On Providing Non-uniform Scheduling Guarantees in a Wireless Network

Technical Report

Vartika Bhandari
University of Illinois at Urbana-Champaign
vartikab@acm.org

Nitin H. Vaidya
University of Illinois at Urbana-Champaign
nhv@illinois.edu

Abstract

Significant research effort has been directed towards the design and performance analysis of imperfect scheduling policies for wireless networks. These imperfect schedulers are of interest despite being sub-optimal, as they allow for more tractable implementation at the expense of some loss in performance. However much of this prior work takes a uniform scaling approach to analyzing scheduling performance, whereby the performance of a scheduling policy is characterized in terms of a single scalar quantity, the efficiency-ratio. While suitable for characterizing worst-case performance, this approach limits one's ability to understand the different extents of performance degradation that may be experienced by different links in a network. Such an understanding is very valuable when average performance is of greater interest than the worst-case, or when certain links are more important than others. Furthermore, once one approaches scheduler design with non-uniform performance guarantees in mind, one finds that simple modifications to well-known scheduling algorithms can yield substantially improved non-uniform scaling results compared to the original algorithms. In this paper, we make a comprehensive case for adopting such an approach by presenting non-uniform scaling results for a set of algorithms that are variants of well-known algorithms from the class of maximal schedulers.

I. Introduction

Substantial recent research effort has been directed towards the design of *imperfect* scheduling policies [1], [2], [3], [4] for wireless networks, and analyzing their performance. These imperfect schedulers are of interest despite being sub-optimal, as they allow for more tractable implementation at the expense of some loss in performance. However, much of this prior work takes a uniform scaling approach to analyzing scheduling performance whereby the performance of a scheduling policy is characterized in terms of a single scalar quantity–the *efficiency-ratio*.[1] While this leads to a compact and simple characterization, it ties down the performance criterion to the worst-case degradation experienced by any link in the network. In a large range of scenarios, it is likely that many or most links in the network may be able to achieve much better throughput. When the average experience of most links is more important than the worst-case, it is more relevant to consider the performance achieved by each link, rather than use the performance of the worst-case link as a metric. Similarly, when all links are not equally important, one may care about trying to provide performance guarantees proportional to each link's importance. In such scenarios, it is important to be able to understand what kind of differentiated guarantees a scheduling algorithm can provide to different links. Thus, it is very relevant to attempt performance analysis based on an *efficiency-vector*[2] rather than a scalar *efficiency-ratio*.

While most of the relevant prior work takes a uniform scaling approach, it must be noted that some non-uniform scaling bounds were indeed proved in [5] for a maximal scheduling algorithm. More recently, non-uniform scaling bounds for the Longest Queue First scheduling algorithm were proved in [6].

In this paper, we make a much more comprehensive attempt to make a case for efficiency-vector based performance analysis. In particular, we show that simple modifications involving introduction of priorities to well known scheduling algorithms from the class of maximal schedulers, e.g., maximal scheduling with thresholds and centralized greedy maximal scheduling, enables one to achieve improved non-uniform bounds. This suggests that it may be possible to identify certain algorithm parameters (e.g., link priority), the careful adaptation of which can enable one to achieve desired differentiated performance guarantees over a range of scenarios.

This is a revised version of the technical report *Differentiated Performance Analysis of Maximal Scheduling in a Wireless Network* dated March 2009. Revision was performed in December 2009/January 2010.

Vartika Bhandari is now with Google Inc.

This research was supported in part by US Army Research Office grant W911NF-05-1-0246.

[1] The efficiency-ratio of an imperfect scheduler is said to be γ if: given any load vector $\vec{\lambda}$ such that the optimal scheduler can stabilize the network with load $\vec{\lambda}$, the imperfect scheduler can stabilize it for the scaled down vector $\gamma\vec{\lambda}$. The corresponding reduced rate region is referred to as the γ-reduced rate region.

[2] Analogous to efficiency-ratio, we can say that an algorithm achieves an efficiency-vector of $\vec{\gamma} = [\gamma_l]$ if: given any load vector $\vec{\lambda}$, such that the optimal scheduler can stabilize the network with load $\vec{\lambda}$, the imperfect scheduler can stabilize it for the scaled down vector $\vec{\gamma} \bullet \vec{\lambda}$) where we define $\vec{x} \bullet \vec{y}$ as the componentwise product of x and y, i.e., $\vec{x} \bullet \vec{y} = \vec{z}$ where $z_l = x_l y_l$. The corresponding reduced rate region is referred to as the $\vec{\gamma}$-reduced rate region.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
JAN 2010		**00-00-2010 to 00-00-2010**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
On Providing Non-uniform Scheduling Guarantees in a Wireless Network	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Illinois at Urbana-Champaign,Department of Electrical and Computer Engineering,Coordinated Science Laboratory,Urbana,IL,61801**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Significant research effort has been directed towards the design and performance analysis of imperfect scheduling policies for wireless networks. These imperfect schedulers are of interest despite being sub-optimal, as they allow for more tractable implementation at the expense of some loss in performance. However much of this prior work takes a uniform scaling approach to analyzing scheduling performance, whereby the performance of a scheduling policy is characterized in terms of a single scalar quantity, the efficiency-ratio. While suitable for characterizing worst-case performance, this approach limits one?s ability to understand the different extents of performance degradation that may be experienced by different links in a network. Such an understanding is very valuable when average performance is of greater interest than the worst-case, or when certain links are more important than others. Furthermore, once one approaches scheduler design with non-uniform performance guarantees in mind, one finds that simple modifications to well-known scheduling algorithms can yield substantially improved non-uniform scaling results compared to the original algorithms. In this paper, we make a comprehensive case for adopting such an approach by presenting non-uniform scaling results for a set of algorithms that are variants of well-known algorithms from the class of maximal schedulers.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**11**	

