi(L)\ \}. \tag{4–23}$$

Then, the set σ-local pooling factor for L is

$$\sigma_L^* = \sup\{\sigma \mid \sigma \in \Theta_L\} = \inf\{\sigma \mid \sigma \in \Theta_L^c\}. \tag{4–24}$$

The following are some elementary facts. Since $0 \notin \Phi(L)$, by (4–23), $\sigma_L^* > 0$. By considering $\mu = \nu$ in (4–22), we see that $\sigma_L^* \leq 1$. If $\sigma \in \Theta_L^c$, then $(\sigma - \epsilon)\mu > \nu$ for small enough $\epsilon > 0$, where L, μ and ν are as in the definition of Θ_L^c. Hence, Θ_L^c is an open set on $\mathbb{R}$. In fact, $\Theta_L^c = (\sigma_L^*, \infty)$; $\Theta_L = [0, \sigma_L^*]$.

The following lemma says that σ_L^* can be found by a well-defined optimization problem where the constraint region is a closed set. The constraint region can be thought as being compact when taking into account the fact that σ_L^* is bounded from above by 1. The fact in the lemma needs to be explicitly stated since the infimum-based definition in (4–24) is not over a closed set in variables (σ, μ, ν).

Lemma 38. *For any non-empty $L \subseteq V$, σ_L^* is the optimal value of the following optimization problem.*

$$\textbf{\textit{(I)}} \min_{\sigma,\mu,\nu} \quad \sigma \tag{4–25}$$

$$\textit{subject to} \;\; \sigma\mu \geq \nu \tag{4–26}$$

$$\mu, \nu \in \Phi(L) \tag{4–27}$$

$$\Phi(L) = \sum_S \pi(S) Co(M_L^S). \tag{4–28}$$

Proof. By (4–23) and (4–24), $\sigma_L^*(\pi)$ is the optimal value of the following problem.

$$\textbf{(II)} \inf_{\sigma,\mu,\nu} \sigma \tag{4–29}$$

$$\text{subject to } \sigma\mu > \nu \tag{4–30}$$

$$\mu, \nu \in \Phi(L) \tag{4–31}$$

$$\Phi(L) = \sum_S \pi(S) Co(M_L^S). \tag{4–32}$$

Let the constraint sets of the optimization problems (I) and (II) be denoted by Y_1 and Y_2 ($Y_2 = \Theta_L^c$), respectively, both of which lie in $\mathbb{R} \times \mathbb{R}^{|L|} \times \mathbb{R}^{|L|}$. We will show that Y_1 is the closure of Y_2. For this, we need to show that every point in $Y_1 \backslash Y_2$ is a limit point of Y_2. Let a point $(\sigma, \mu, \nu) \in Y_1 \backslash Y_2$, which is characterized by $\sigma\mu \geq \nu$ with $\sigma\mu_k = \nu_k$ for some $k \in L$. If $\mu > 0$, we only need to increase σ by a little bit to find a point in Y_2. To handle the general case where $\mu_k = 0$ for some $k \in L$, note that $\nu_k = 0$. Since M_L^S includes all maximal vectors under channel state S and $\bigcup_{S \in \mathcal{S}} V_S = V$, there exists $\omega \in \Phi(L)$ such that $\omega > 0$. Then, a vector $\hat{\mu} = (1 - \epsilon_1)\mu + \epsilon_1\omega$, where $0 < \epsilon_1 \leq 1$, has the property that $\hat{\mu} > 0$. Note that $\hat{\mu} \in \Phi(L)$. We can choose ϵ_1 small enough and choose $\epsilon_2 > 0$ accordingly such that $(\sigma + \epsilon_2)\hat{\mu} > \nu$ and also $(\sigma + \epsilon_2, \hat{\mu}, \nu)$ is in the ϵ-open ball around (σ, μ, ν). Hence, Y_1 is the closure of Y_2.

Next, $(\sigma, \mu, \nu) \mapsto \sigma$ is a continuous function on $\mathbb{R} \times \mathbb{R}^{|L|} \times \mathbb{R}^{|L|}$. Hence, the two problems have the same optimal value and the optimum is attained for problem (I). $\square$

Since the optimization problem (I) has a continuous objective function and the constraint set is closed, the optimum is attained in its constraint set. The optimization problem (II) is not attained.

The optimization characterization of σ_L^* is useful since one can apply the duality theory to derive important results and insights. The problem (I) has an alternative form, which is a linear program. From the linear program, we can obtain the following dual problem. Suppose M_L^S has $c(L, S)$ columns. Let e_n be the vector $(1, 1, \dots, 1)'$ with n 1's.

Lemma 39. *σ_L^* is the optimal value of the following optimization problem.*

(Dual)

$$\max_{x \geq 0, (y_S)_{S\in\mathcal{S}}, (w_S)_{S\in\mathcal{S}}} \sum_S w_S$$

$$\textit{subject to } \pi(S)\, x' M_L^S \leq y_S e'_{c(L,S)}, \forall S \in \mathcal{S}$$

$$\pi(S) x' M_L^S \geq w_S e'_{c(L,S)}, \forall S \in \mathcal{S}$$

$$\sum_S y_S = 1,$$

where $x \in \mathbb{R}_+^{|L|}$, $y_S, w_S \in \mathbb{R}$ for each $S \in \mathcal{S}$.

Proof. The problem (Dual) is the dual problem of a linear version of the problem (I). For every channel state $S \in \mathcal{S}$, let α_S and β_S be non-negative vectors of dimension $c(L, S)$. Problem (I) can be written as follows.

$$\min_{\sigma, (\alpha_S)_{S\in\mathcal{S}}, (\beta_S)_{S\in\mathcal{S}}} \sigma$$

$$\text{subject to } \sigma \sum_S \pi(S) M_L^S \alpha_S \geq \sum_S \pi(S) M_L^S \beta_S$$

$$\alpha'_S e_{c(L,S)} = 1, \forall S \in \mathcal{S}$$

$$\beta'_S e_{c(L,S)} = 1, \forall S \in \mathcal{S}$$

$$\alpha_S, \beta_S \geq 0, \forall S \in \mathcal{S}.$$

Let $\gamma_S = \sigma \alpha_S$ for every state S. The problem can be further written as,

$$\min_{\sigma, (\gamma_S)_{S\in\mathcal{S}}, (\beta_S)_{S\in\mathcal{S}}} \sigma \tag{4–33}$$

$$\text{subject to } \sum_S \pi(S) M_L^S \gamma_S \geq \sum_S \pi(S) M_L^S \beta_S \tag{4–34}$$

$$\gamma'_S e_{c(L,S)} = \sigma, \forall S \in \mathcal{S} \tag{4–35}$$

$$\beta'_S e_{c(L,S)} = 1, \forall S \in \mathcal{S} \tag{4–36}$$

$$\gamma_S, \beta_S \geq 0, \forall S \in \mathcal{S}. \tag{4–37}$$

Let $x \in \mathbb{R}_+^{|L|}$, $(y_S)_{S\in\mathcal{S}}$, $(z_S)_{S\in\mathcal{S}}$ to be dual variables associated with (4–34), (4–35) and (4–36), respectively. Then, the dual problem is

$$\max_{x\geq 0,(y_S)_{S\in\mathcal{S}},(z_S)_{S\in\mathcal{S}}} \quad -\sum_S z_S$$

$$\text{subject to} \quad \pi(S)x'M_L^S + z_S e'_{c(L,S)} \geq 0, \forall S \in \mathcal{S}$$

$$y_S e'_{c(L,S)} - \pi(S)x'M_L^S \geq 0, \forall S \in \mathcal{S}$$

$$\sum_S y_S = 1.$$

Let $w_S = -z_S$, we get the optimization problem in the lemma. □

4.3.2 Relation between Set and Link σ-Local Pooling

Lemma 40. *For a link $l \in V$, σ_l^* is the smallest σ_L^* for all $L \subseteq V$ that contains l, i.e.,*

$$\sigma_l^* = \min_{\{L\subseteq V \mid l\in L\}} \sigma_L^* \tag{4–38}$$

Proof. The proof is by the definitions of σ_l^* and σ_L^*. □

Corollary 5. *Let $L \subseteq V$ be an arbitrary non-empty set. For all $l \in L$, $\sigma_L^* \geq \sigma_l^*$.*

4.3.3 Limiting Set

A limiting set for a link l is a subset of the links, $L \subseteq V$ with $l \in L$, that achieves σ_l^* (see Lemma 40 or (4–19)). The significance of a limiting set is that it is the set of links whose interference with l prevents σ_l^* from becoming larger, hence, the term *limiting*. Therefore, it is the limiting set for a link, instead of the complete network, that represents structural constraints for the link. While the network can be large, the limiting set for a link may contain a much smaller number of links. Hence, finding the limiting set and understanding its properties have both theoretical and practical significance.

Definition 27. *For any link $l \in V$, a set $L \subseteq V$ is called a* **limiting set for link** *l if $l \in L$ and there exist $\mu, \nu \in \Phi(L)$ such that $\sigma_l^* \mu \geq \nu$.*

Lemma 41. *For any link l, a limiting set for l exists.*

Proof. The proof is omitted for brevity. $\square$

Note that the limiting set for a link is not necessarily unique.

Lemma 42. *A set $L \subseteq V$ containing link l is a limiting set for l if and only if $\sigma_L^* = \sigma_l^*$.*

Proof. Suppose L is a limiting set for l. Then, there exist $\mu, \nu \in \Phi(L)$ such that $\sigma_l^* \mu \geq \nu$. By Lemma 38, $\sigma_L^* \leq \sigma_l^*$. Combining this with Corollary 5, we have $\sigma_L^* = \sigma_l^*$.

Conversely, suppose $\sigma_L^* = \sigma_l^*$. Then, by Lemma 38, there exist $\mu, \nu \in \Phi(L)$ such that $\sigma_l^* \mu \geq \nu$. By the definition, L is a limiting set for l. $\square$

Corollary 6. *Given a non-empty set of links $L \subseteq V$, if $\sigma_L^* = \max_{l \in L} \sigma_l^*$, then L is a limiting set for each link in the set $\arg\max_{l \in L} \sigma_l^*$.*

For a link l with $\sigma_l^* = 1$, any set L containing l is a limiting set for l, since we can choose $\mu = \nu$ in Definition 27. Hence, when $\sigma_l^* = 1$, the notion of a limiting set is trivial, and the corresponding limiting sets are called *trivial*. Only when $\sigma_l^* < 1$, the notion of a limiting set is consequential.

Let $E = \bigcup_S E_S$ and let G be the graph $G = (V, E)$.

Definition 28. *We say a set of nodes are* **weakly connected** *if they are connected in the graph G. We say two nodes are* **neighbors** *of each other if they are neighbors in G. Equivalently, two nodes are neighbors of each other if they are neighbors in G_S for some channel state S.*

For a node $i \in V$, the neighborhood of i is denoted by $N(i)$, which is the set of all neighbors of node i.

Lemma 43. *For any link $l \in V$, one of its limiting sets is weakly connected.*

Proof. Suppose L is an arbitrary limiting set for l. If L is weakly connected, then there is nothing to prove. Otherwise, let $L' \subseteq L$ with $l \in L'$ be the largest subset of L that contains l and is weakly connected. We only need to show that L' is also a limiting set for l.

Since L is a limiting set for link l, by Lemma 42, we must have $\sigma_l^* = \sigma_L^*$. From Lemma 38, we know there are two vectors $\mu, \nu \in \Phi(L)$ such that $\sigma_L^* \mu \geq \nu$. Since L' is the largest weakly connected subset of L containing l, any link in L' does not interfere with any link in $L \setminus L'$ under any channel state. Thus, under any channel state S, a schedule that is maximal with respect to L (i.e., the corresponding vector is in M_L^S) is also maximal with respect to the L' (i.e., the vector, when restricted to L', is in $M_{L'}^S$). Hence, $[\mu]_{L'}, [\nu]_{L'} \in \Phi(L')$ and $\sigma_l^*[\mu]_{L'} \geq [\nu]_{L'}$. Therefore, by Lemma 38, $\sigma_l^* \geq \sigma_{L'}^*$. Also, since $l \in L'$, we have $\sigma_l^* \leq \sigma_{L'}^*$ by Corollary 5. Hence, $\sigma_l^* = \sigma_{L'}^*$. According to Lemma 42, L' is a limiting set for l. □

Remark: Lemma 43 suggests that, to find a limiting set for a link l, only those weakly connected subsets containing l need to be inspected.