It must be noted that maximal schedulers are of practical interest, since they can potentially be approximately implemented using backoff schemes [7], [8], or probabilistic random-access schemes [9]. These approaches can also be modified to allow for prioritization through suitable modulation of backoff intervals and/or access probabilities, and thus the results presented in this paper can provide useful insight for practical MAC protocol design.

II. Notation and Terminology

We assume the availability of a single channel for communication. The wireless network is viewed as a directed graph, with each directed link in the graph representing an available (directed) communication link between a node pair capable of communication with non-zero rate. We model interference using a *conflict* relation between links. Two links are said to conflict with each other if it is only feasible to schedule at most one of the links at any given time. The conflict relation is assumed to be symmetric. The conflict-based interference model provides a tractable approximation of reality – while it does not capture the wireless channel precisely, it is more amenable to analysis. Such conflict-based interference models have also been used in past related work (e.g., [10], [11]), etc.

We assume a single channel of operation. Time is assumed to be slotted, with the slot duration being 1 unit time (i.e., we use slot duration as the time unit). In each time slot, the scheduler used in the network determines which links should transmit in that time slot. We also adopt the following convention: at the beginning of each time-slot, the scheduling decisions are taken, and transmissions occur. Then new arrivals occur at the end of the slot.

We now introduce some notation and terminology.

- $\mathcal{L}$ denotes the set of directed links in the network.
- $\mathbf{I}(l)$ denotes the set of links that conflict with link l. As a matter of convention we assume that $l \in \mathbf{I}(l)$.
- K_l denotes the maximum number of links in $\mathbf{I}(l)$ that can be scheduled simultaneously if l is not scheduled.
- K is the largest value of K_l over all links l, i.e., $K = \max_l K_l$.
- $\widetilde{K_l} = \max\{1, K_l\}$.
- $\widetilde{K} = \max\{1, K\}$.
- $I_{max} = \max_{l \in \mathcal{L}} |\mathbf{I}(l)|$

We limit our focus to single-hop flows. Thus, all traffic over link l can be viewed as a single aggregated flow over that link. We also use the following notational convention for convenience: given vector $\overrightarrow{\gamma} = [\gamma_1, \gamma_2, ..., \gamma_{\mathcal{L}}]$, $\overrightarrow{\gamma}^{-1}$ denotes the vector $[\frac{1}{\gamma_1}, \frac{1}{\gamma_2}, ... \frac{1}{\gamma_{|\mathcal{L}|}}]$.

III. Related Work

The seminal work of Tassiulas and Ephremides[12] yielded a *throughput-optimal* scheduler (the Dynamic Backpressure Scheduler), which can schedule all "feasible" traffic flows without resulting in unbounded queues. However, such an optimal scheduler is difficult to implement in practice. Hence, various imperfect scheduling strategies that trade-off throughput for simplicity have been proposed in [1], [2], [3], [4] amongst others. A queue-loading rule for maximal scheduling in multi-channel wireless networks is presented in [11].

Related to this work, [5] presents some non-uniform scaling results for a simple maximal scheduler with threshold-rule. It is shown that each link achieves a scaling of $\frac{1}{\max_{k \in \mathbf{I}(l)} \widetilde{K_k}}$. In Section IV of this paper, we show how introducing prioritization in a maximal scheduler with threshold-rule helps improve the achievable non-uniform scaling guarantees. In [13] uniform scaling results are presented for certain maximal schedulers with priorities. Their focus is on proving rate-stability. While this paper also considers certain maximal schedulers with priorities, we focus on *non-uniform bounds*, and prove *queue-stability*, which is a stronger condition.

More recently, non-uniform scaling results for Longest Queue First scheduling have been presented in [6].