4.3.4 Performance Guarantees of LQF - A Revisit

With the development of set σ-local pooling, we can state the following sufficient condition for stability under LQF under channel fading.

Theorem 4.2. *Suppose the arrival rate vector λ satisfies the condition that, for every non-empty $L \subseteq V$, $[\lambda]_L \in \sigma_L^* \Lambda_L^o$. Then, the network is stable under the LQF policy.*

Proof. Since the proof is similar to that for Theorem 4.1, we will be brief and omit some arguments, which can be found in the proof for Theorem 4.1. We will consider the fluid limit of the queue process, denoted by $\{q_l(t)\}_{t \geq 0}$, for all $l \in V$. Consider a fixed time instance t. Let L be the set of those longest queues (with equal length) whose time derivatives at t, $\dot{q}_l(t)$, are the largest (also identical) under the particular LQF policy being used.

By the assumption of the theorem, there exists $\mu_L \in \Phi(L)$ such that $[\lambda]_L < \sigma_L^* \mu_L$. For this μ_L and any other $\nu_L \in \Phi(L)$, $\sigma_L^* \mu_L \not\geq \nu_L$ by the definition of σ_L^*. Hence, there exists a link $k \in L$ such that $\sigma_L^* \mu_k \leq \nu_k$. Then, $\lambda_k < \nu_k$. If ν_L is the service rate vector (in the fluid limit) for the queues in L, the queue at link k decreases at the rate $\nu_k - \lambda_k$. Since all queues in the set L change at the same rate, they all decrease at the rate $\nu_k - \lambda_k$, which is positive. □

With the relationship between set and link σ-local pooling, we can show Theorem 4.1 is implied by Theorem 4.2. Hence, the condition of Theorem 4.2 for stability under LQF is more general than that of Theorem 4.1. This shows one of the utilities provided by our theoretical development of set and link σ-local pooling.

Proof. (Alternative Proof of Theorem 4.1) Consider any link set $L \subseteq V$. Let $l \in \text{argmax}_{k \in L} \sigma_k^*$. Since $\lambda \in \Sigma^*(G)\Lambda^o$, there exists $\mu \in \Phi(V)$ such that $\lambda < \Sigma^*(G)\mu$. This implies $[\lambda]_L < \Sigma_L^*[\mu]_L$, where Σ_L^* denotes the restriction of $\Sigma^*(G)$ to L, i.e., the submatrix of $\Sigma^*(G)$ with only the rows and columns corresponding to the set L. Hence, $[\lambda]_L < \sigma_l^*[\mu]_L \leq \sigma_L^*[\mu]_L$, where we have used Corollary 5 in the second inequality. It is easy to see that there exists $\hat{\mu}_L \in \Phi(L)$ such that $[\mu]_L \leq \hat{\mu}_L$. Hence, $[\lambda]_L \in \sigma_L^*\Lambda_L^o$. By Theorem 4.2, the queues are stable under LQF. □

4.4 Estimating $\Sigma^*(G)$ Matrix

4.4.1 Estimating Set σ-Local Pooling Factor

In Section 4.3.1, we introduced a linear programming formulation (LP), (4–33)-(4–37), for calculating the σ-local pooling factor for a set of links. Although linear programs can be solved in polynomial time in terms of the problem size, our formulation contains exponentially many decision variables and is computationally intractable for large networks. This section concentrates on providing methods to estimate a set σ-local pooling factor.

Consider the dual problem in Lemma 39. If we let $y_S = \pi(S)$ for every state S, the optimal value under this additional constraint will serve as a lower bound for the set σ-local pooling factor. The optimization problem under the additional constraint is as follows.

$$\textbf{(III)} \quad \max_{x \geq 0, (w_S)_{S \in \mathcal{S}}} \sum_S w_S \tag{4–39}$$

$$\text{subject to} \quad x' M_L^S \leq e'_{c(L,S)}, \forall S \in \mathcal{S} \tag{4–40}$$

$$\pi(S) x' M_L^S \geq w_S e'_{c(L,S)}, \forall S \in \mathcal{S}. \tag{4–41}$$

Consider a fixed set $L \subseteq V$. Let $\{m_1^S, m_2^S, \ldots, m_{c(L,S)}^S\}$ represent all the maximal schedules with respect to the set L under channel state S; that is, each m_i^S is the i^{th} column of the matrix M_L^S. Consequently, the problem III can be rewritten as follows.

$$\max_{x \geq 0, (w_S)_{S \in \mathcal{S}}} \sum \pi(S) w_S \tag{4–42}$$

$$\text{subject to} \quad \max_i x' m_i^S \leq 1, \forall S \in \mathcal{S} \tag{4–43}$$

$$\min_i x' m_i^S \geq w_S, \forall S \in \mathcal{S}. \tag{4–44}$$

Observe that the problem (4–42)-(4–44) is a weight assignment problem on the nodes, where the weight vector is x. We define *the weight of a schedule* to be the sum of the weights of all the active nodes in the schedule. The constraint (4–43) implies that, for every channel state, the maximum weight of any schedule is bounded from above by 1. The constraint (4–44) indicates that, for every channel state, the minimum weight of any schedule is bounded from below by w_S.