IV. Local K-Precedence Based Maximal Scheduler

A maximal scheduler much studied in prior work such as [3], [10], [5] for its potential amenability to distributed implementation is the following:

Maximal Scheduler with Threshold Rule*:* At the beginning of each slot t: all those links l with $q_l(t) \geq r_l$ participate in the scheduling process for that slot. From amongst the participating links, a maximal schedule is computed, i.e., if a participating link l is not scheduled, then some link conflicting with l must be scheduled. The following uniform and non-uniform bounds are known for this scheduler:

- *Uniform Bound:* As proved in [3], [10], this scheduler can achieve an efficiency-ratio:

$$\gamma = \frac{1}{\widetilde{K}} \tag{1}$$

- *Non-uniform Bound:* As proved in [5], this scheduler can achieve an efficiency-vector:

$$\overrightarrow{\gamma} \text{ where } \gamma_l = \frac{1}{\max\limits_{k \in \mathbf{I}(l)} \widetilde{K}_k} \tag{2}$$

We now describe a simple variation on the maximal scheduler with threshold rule:

Local K-precedence based Maximal Scheduler*:* In each time slot t, only those links l with $q_l(t) \geq r_l$ participate in scheduling. The scheduler computes a schedule with the following property:

If link l participates in scheduling, the either l is scheduled, or some conflicting link $k \in \mathbf{I}(l)$ with $\widetilde{K}_k \geq \widetilde{K}_l$ is scheduled.

An alternative description in terms of priority-assignment is as follows:

Each link l has a priority value $\phi(l) = \widetilde{K} - \widetilde{K}_l + 1$, where $\phi(l) < \phi(k)$ implies l has higher priority than k. In each slot, a maximal schedule is computed from amongst participating links by following the priority order. Thus, either a participating link l is scheduled, or some link $k \in \mathbf{I}(l) \setminus \{l\}$ with equal or higher priority must be scheduled.

An approximation to such a scheduler can be implemented using a backoff based procedure, where each link l chooses a backoff value proportional on $\phi(l)$ (e.g., a link l could choose $\widetilde{K} - \widetilde{K}_l + 1$ as its backoff). Since $\widetilde{K}$ is typically a small constant for most wireless networks, the overhead incurred by the backoff window would be small.

The following assumptions are made about the arrival and channel rate processes:

The arrival process for link l is i.i.d. over all time-slots t, and is denoted by $\{\lambda_l(t)\}$, with $E[\lambda_l(t)] = \lambda_l$. We make no assumption about independence of arrival processes for two links l,k. However, we consider only the class of arrival processes for which $E[\lambda_l(t)\lambda_k(t)]$ is bounded, i.e., $E[\lambda_l(t)\lambda_k(t)] \leq \eta$ for all $l \in \mathcal{L}, k \in \mathcal{L}$, where η is a suitable constant. The rate r_l achievable on a link l is assumed to be time-invariant.

Theorem 1: The local K-precedence based scheduler can achieve an efficiency-vector $\overrightarrow{\gamma} = [\gamma_{l_1}, \gamma_{l_2}, ..., \gamma_{l_{|\mathcal{L}|}}]$ where:

$$\gamma_{l_i} = \frac{1}{\max\{1, K_{l_i}\}} = \frac{1}{\widetilde{K_{l_i}}}$$

The proof is presented in the appendix.

V. A General Bound for Prioritized Maximal Schedulers with Thresholds

The scheduler described in Section IV involves assignment of priorities to links. In this section, we make an effort to better understand the non-uniform scaling behavior of any generic maximal scheduler with thresholds and priorities.

We consider any arbitrary priority assignment to links. Unlike [13], we do not assume that the priorities are unique. Thus, two links may have equal priority. Moreover, the priorities do not even have to be locally unique, i.e., a link l and a link $k \in \mathbf{I}(l) \setminus \{l\}$ may have the same priority. Though this complicates the analysis slightly compared to the case of unique priorities, it is useful to consider this more general case for the following reason: in practice a prioritized scheduler might be implemented using a differentiated backoff mechanism. In such a scenario, the number of slots in the backoff window must be at least as many as the number of locally distinct priorities. Therefore, assigning unique priorities to all links would implies that the window-size must increase linearly in the number of network links, or at the very least linearly in I_{max}. In a large network with variable node density, it may be more practical to allocate priorities from a smaller set. In fact, we remark that the scheduler described in Section IV also assigns potentially non-unique priorities, since many links l (some of which may be mutually conflicting) may have the same value of K_l.

As in Section IV, we denote the priority of a link l by $\phi(l)$. $\phi(l) < \phi(k)$ implies that l has *higher priority* than k.

Let $\mathbf{H}(l) = \{k | k \in \mathbf{I}(l), \phi(k) < \phi(l)\}$. Thus, $\mathbf{H}(l)$ is the set of links that have a conflict with l and have strictly higher priority than l.

Let $\mathbf{Z}(l) = \{k | k \in \mathbf{I}(l), \phi(k) = \phi(l)\}$. Thus, $\mathbf{Z}(l)$ is the set of links in $\mathbf{I}(l)$ that have the same priority as l. Note that $l \in \mathbf{Z}(l)$.

Let h_l be the maximum number of links in $\mathbf{H}(l) \cup \mathbf{Z}(l)$ that can be concurrently scheduled if l is not scheduled, and $\widetilde{h_l} = \max\{1, h_l\}$.