It is easy to see that, in an optimal solution to the problem (4–42)-(4–44), denoted by $(w_S^*, S \in \mathcal{S}; x^*)$, equality is achieved in (4–44) by some schedules under every channel state, and equality is achieved in (4–43) by some schedules under some channel state. Otherwise, the objective value can be further improved. Hence, in an optimal solution, the weight of any maximum-weight schedule over all channel states is 1. This can be interpreted as that the weight assignment needs to be normalized according to the maximum schedule weight. The weight of any minimum-weight schedule under the channel state S is w_S^*. With some thought, the weight assignment problem (4–42)-(4–44) can be reformulated as finding the node weights to maximize the expectation of the normalized minimum schedule weights, where the expectation is taken over all the channel states. That is,[4]

$$\sigma_L^* \geq \sum_S \pi(S) w_S^* = \max_{x \geq 0} \sum_S \pi(S) \frac{\min_i x' m_i^s}{\max_{i,S} x' m_i^s}. \tag{4–45}$$

The new formulation in (4–45) provides a simple way to derive a lower bound for σ_L^*, which is by assigning some particular weights to the nodes and calculating the ratio as in (4–45). Next, we will use this idea to derive lower-bounds for σ_L^*. We denote the component-wise sum of a vector m by $||m||_1$, which is the 1-norm of m. Then, we have the following.

Lemma 44. *For a non-empty set $L \subseteq V$, we have*

$$\sigma_L^* \geq \sum_S \pi(S) \frac{\min_i ||m_i^S||_1}{\max_{i,S} ||m_i^S||_1}. \tag{4–46}$$

Proof. In (4–45), we assign identical weights to all nodes in L, i.e, $x_j = 1$ for all $j \in L$. $\square$

To improve the lower bound, we can extend the weight assignment approach to the subsets of L.

Lemma 45. *For a non-empty set $L \subseteq V$, we have*

$$\sigma_L^* \geq \max_{L' \subseteq L} \sum_S \pi(S) \frac{\min_i ||[m_i^S]_{L'}||_1}{\max_{i,S} ||[m_i^S]_{L'}||_1}. \tag{4–47}$$

Proof. Assign $x_j = 1$ for all nodes $j \in L'$ and $x_j = 0$ otherwise. $\square$

We next show how to estimate σ_L^* using the primal formulation, instead of the dual formulation.

[4] We assume the convention $\frac{0}{0} = 0$ so that $x = 0$ is not optimal.

Lemma 46. *For a non-empty set $L \subseteq V$, we have*

$$\sigma_L^* \geq \frac{\sum_S \pi(S) \min_i ||m_i^S||_1}{\sum_S \pi(S) \max_i ||m_i^S||_1}. \tag{4–48}$$

Proof. Consider the constraint (4–26) in problem **I**. Suppose we add the inequality over all components of the vectors μ and ν, we get

$$\sigma(\sum_{i \in L} \mu(i)) \geq \sum_{i \in L} \nu(i).$$

Write $\mu = \sum_S \pi(S)\xi^S$ and $\nu = \sum_S \pi(S)\hat{\xi}^S$, where $\xi^S, \hat{\xi}^S \in Co(M_L^S)$ for each S. We have

$$\sigma(\sum_{i \in L} \sum_S \pi(S)\xi^S(i)) \geq \sum_{i \in L} \sum_S \pi(S)\hat{\xi}^S(i),$$

which is the same as

$$\sigma(\sum_S \pi(S) \sum_{i \in L} \xi^S(i)) \geq \sum_S \pi(S) \sum_{i \in L} \hat{\xi}^S(i).$$

We also have

$$\min_i ||m_i^S||_1 \leq \sum_{i \in L} \xi^S(i) \leq \max_i ||m_i^S||_1$$
$$\min_i ||m_i^S||_1 \leq \sum_{i \in L} \hat{\xi}^S(i) \leq \max_i ||m_i^S||_1.$$

Hence,

$$\begin{aligned} \sigma_L^* &\geq \frac{\sum_S \pi(S) \sum_{i \in L} \hat{\xi}^S(i)}{\sum_S \pi(S) \sum_{i \in L} \xi^S(i)} \\ &\geq \frac{\sum_S \pi(S) \min_i ||m_i^S||_1}{\sum_S \pi(S) \max_i ||m_i^S||_1}. \end{aligned}$$

□

Remark: A similar result is also given in [29].

The lower bound in Lemma 46 is no less than the lower bound in Lemma 44, because

$$\frac{\sum_S \pi(S)\min_i ||m_i^S||_1}{\sum_S \pi(S)\max_i ||m_i^S||_1} \geq \frac{\sum_S \pi(S)\min_i ||m_i^S||_1}{\sum_S \pi(S)\max_{i,S} ||m_i^S||_1}$$
$$= \sum_S \pi(S)\frac{\min_i ||m_i^S||_1}{\max_{i,S} ||m_i^S||_1}.$$

To improve the lower bound or to simplify the calculation, we also have a subset version of Lemma 46 as follows:

Lemma 47. *For a non-empty set $L \subseteq V$, we have*

$$\sigma_L^* \geq \max_{L' \subseteq L} \frac{\sum_S \pi(S)\min_i ||[m_i^S]_{L'}||_1}{\sum_S \pi(S)\max_i ||[m_i^S]_{L'}||_1}. \tag{4–49}$$

This lemma can be simply proved by similar scheme in the proof of lemma 46. Furthermore, in case there is only one channel state(i.e. no channel fading), the lemma 46 and 47 are equivalent to the lemma 14 and lemma 17 in [23] correspondingly.

4.4.2 Estimating Link σ-Local Pooling Factors

By Lemma 42, the σ-local pooling factor for a link is equal to the σ-local pooling factor of its limiting set; by Lemma 43, there is a weakly connected limiting set. In this part, we focus on deriving lower bounds for link σ-local pooling factors based on these facts.

Definition 29. *The* **interference degree of node** i **under channel state** S*, denoted by* $d^S(i)$*, is equal to the maximum number of nodes in the neighborhood of* i *that can be simultaneously activated under channel state* S*.*

Let $d_L^S(i)$ be the interference degree of node i restricted to the set L under the channel state S.

Let $I(S,i)$ be defined by $I(S,i) = \mathbf{1}_{V_S}(i)$, for each $S \in \mathcal{S}$ and $i \in V$, where $\mathbf{1}_{V_S}(i)$ is an indicator function.

Lemma 48. *For any non-empty set $L \subseteq V$, we have*

$$\sigma_L^* \geq \max_{i \in L} \frac{\sum_S \pi(S) I(S,i)}{\sum_S \pi(S) d_L^S(i)}. \tag{4–50}$$

The lemma is true by applying Lemma 47 and let $L' = N(i) \cap L$ for each i.

We present an algorithm (Algorithm 2), which finds a separate lower bound for each link σ-local pooling factor based on the interference degrees.

Algorithm 2 Bounds for Link σ-Local Pooling Factors

1: INPUT: A set of interference graphs $G_S = (V_S, E_S)$ for all $S \in \mathcal{S}$, and π
2: OUTPUT: Lower bounds for link σ-local pooling factors for all links, $(\sigma_l)_{l \in V}$.
3: Initialization: $L_1 \leftarrow V, \sigma \leftarrow 1$
4: **for all** $1 \leq i \leq |V|$ **do**
5:     Choose a node $l \in \arg\max_{k \in L_i} \frac{\sum_S \pi(S) I(S,k)}{\sum_S \pi(S) d_{L_i}^S(k)}$
6:     **if** $\frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_{L_i}^S(l)} < \sigma$ **then**
7:         $\sigma_l \leftarrow \frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_{L_i}^S(l)}$
8:         $\sigma \leftarrow \frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_{L_i}^S(l)}$
9:     **else**
10:         $\sigma_l = \sigma$
11:     **end if**
12:     $L_{i+1} \leftarrow L_i \setminus l$
13:     $i \leftarrow i + 1$
14: **end for**
15: Return $(\sigma_l)_{l \in V}$.

Theorem 4.3. *Let $(\sigma_l)_{l \in V}$ be the values returned by Algorithm 2. Then, $\sigma_l^* \geq \sigma_l$ for each l.*

Proof. Consider the first round where L_1 contains all the nodes in V. Suppose we select a node l with the maximum $\frac{\sum_S \pi(S) I(S,k)}{\sum_S \pi(S) d_{L_1}^S(k)}$. Let L be a limiting set for node l. We have $\frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_L^S(l)} \geq \frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_{L_1}^S(l)}$ since $L \subseteq L_1$. By applying Lemma 42 and Lemma 48, $\sigma_l^* = \sigma_L^* \geq \frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_L^S(l)} \geq \frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_{L_1}^S(l)} = \sigma_l$.

We make the induction hypothesis that $\sigma_l^* \geq \sigma_l$ for all the removed nodes (in line 12) in the first h iterations of the algorithm.

For the removed node l in the $(h+1)$-th iteration, let L be a limiting set of node l. There are two possibilities: (1) L contains a previously removed node; or (2) L does not contain any previously removed node. In case (1), let us assume L contains a previously removed node k. By Corollary 5, $\sigma_l^* = \sigma_L^* \geq \sigma_k^*$. Since σ is non-increasing in the algorithm, we have $\sigma[k] \geq \sigma[l]$, where $\sigma[k]$ (or $\sigma[l]$) is the value of σ just before k (or l, respectively) is removed. Hence, $\sigma_l^* = \sigma_L^* \geq \sigma_k^* \geq \sigma_k = \sigma[k] \geq \sigma[l] = \sigma_l$. The second inequality is due to be induction hypothesis.

In case (2), since the limiting set L for l does not contain any previously removed nodes, we must have $L \subseteq L_{h+1}$, and hence, $\frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_L^S(l)} \geq \frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_{L_{h+1}}^S(l)}$. Thus, $\sigma_l^* = \sigma_L^* \geq \frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_L^S(l)} \geq \frac{\sum_S \pi(S) I(S,l)}{\sum_S \pi(S) d_{L_{h+1}}^S(l)} \geq \sigma_l$. The first inequality is due to Lemma 48; the last inequality is due to lines 6-11.

Therefore, in either case, $\sigma_l^* \geq \sigma_l$ holds. By induction, $\sigma_l^* \geq \sigma_l$ for every l. □

Example: Consider the interference graph in Fig. 4-1, which contains 4 nodes. For this graph, Algorithm 2 works as follows.

1. Initially, $\sigma = 1$ and $L_1 = V$.

2. We have $\frac{\sum_S \pi(S) I(S,1)}{\sum_S \pi(S) d_{L_1}^S(1)} = 2/3$, $\frac{\sum_S \pi(S) I(S,2)}{\sum_S \pi(S) d_{L_1}^S(2)} = 3/7$, $\frac{\sum_S \pi(S) I(S,3)}{\sum_S \pi(S) d_{L_1}^S(3)} = 2/3$, $\frac{\sum_S \pi(S) I(S,4)}{\sum_S \pi(S) d_{L_1}^S(4)} = 1$.

   We get $\sigma_4 = \sigma = 1$.

   Remove node 4 from L_1 to form L_2.

3. We have $\frac{\sum_S \pi(S) I(S,1)}{\sum_S \pi(S) d_{L_2}^S(1)} = 2/3$, $\frac{\sum_S \pi(S) I(S,2)}{\sum_S \pi(S) d_{L_2}^S(2)} = 3/4$, $\frac{\sum_S \pi(S) I(S,3)}{\sum_S \pi(S) d_{L_2}^S(3)} = 2/3$.

   We get $\sigma_2 = 3/4$. Set $\sigma = 3/4$.

   Remove node 2 from L_2 to form L_3.

4. We have $\frac{\sum_S \pi(S) I(S,1)}{\sum_S \pi(S) d_{L_3}^S(1)} = 1$, $\frac{\sum_S \pi(S) I(S,3)}{\sum_S \pi(S) d_{L_3}^S(3)} = 1$.

   We get $\sigma_1 = \sigma = 3/4$.

   Remove node 1 from L_3 to form L_4.

5. We have $\frac{\sum_S \pi(S) I(S,3)}{\sum_S \pi(S) d_{L_4}^S(3)} = 1$.

   We get $\sigma_3 = \sigma = 3/4$.

Remove node 3 from L_4.