Let $H_l = \max\limits_{k \in \mathbf{I}(l) \setminus \mathbf{H}(l)} \widetilde{h_k}$. It is not hard to see that for any link $k \in \mathbf{H}(l) \cup \mathbf{Z}(l)$, $l \in \mathbf{I}(k) \setminus \mathbf{H}(k)$, and therefore by definition:

$$\forall\ k \in \mathbf{H}(l) \cup \mathbf{Z}(l) : H_k \geq \widetilde{h_l} \tag{3}$$

Let $\overrightarrow{r}$ denote the vector of link-rates.

Consider the following scheduler:

In slot t, only links l with $q_l(t) \geq r_l$ participate, and a maximal schedule is computed from amongst participating links following priority order (equal priority links can be handled in arbitrary mutual order). Thus, if a link l participates and is not scheduled in slot t, this implies that some $k \in \mathbf{H}(l) \cup \mathbf{Z}(l) \setminus \{l\}$ must be scheduled in slot t.

Note that a link l that participates in scheduling can only be blocked by links in $\mathbf{H}(l) \cup \mathbf{Z}(l)$ since these have higher or equal priority to it.

We make the same assumptions about the arrival and link rate processes as in Section IV.

Theorem 2: Any prioritized maximal scheduler with thresholds having priority-vector $\overrightarrow{\phi}$ can stabilize any load-vector $\overrightarrow{\lambda}$ for which $\overrightarrow{\lambda}+\varepsilon_o\overrightarrow{r}$ lies within the $\overrightarrow{\gamma}$-reduced rate region, where $0<\varepsilon_o<1$ is a positive constant which can be chosen to be arbitrarily small (e.g., ε_o can be chosen to be 10^{-5}), and $\gamma_l=\frac{1}{H_l}$.
The proof is presented in the appendix.

VI. A Centralized Greedy Maximal Scheduler with Modified Weights

For the results in this section, we consider only the class of arrival processes with bounded second moments, i.e., $E[\lambda_l(t)^2]\leq\eta$ for all $l\in\mathcal{L}$, where η is a suitable constant. For simplicity, we retain the assumption of time-invariant link-rates, but the result of this section can be generalized to a wider class of well-behaved rate processes. For each link l, $r_l\leq R_{max}$ where R_{max} is some constant.

The centralized greedy maximal (CGM) scheduler is a well-studied instance of the class of maximal schedulers. It operates in the following manner:

In each time-slot t:

1) For each link l, compute link weight $w_l=q_l(t)r_l$.
2) Sort the links l in non-increasing order of w_l.
3) Add the first link in the sorted list (i.e., the one with highest weight) to the schedule for the time-slot, and remove from the list all links that are no longer feasible (due to conflicts).
4) Repeat step 3 until the list is exhausted (i.e., no more links can be added to the schedule).

For this scheduler, it is known that the efficiency-ratio is at least $\frac{1}{K}$.

We now describe a variant of the CGM Scheduler analogous to the local K-precedence based threshold maximal scheduler for which it is possible to prove non-uniform guarantees. This scheduler computes the weight for each link in a slightly different manner to that used by the CGM scheduler. In time-slot t:

1) For each link l, compute $w_l=\frac{q_l(t)r_l}{\widetilde{K}_l}$.
2) Sort the links l in non-increasing order of w_l.
3) From the sorted list, select the first link, i.e., the one with maximum weight, and include it in the schedule; eliminate all links conflicting with it
4) Repeat step 3 till no more links remain.

The rate allocated to a link l during slot t by the scheduler is denoted by $x_l(t)$. If a link is selected for scheduling in slot t, then $x_l(t)=r_l$, else $x_l(t)=0$.

$\mathcal{R}$ denotes the set of all feasible rate-allocations (these are rate-allocations that result from some conflict-free schedule).

Theorem 3: The centralized greedy maximal scheduler that uses link-weights $w_l=\frac{q_l(t)r_l}{\widetilde{K}_l}=\frac{q_l(t)r_l}{\max\{1,K_l\}}$ can achieve an efficiency-vector of $\overrightarrow{\gamma}$, where $\gamma_l=\frac{1}{\widetilde{K}_l}=\frac{1}{\max\{1,K_l\}}$.

To prove this, we first state and prove the following claim:

Lemma 1: If a scheduler selects the set of links to schedule, such that, in each slot, $\sum_{l\in\mathcal{L}}q_l(t)x_l(t)\geq\max_{\overrightarrow{y}\in\mathcal{R}}\sum_{l\in\mathcal{L}}q_l(t)\frac{y_l}{\widetilde{K}_l}$, then this scheduler achieves an efficiency-vector of $\overrightarrow{\gamma}$, where $\gamma_l=\frac{1}{\widetilde{K}_l}$.

Proof: Let $\overrightarrow{\lambda}$ be a traffic vector within the reduced rate-region. Given vector $\overrightarrow{\gamma}$, denote by $\overrightarrow{\gamma}^{-1}$ the vector $[\frac{1}{\gamma_1},\frac{1}{\gamma_2},\ldots\frac{1}{\gamma_{|\mathcal{L}|}}]$. Then $\overrightarrow{\gamma}^{-1}\bullet\overrightarrow{\lambda}$ lies within the convex-hull of $\mathcal{R}$ (recall the definition of $\overrightarrow{x}\bullet\overrightarrow{y}$ as the componentwise product of $\overrightarrow{x}$ and $\overrightarrow{y}$). Hence, $\overrightarrow{\lambda}$ lies within the convex hull of $\mathcal{R}'=\overrightarrow{\gamma}\bullet\mathcal{R}$. Therefore:

$$(1+\varepsilon)(\overrightarrow{q}\cdot\overrightarrow{\lambda})\leq\max_{\overrightarrow{y}\in\mathcal{R}'}\overrightarrow{q}\cdot\overrightarrow{y} \tag{4}$$

The dynamics of the queues in the network is as follows:

$$q_l(t+1)=(q_l(t)-x_l(t))_+ +\lambda_l(t) \tag{5}$$

where $x_l(t)$ is either r_l or 0 depending on whether l is scheduled or not.

Consider the following Lyapunov function:

$$V_q(t)=\sum_{l\in\mathcal{L}}(q_l(t))^2 \tag{6}$$

Noting that $(q_l(t+1))^2=((q_l(t)-x_l(t))_+ +\lambda_l(t))^2\leq\max\{(q_l(t)+(\lambda_l(t)-x_l(t)))^2,(\lambda_l(t))^2\}\leq(q_l(t)+(\lambda_l(t)-x_l(t)))^2+$

$(\lambda_l(t))^2$, we obtain the following:

$$\begin{aligned}
&E\left[\Delta(V_q(t))|\overrightarrow{q}(t)\right] \\
&= E\left[\sum_{l\in\mathcal{L}}(q_l(t+1))^2 - \sum_{l\in\mathcal{L}}q_l(t)^2\Big|\overrightarrow{q}(t)\right] \\
&\leq E\left[\sum_{l\in\mathcal{L}}\Big((q_l(t)+(\lambda_l(t)-x_l(t)))^2+(\lambda_l(t))^2-q_l(t)^2\Big)\Big|\overrightarrow{q}(t)\right] \\
&\leq 2E\left[\sum_{l\in\mathcal{L}}q_l(t)\left(\lambda_l(t)-x_l(t)\right)\Big|\overrightarrow{q}(t)\right]+C_1 \\
&\qquad\qquad \text{where } C_1=\eta+R_{max}^2 \\
&= 2\left(\sum_{l\in\mathcal{L}}q_l(t)\lambda_l-\sum_{l\in\mathcal{L}}q_l(t)x_l(t)\right)+C_1 \\
&\leq -\varepsilon\sum_{l\in\mathcal{L}}\lambda_l q_l(t)+C_1 \\
&\text{if } \sum_{l\in\mathcal{L}}q_l(t)x_l(t)\geq\max_{\vec{y}\in\mathcal{R}'}\left(\sum_{l\in\mathcal{L}}q_l(t)y_l\right) \text{ (using (4))}
\end{aligned} \tag{7}$$

■

We remark that Lemma 1 can be viewed as the non-uniform analogue of Proposition 3 in [1].

Let $\mathcal{S}_g$ denote the set of links selected by the CGM scheduler with modified weights. Consider any $l\in\mathcal{S}_g$. Let us denote by $\mathcal{B}(l)$ the maximum weight independent subset of links in $\mathbf{I}(l)\setminus\{l\}$ that were still eligible in the step when l was chosen. Evidently $|\mathcal{B}(l)|\leq K_l$. Furthermore, if $\mathcal{S}_{opt}$ is the set of links selected by a scheduler that maximizes $\sum_{l\in\mathcal{L}}q_l(t)\frac{x_l(t)}{\widetilde{K}_l}=\sum_{l\in\mathcal{S}_{opt}}q_l(t)\frac{r_l}{\widetilde{K}_l}$, then $\sum_{l\in\mathcal{S}_{opt}}\frac{q_l(t)r_l}{\widetilde{K}_l}\leq\sum_{l\in\mathcal{S}_g}\max\left\{\frac{q_l(t)r_l}{\widetilde{K}_l},\sum_{k\in\mathcal{B}(l)}\frac{q_k(t)r_k}{\widetilde{K}_k}\right\}$, since each link $l\in\mathcal{S}_g$ either also occurs in $\mathcal{S}_{opt}$ and thereby contributes its weight to it, or is the cause of blocking in $\mathcal{S}_g$ a set of links that occur in $\mathcal{S}_{opt}$, whose weight cannot exceed $\sum_{k\in\mathcal{B}(l)}\frac{q_k(t)r_k}{\widetilde{K}_k}$ by definition.