4.5 Summary

In this chapter, we investigate the performance of the LQF scheduling policy in wireless networks with interference graph model under channel fading. We generalize the fading structure defined in [29] to adapt wider class of fading. Specifically, we allow the difference of interference graph in different channel states. Under the newly defined fading structure, two links interfere with each other in one channel state may not interfere with each other in another channel state. Meanwhile, like the fading model in [29], only a subset of links which determined by the current channel states are permitted to perform data transmission. For this generalized fading channel model, we introduce the concept of link σ-local pooling, which allows heterogeneous performance characterization of individual link. We define the $\Sigma^*(G)$ diagonal matrix, which contains the link σ-local pooling factors in the diagonal entries, as a generalization of the Fading Local Pooling factor(F-LPF), in [29]. The matrix $\Sigma^*(G)$ provides a refined performance characterization for LQF under channel fading. We show that a larger stability region of LQF are captured with the theory and tool developed in this chapter.

We then introduce a set of theory that helps to apply the new idea of link σ-local pooling under channel fading. The concepts of σ-local pooling for a set of links and the limiting set for a link are established. Based on the developed theory, we introduce new estimation methods for set and link σ-local pooling factors under channel fading.

4.6 Appendix

4.6.1 Fluid Limit

We consider the wireless network as a slot system, where $\hat{A}_l(k)$ represents the instantaneous arrival on link l at slot k. Let $A_l(k)$ represent the total number of exogenous arrivals to link l up to and including time slot k. We then have:

$$A_l(k) = \sum_{i=1}^{k} \hat{A}_l(i).$$

We assume the channel state does not change within every slot. Let the process of channel variation being $\{\mathbf{1}_S(k), k = 1, 2, 3, ...\}$ for all channel state S, where $\mathbf{1}_S(k) = 1$ if and only if the channel state is S at slot k. According to the assumption, both the arrival process and channel variation process is i.i.d.. Hence, by the strong law of large number, we have almost surely:

$$\lim_{k\to\infty} A_l(k)/k \to \lambda_l.$$
$$\lim_{k\to\infty} \sum_{i=1}^{k} \mathbf{1}_S(i)/k \to \pi(S).$$

Let $F(S)$ denote the set of feasible schedule under channel state S. Note that, the maximal schedule under channel state S is a subset of $F(S)$, i.e. $M^S \subseteq F(S)$. For any schedule $m \subseteq F(S)$, let $T_{m(S)}(k)$ record the number of time slots up to and including slot k, during which the scheduling policy chooses m under channel state S. Note that, the same set of links could be activated under different channel states. However, they will be counted separately as two different schedules, differentiate by the channel state. Let m_l be an identifier of whether set m contains link l, i.e, $m_l = 1$ when m contains link l and $m_l = 0$ otherwise. Let $D_l(k)$ denote the total service for link l up to and including slot k. We then have the following:

$$Q(k) = Q(0) + A(k) - D(k),$$
$$D_l(k+1) - D_l(k) = \begin{cases} \sum_S \sum_{m\in F(S)} m_l(T_{m(S)}(k+1) - T_{m(S)}(k)) & \text{if} Q_l(k) > 0, \\ 0 & \text{otherwise}, \end{cases}$$
$$Q_l(k) \geq 0 \qquad \text{for all } l \in V \text{ and } k \geq 0.$$

Now, consider a sequence of system states indexed by $n = 1, 2, \ldots$. Let

$$A^n(t) = \frac{1}{n}A(\lfloor nt \rfloor),$$
$$Q^n(t) = \frac{1}{n}Q(\lfloor nt \rfloor),$$
$$D^n(t) = \frac{1}{n}D(\lfloor nt \rfloor),$$
$$T^n(t) = \frac{1}{n}T(\lfloor nt \rfloor).$$

Proposition 4.1. *A limit $(\overline{A}(.), \overline{Q}(.), \overline{T}(.), \overline{D}(.))$ of $(A^n(.), Q^n(.), T^n(.), D^n(.))$ as $n \to \infty$ almost surely exists, in the topology of uniform convergence over compact sets, along some subsequence, and satisfies the following properties(where $\frac{d}{dt}$ denotes ordinary differentiation with respect to time):*

$$\overline{Q}(t) = \overline{Q}(0) + \lambda t - \overline{D}(t) \text{ for all } t \geq 0,$$
$$\sum_{m \in F(S)} \frac{d}{dt}\overline{T}_{m(S)}(t) = \pi(S) \text{ for all state } S \text{ and } t \geq 0,$$
$$\frac{d}{dt}\overline{D}_l(t) = \sum_S \sum_{m \in F(S)} m_l \frac{d}{dt}\overline{T}_{m(S)}(t), \text{ if } \overline{Q}_l(t) > 0,$$

$\overline{D}_l(t)$ *and* $\overline{T}_{m(S)}(t)$ *are nonnegative and nondecreasing*

$\overline{Q}_l(t) \geq 0,$

for all state S, *all* $l \in V$ *and* $t \geq 0$.

Moreover, $\overline{Q}(.)$, $\overline{T}_{m(S)}(.)$, and $\overline{D}(.)$ are absolutely continuous. Times $t \geq 0$ for which the derivatives of $\overline{Q}(t)$, $\overline{T}_{m(S)}(t)$, and $\overline{D}(t)$ exist will be called regular times.

By some simple normalization, we have:

$$\sum_{m \in F(S)} \frac{d}{dt}\left(\frac{\overline{T}_{m(S)}(t)}{\pi(S)}\right) = 1 \text{ for all state } S \text{ and } t \geq 0,$$

$$\frac{d}{dt}\overline{D}_l(t) = \sum_S \sum_{m \in F(S)} \pi(S) m_l \frac{d}{dt}\left(\frac{\overline{T}_{m(S)}(t)}{\pi(S)}\right)$$

$$= \sum_S \pi(S) \sum_{m \in F(S)} m_l \frac{d}{dt}\left(\frac{\overline{T}_{m(S)}(t)}{\pi(S)}\right)$$

if $\overline{Q}_l(t) > 0$ for all state S and $t \geq 0$.

Consider the fluid limit of the queue process, denoted by $\{\overline{Q}_l(t)\}_{t \geq 0}$, for all $l \in V$. For a fixed time instance t, let L be the set of those longest queues(with equal length) whose time derivatives at t, $\frac{d}{dt}\overline{Q}_l(t)$, are the largest(also identical) under the given LQF policy. The queues in L will remain the longest with identical length in the next infinitesimally small time interval. Since queues in L are given higher priority by LQF than those in $V \setminus L$ during that infinitesimally small time interval, we have:

$$\left[\sum_{m \in F(S)} \frac{d}{dt}\left(\frac{\overline{T}_{m(S)}(t)}{\pi(S)}\right)\right]_L \in Co(M_L^S).$$

Figure 4-1. Interference graphs of a four-link network under channel fading. S_1, S_2 and S_3 are three channel states.