From the greedy nature of the scheduler, it follows that:

$$\frac{q_l(t)r_l}{\widetilde{K}_l}\geq\frac{q_k(t)r_k}{\widetilde{K}_k}\text{ for all } l\in\mathcal{S}_g, k\in\mathcal{B}(l) \tag{8}$$

Therefore:

$$\begin{aligned}
&q_l(t)r_l\geq\widetilde{K}_l\left(\frac{q_k(t)r_k}{\widetilde{K}_k}\right)\text{ for all } l\in\mathcal{S}_g, k\in\mathcal{B}(l) \\
&\therefore q_l(t)r_l\geq\widetilde{K}_l\left(\max_{k\in\mathcal{B}(l)}\left\{\frac{q_k(t)r_k}{\widetilde{K}_k}\right\}\right)\text{ for all } l\in\mathcal{S}_g \\
&\therefore q_l(t)r_l\geq\sum_{k\in\mathcal{B}(l)}\frac{q_k(t)r_k}{\widetilde{K}_k}\quad(\because|\mathcal{B}_l|\leq K_l) \\
&\therefore\sum_{l\in\mathcal{S}_g}q_l(t)r_l\geq\sum_{l\in\mathcal{S}_{opt}}q_l(t)\left(\frac{r_l}{\widetilde{K}_l}\right)
\end{aligned} \tag{9}$$

In light of Lemma 1, this proves the result.

VII. A Canonical Topology: The Star

In this section, we compare and discuss the implications of our results for a canonical topology—where the link-interference graph is a star (Fig. 1) with one center link and $K\geq 1$ radial links. This topology is often used as an example in work on scheduling algorithms.

In Section IV, we proved that the local K-precedence scheduler achieves an eficiency vector of $\left[\frac{1}{\widetilde{K}_l}\right]$. Since the scheduler of Section IV is also a priority based maximal scheduler, therefore Theorem 2 also applies to it. Thus, this scheduler can stabilize any vector that lies in the $\left[\frac{1}{\widetilde{K}_l}\right]$ reduced region, or in the reduced region specified by Theorem 2.

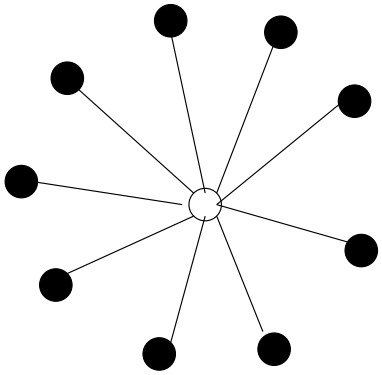

Fig. 1. A Star Topology

Let us consider what would happen when we use the local K-precedence scheduler in the star topology. In this case, the link m corresponding to the center vertex has priority 1 and its $\widetilde{h}_m = 1$, while all other links m' have priority K, and their respective $\widetilde{h}_{m'} = 1$. Therefore, for all links l in the network, it follows that $H_l = 1$. Thus, the local K-precedence based maximal scheduler is within an $\varepsilon_o \overrightarrow{r}$ margin of the optimal for the star topology. Since ε_o can be chosen to be extremely small, this is near-optimal. Thus, our non-uniform scaling results also yield a close-to-optimal uniform-scaling bound for this particular topology.

Note that the vanilla maximal scheduler with thresholds, from which the above scheduler is derived, can be shown to have an efficiency-ratio no better than $\frac{1}{K}$ in the case of the star topology. Thus, the use of precedence based on K_l yields a very substantial improvement in performance in this case.

It must be noted that for the special case of the star topology, other prior work has also shown performance-improvement when priority is given to the center link. In [13], it is shown that giving higher priority to the center link when performing prioritized maximal scheduling allows one to achieve rate-stability for all vectors within the rate-region. Note that our result proves queue-stability, which is a much stronger result. Similarly, in [9], it is shown that when using a random access protocol, breaking ties in favor of the center link yields substantially better performance than $\frac{1}{K}$.

VIII. Discussion

The results presented in this paper are not only examples of non-uniform performance analysis, but also highlight how it may be possible to achieve desirable non-uniform guarantees by appropriate assignment of priorities to links. For instance, our non-uniform performance bound for the local K-precedence based scheduler of Section IV is an improvement over the previous known uniform and non-uniform bounds for the vanilla maximal scheduler with thresholds [3], [5]. Our general result for any prioritized scheduler (Theorem 2) can be helpful in determining suitable priority assignments for small known-topology networks to achieve desired differentiated levels of performance.

It must also be noted that our result for the modified-weight CGM scheduler proves the same non-uniform bound as for the local K-precedence scheduler; however, the two schedulers achieve this bound in different ways: the CGM variant effectively gives precedence to links l with lower $\widetilde{K}_l$ by using weights inversely proportional to $\widetilde{K}_l$, whereas the local K-precedence scheduler gives precedence to links with larger $\widetilde{K}_l$. This is not surprising as the two algorithms operate quite differently. The CGM approach gives precedence according to weight, and thus, a single higher weight link can prevent concurrent scheduling of multiple links with only slightly lower weight. Modifying the weight formulation to privilege lower $\widetilde{K}_l$ addresses this. On the other hand, the maximal scheduler with thresholds chooses any maximal schedule from amongst eligible links, and thus, a link l with large $\widetilde{K}_l$ may get a much lower fraction of time if links in $\mathbf{I}(l)$ which could potentially have been active concurrently, become eligible at different times, and are scheduled sequentially, thereby increasing the fraction of time it is blocked by up to a factor of $\widetilde{K}_l$. Giving priority to links with higher $\widetilde{K}_l$ addresses this.

It must also be emphasized that explicitly seeking to prove non-uniform performance bounds can also lead to a paradigm-shift in the manner in which scheduler design is approached. For instance, the prioritized schedulers discussed in this paper resulted from an effort to identify the circumstances in which certain links in a network could be guaranteed better scaling than the remaining links. More specifically, these simple schedulers suggest that, given a network where we may seek to provide different levels of service for different links, one can potentially leverage tunable parameters such as link priority to design algorithms that provably achieve the desired non-uniform performance bounds.

References

[1] X. Lin and N. B. Shroff, "The impact of imperfect scheduling on cross-layer rate control in wireless networks," in *Proceedings of IEEE INFOCOM*, 2005, pp. 1804–1814.

[2] X. Wu and R. Srikant, "Scheduling efficiency of distributed greedy scheduling algorithms in wireless networks," in *Proceedings of IEEE INFOCOM*, 2006.
[3] X. Wu, R. Srikant, and J. R. Perkins, "Queue-length stability of maximal greedy schedules in wireless networks," in *Workshop on Information Theory and Applications*, 2006.
[4] G. Sharma, R. R. Mazumdar, and N. B. Shroff, "On the complexity of scheduling in wireless networks," in *MobiCom '06: Proceedings of the 12th annual international conference on Mobile computing and networking*. ACM, 2006, pp. 227–238.
[5] S. Sarkar, P. Chaporkar, and K. Kar, "Fairness and throughput guarantees with maximal scheduling in multi-hop wireless networks," in *WiOpt*, 2006, pp. 286–298.
[6] B. Li, C. Boyaci, and Y. Xia, "A refined performance characterization of longest-queue-first policy in wireless networks," in *MobiHoc '09: Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*. ACM, 2009, pp. 65–74.
[7] X. Lin and S. B. Rasool, "Constant-time distributed scheduling policies for ad hoc wireless networks," in *Proc. of IEEE Conference on Decision and Control*, 2006.
[8] C. Joo and N. B. Shroff, "Performance of random access scheduling schemes in multi-hop wireless networks," in *Proceedings of IEEE INFOCOM*, 2007, pp. 19–27.
[9] A. Proutiere, Y. Yi, and M. Chiang, "Throughput of random access without message passing," in *CISS*, 2008, pp. 509–514.
[10] X. Wu, R. Srikant, and J. R. Perkins, "Scheduling efficiency of distributed greedy scheduling algorithms in wireless networks," *IEEE Trans. Mob. Comput.*, vol. 6, no. 6, pp. 595–605, 2007.
[11] X. Lin and S. Rasool, " A Distributed Joint Channel-Assignment, Scheduling and Routing Algorithm for Multi-Channel Ad-hoc Wireless Networks," in *Proceedings of IEEE INFOCOM*, May 2007, pp. 1118–1126.
[12] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.
[13] Q. Li and R. Negi, "Scheduling in multi-hop wireless networks with priorities," *CoRR*, vol. abs/0901.2922, 2009.
[14] M. J. Neely, E. Modiano, and C. E. Rohrs, "Dynamic power allocation and routing for time varying wireless networks," in *Proceedings of IEEE INFOCOM*, 2003.

APPENDIX

Proof of Theorem 1: Let $x_l(t)$ denote the service received by link l during slot t. Thus, $x_l(t) = 0$ if l is not scheduled during the slot, and $x_l(t) = r_l$ otherwise.

The queue dynamics are as follows:

$$q_l(t+1) = q_l(t) + \lambda_l(t) - x_l(t) \tag{10}$$

We use the following Lyapunov function to prove queue-stability:

$$V_q(\overrightarrow{q}(t)) = \sum_{l\in\mathcal{L}} \frac{\widetilde{K}_l q_l(t)}{r_l} \left[\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k q_k(t)}{r_k}\right] \tag{11}$$

It can be seen that:

$$\begin{aligned}
&V_q(\overrightarrow{q}(t+1)) - V_q(\overrightarrow{q}(t)) = \sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l q_l(t+1)}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k q_k(t+1)}{r_k}\right)\right] - \sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k q_k(t)}{r_k}\right)\right] \\
&= \sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l (q_l(t)+q_l(t+1)-q_l(t))}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k (q_k(t)+q_k(t+1)-q_k(t))}{r_k}\right)\right] - \sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k q_k(t)}{r_k}\right)\right] \\
&= \sum_{l\in\mathcal{L}} \frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k q_k(t)}{r_k}\right) + \sum_{l\in\mathcal{L}} \frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k (q_k(t+1)-q_k(t))}{r_k}\right) + \sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l (q_l(t+1)-q_l(t))}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k q_k(t)}{r_k}\right)\right] \\
&\qquad + \sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l (q_l(t+1)-q_l(t))}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k (q_k(t+1)-q_k(t))}{r_k}\right)\right] - \sum_{l\in\mathcal{L}} \frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k q_k(t)}{r_k}\right) \\
&= \sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k (q_k(t+1)-q_k(t))}{r_k}\right)\right] + \sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l (q_l(t+1)-q_l(t))}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k q_k(t)}{r_k}\right)\right] \\
&\qquad + \sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l (q_l(t+1)-q_l(t))}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k (q_k(t+1)-q_k(t))}{r_k}\right)\right] \\
&= 2\sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k (q_k(t+1)-q_k(t))}{r_k}\right)\right] + \sum_{l\in\mathcal{L}} \left[\frac{\widetilde{K}_l (q_l(t+1)-q_l(t))}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k (q_k(t+1)-q_k(t))}{r_k}\right)\right] \\
&\text{since } k\in\mathbf{I}(l) \implies l\in\mathbf{I}(k) \text{ from the symmetric conflicts assumption}
\end{aligned} \tag{12}$$