CHAPTER 5
CONCLUSION

In this dissertation, we address the problem of performance guarantee of the specific greedy scheduling algorithm Longest Queue First(LQF) in wireless networks. We aim to enlarge the performance lower bound and capture the stability region of LQF more precisely.

In our first piece of work, we provide a multiple-parameter based framework on performance characterization of the LQF policy. In particular, we introduce the concept of link σ-local pooling, which helps to characterize the individual link performance heterogeneously. We define the $\Sigma^*(G)$ diagonal matrix, which contains the link σ-local pooling factors in the diagonal entries, extending the concept of the network σ-local pooling, $\sigma^*(G)$, in [18]. We show that using the diagonal matrix $\Sigma^*(G)$, we captures a larger region of stability of LQF than previous results.

In our second piece of work, we further improve the performance characterization of LQF. Contrary to the previously-known regions of stability, the closures of the newly discovered stability regions contain all the extreme points of the capacity region Λ, but they are not convex in general. The study reveals a counter-intuitive phenomenon where increasing the arrival rates helps LQF to stabilize the network. Using the theories and concepts developed in this study, we well explained the reason of this counter-intuitive situation.

In our third piece of work, we focus on the performance of LQF in a more generic wireless network model, where the channel state is time-varying. Unlike the channel fading model discussed in [29], we allow the change of underlying interference graph in addition to the alternating On and OFF channel state for each link. We develop a multiple-parameter based framework in order to capture a larger stability region of LQF under channel fading. We also establish an efficient estimation algorithm of the performance lower bound of LQF based on the theories and concepts we have build.

REFERENCES

[1] Andrews, Matthew, Kumaran, Krishnan, Ramanan, Kavita, Stolyar, Alexander, Vijayakumar, Rajiv, and Whiting, Phil. "Scheduling in a queueing system with asynchronously varying service rates." *Probab. Eng. Inf. Sci.* 18 (2004).2: 191–217.

[2] Balakrishnan, H., Barrett, C., Kumar, V., Marathe, M., and Thite, S. "The distance-2 matching problem and its relationship to the MAC-layer capacity of ad hoc networks." *IEEE Journal On Selected Areas In Communications* 22 (2004).6: 1069–1079.

[3] Birand, Berk, Chudnovsky, Maria, Ries, Bernard, Seymour, Paul, Zussman, Gil, and Zwols, Yori. "Analyzing the performance of greedy maximal scheduling via local pooling and graph theory." Tech. rep., Columbia University, 2009.

[4] ———. "Analyzing the performance of greedy maximal scheduling via local pooling and graph theory." *Proceedings of IEEE INFOCOM*. 2010.

[5] Borbash, S.A. and Ephremides, A. "Wireless link scheduling with power control and SINR constraints." *Information Theory, IEEE Transactions on* 52 (2006).11: 5106 –5111.

[6] Boyaci, Cem., Li, Bo, and Xia, Ye. "An investigation on the nature of wireless scheduling." *Proceedings of IEEE INFOCOM*. 2010.

[7] Brzezinski, A., Zussman, G., and Modiano, E. "Local pooling conditions for joint routing and scheduling." *Information Theory and Applications Workshop*. 2008, 499–506.

[8] Brzezinski, Andrew, Zussman, Gil, and Modiano, Eytan. "Enabling distributed throughput maximization in wireless mesh networks: a partitioning approach." *Proceedings of ACM MobiCom*. 2006, 26–37.

[9] Chaporkar, Prasanna, Kar, Koushik, and Sarkar, Saswati. "Throughput guarantees through maximal scheduling in wireless networks." *Proceedings of 43d Annual Allerton Conference on Communication, Control and Computing*. 2005, 28–30.

[10] Chen, L., Low, S. H., Chiang, M., and Doyle, J. C. "Cross-layer congestion control, routing and scheduling design in Ad Hoc wireless networks." *Proceedings of IEEE INFOCOM*. 2006.

[11] Dai, J. G. "On positive Harris recurrence of multiclass queueing networks: A unified approach via fluid limit models." *Annals of Applied Probability* 5 (1995): 49–77.

[12] Dimakis, Antonis and Walrand, Jean. "Sufficient conditions for stability of Longest-Queue-First scheduling: Second-Order Properties Using Fluid Limits." *Advances in Applied Probability* 38 (2006): 505–521.

[13] Eryilmaz, A., Srikant, R., and Perkins, J.R. “Stable scheduling policies for fading wireless channels.” *Networking, IEEE/ACM Transactions on* 13 (2005).2: 411 – 424.

[14] Georgiadis, Leonidas, Neely, Michael J., and Tassiulas, Leandros. “Resource allocation and cross-layer control in wireless networks.” *Found. Trends Netw.* 1 (2006).1: 1–144.

[15] Gupta, A., Lin, X., and Srikant, R. “Low-complexity distributed scheduling algorithms for wireless networks.” *Proceedings of IEEE INFOCOM*. 2007.

[16] Gupta, Piyush and Kumar, P. R. “The capacity of wireless networks.” *IEEE Transactions on Information Theory* 46 (2000).2: 388–404.

[17] Joo, Changhee. “A local greedy scheduling scheme with provable performance guarantee.” *Proceedings of MobiHoc*. 2008, 111–120.

[18] Joo, Changhee, Lin, Xiaojun, and Shroff, N. B. “Understanding the capacity region of the greedy maximal scheduling algorithm in multi-hop wireless networks.” *Proceedings of IEEE INFOCOM*. 2008.

[19] Joo, Changhee and Shroff, Ness. “Performance of random access scheduling schemes in multi-hop wireless networks.” *Proceedings of IEEE INFOCOM*. 2007.

[20] Le, Long Bao, Modiano, E., and Shroff, N.B. “Optimal control of wireless networks with finite buffers.” *INFOCOM, 2010 Proceedings IEEE*. 2010, 1 –9.

[21] Le, Long Bao, Modiano, Eytan, Joo, Changhee, and Shroff, Ness B. “Longest-queue-first scheduling under SINR interference model.” *Proceedings of the eleventh ACM international symposium on Mobile ad hoc networking and computing*. MobiHoc ’10. New York, NY, USA: ACM, 2010, 41–50.

[22] Leconte, Mathieu, Ni, Jian, and Srikant, R. “Improved bounds on the throughput efficiency of greedy maximal scheduling in wireless networks.” *Proceedings of MobiHoc*. 2009.

[23] Li, Bo, Boyaci, Cem, and Xia, Ye. “A refined performance characterization of longest-queue-first policy in wireless networks.” *Proceedings of MobiHoc ’09*. New Orleans, LA, USA, 2009, 65–74. Also accepted to *IEEE/ACM Transactions on Networking*.

[24] Lin, Xiaojun and Rasool, Shahzada B. “Constant-time distributed scheduling policies for ad hoc wireless networks.” *Proceedings of the IEEE CDC*. 2006.

[25] Lin, Xiaojun, Shroff, Ness B., and Srikant, R. “The impact of imperfect scheduling on cross-layer rate control in wireless networks.” *IEEE/ACM Transaction on Networking* 14 (2006).2: 302–315.

[26] Modiano, Eytan, Shah, Devavrat, and Zussman, Gil. “Maximizing throughput in wireless networks via gossiping.” *ACM SIGMETRICS Performance Evaluation Review* 34 (2006).1: 27–38.

[27] Neely, M., Modiano, E., and Rohrs, C. “Dynamic power allocation and routing for time varying wireless networks.” 2003.

[28] Neely, Michael J. “Energy Optimal Control for Time-Varying Wireless Networks.” *IEEE Transactions on Information Theory* 52 (2006).7: 2915–2934.

[29] Reddy, Akula A, Sanghavi, Sujay, and Shakkottai, Sanjay. “On the effect of channel fading on greedy scheduling.” *Proceedings of IEEE INFOCOM*. 2012.

[30] Ries, Maria Chudnovsky Bernard and Zwols, Yori. “Claw-free graphs with strongly perfect complements. fractional and integral version.” Tech. rep., Columbia University, 2010.

[31] Sanghavi, Sujay, Bui, Loc, and Srikant, R. “Distributed link scheduling with constant overhead.” *ACM SIGMETRICS Performance Evaluation Review* 35 (2007).1: 313–324.

[32] Scheinerman, Edward R. and Ullman, Daniel H. *Fractional graph theory: A rational approach to the theory of graphs*. New York, USA: John Wiley & Sons Inc., 1997.

[33] Shakkottai, Sanjay, Srikant, R., and Stolyar, Alexander L. “Pathwise optimality of the exponential scheduling rule for wireless channels.” *Advances in Applied Probability*. 2004, 2004.

[34] Shakkottai, Sanjay and Stolyar, Alexander L. “Scheduling for multiple flows sharing a time-varying channel: the exponential rule.” (2000).

[35] Sharma, G., Mazumdar, R. R., and Shroff, N. B. “On the complexity of scheduling in wireless networks.” *Proccedings of ACM MobiCom*. 2006, 227–238.

[36] Sharma, Gaurav, Shroff, Ness B., and Mazumdar, Ravi R. “Maximum weighted matching with interference constraints.” *PERCOMW ’06: Proceedings of the 4th annual IEEE international conference on Pervasive Computing and Communications Workshops*. 2006.

[37] ———. “Maximum weighted matching with interference constraints.” *PERCOMW ’06: Proceedings of the 4th annual IEEE international conference on Pervasive Computing and Communications Workshops*. Washington, DC, 2006.

[38] ———. “Joint congestion control and distributed scheduling for throughput guarantees in wireless networks.” *Proceedings of IEEE INFOCOM*. 2007, 2072–2080.

[39] Sridharan, Arun, Koksal, C.Emre, and Uysal-Biyikoglu, Elif. “A greedy link scheduler for wireless networks with Gaussian multiple access and broadcast channels.” *Proceedings of IEEE INFOCOM*. 2010.

[40] Tassiulas, L. “Scheduling and performance limits of networks with constantly changing topology.” *Information Theory, IEEE Transactions on* 43 (1997).3: 1067 –1073.

[41] Tassiulas, L. and Ephremides, A. “Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks.” *IEEE Transactions on Automatic Control* 37 (1992).12: 1936–1948.

[42] ———. “Dynamic server allocation to parallel queues with randomly varying connectivity.” *Information Theory, IEEE Transactions on* 39 (1993).2: 466 –478.

[43] Tassiulas, Leandros. “Linear complexity algorithms for maximum throughput in radio networks and input queued switches.” *Proceedings of IEEE INFOCOM*. 1998.

[44] Wan, Peng-Jun, Xu, Xiaohua, Wang, Zhu, Tang, Shaojie, and Wan, Zhiguo. “Stability analyses of longest-queue-first link scheduling in MC-MR wireless networks.” *Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing*. MobiHoc ’12. New York, NY, USA: ACM, 2012, 45–54.

[45] West, Douglas. *Introduction to graph theory*. Prentice Hall, 2001.

[46] Wu, Xinzhou, Srikant, R., and Perkings, James R. “Scheduling efficiency of distributed greedy scheduling algorithms in wireless networks.” *Proceedings of IEEE INFOCOM*. 2006.

[47] Zussman, G., Brzezinski, A., and Modiano, E. “Multihop local pooling for distributed throughput maximization in wireless networks.” *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*. 2008, 1139 –1147.

BIOGRAPHICAL SKETCH

Bo Li received his bachelor's and master's degrees in department of electronics and information engineering from Huazhong University of Science and Technology, P.R. China, in 2004 and 2006, respectively. He received his Ph.D from the Computer and Information Science and Engineering department of University of Florida in the summer of 2012. His research interests are in the computer networking area, including link scheduling for wireless networks, network resource allocation. He is also interested in convex optimization, stochastic process, graph theory and algorithm.