Denote by $\mathcal{L}'(t)$ the set of links l for which $q_l(t) \geq r_l$. This set of links participates in the scheduling process for slot t. From the scheduler definition, it follows that, for all $l \in \mathcal{L}'(t)$:

$$\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k x_k(t)}{r_k} \geq \widetilde{K}_l \tag{13}$$

If $\overrightarrow{\lambda}$ lies within the γ_l-reduced rate region, then, by assumption, there exists some scheduling algorithm that achieves stability with load vector $\overrightarrow{\lambda'} = \overrightarrow{\gamma}^{-1} \bullet \overrightarrow{\lambda}$.

Noting that in the current case $\gamma_l = \frac{1}{\widetilde{K}_l}$, this implies existence of an average service-rate vector $\overline{x_l}$ for all links l satisfying the following for some $\varepsilon > 0$:

$$(1+\varepsilon)\widetilde{K}_l\lambda_l \leq \overline{x_l} \text{ for all links } l \tag{14}$$

$$\sum_{k\in\mathbf{I}(l)} \frac{\overline{x_k}}{r_k} \leq \max\{1, K_l\} = \widetilde{K}_l \text{ for all links } l \tag{15}$$

Le $Q_{init} = \max_{l\in\mathcal{L}} \frac{\widetilde{K}_l q_l(0)}{r_l}$. Let $y_{min} = \min_{l\in\mathcal{L},\ \lambda_l>0} \frac{\widetilde{K}_l\lambda_l}{r_l}$. Using (12):

$$\begin{aligned}
&E[V_q(\overrightarrow{q}(t+1)) - V_q(\overrightarrow{q}(t))|\overrightarrow{q}(t)] \\
&= 2\sum_{l\in\mathcal{L}} \frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)} E\left[\frac{\widetilde{K}_k(q_k(t+1)-q_k(t))}{r_k}\Big|\overrightarrow{q}(t)\right]\right) + \sum_{l\in\mathcal{L}} E\left[\frac{\widetilde{K}_l(q_l(t+1)-q_l(t))}{r_l}\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k(q_k(t+1)-q_k(t))}{r_k}\right)\Big|\overrightarrow{q}(t)\right] \\
&\leq 2\sum_{l\in\mathcal{L}} \frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)} \frac{\widetilde{K}_k}{r_k} E\left[\lambda_k(t) - x_k(t)\Big|\overrightarrow{q}(t)\right]\right) + \sum_{l\in\mathcal{L}} E\left[\frac{\widetilde{K}_l\lambda_l(t)}{r_l}\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k(t)}{r_k}\right] \\
&= 2\sum_{l\in\mathcal{L}} \frac{\widetilde{K}_l q_l(t)}{r_l}\left(E\left[\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k(t)}{r_k}\right] - E\left[\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k x_k(t)}{r_k}\Big|\overrightarrow{q}(t)\right]\right) + \sum_{l\in\mathcal{L}} E\left[\frac{\widetilde{K}_l\lambda_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k(t)}{r_k}\right)\right] \\
&\leq 2\sum_{l\in\mathcal{L}} \frac{\widetilde{K}_l q_l(t)}{r_l}\left(E\left[\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k(t)}{r_k}\right] - E\left[\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k x_k(t)}{x_k}\Big|\overrightarrow{q}(t)\right]\right) + C_1 \\
&= 2\sum_{l\in\mathcal{L}} \frac{\widetilde{K}_l ql(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k}{r_k} - E\left[\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k x_k(t)}{r_k}|\overrightarrow{q}(t)\right]\right) + C_1 \\
&= 2\sum_{l\in\mathcal{L}'(t)} \frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k}{r_k} - E\left[\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k x_k(t)}{r_k}|\overrightarrow{q}(t)\right)\right]\right) \\
&\qquad + 2\sum_{l\in\mathcal{L}\setminus\mathcal{L}'(t)} \frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k}{r_k} - E\left[\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k x_k(t)}{r_k}|\overrightarrow{q}(t)\right]\right) + C_1 \\
&\leq 2\sum_{l\in\mathcal{L}'(t)} \frac{\widetilde{K}_l q_l(t)}{r_l}\left[\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k}{r_k} - \sum_{k\in\mathbf{I}(l)}\frac{\overline{x_k}}{r_k}\right) + \left(\sum_{k\in\mathbf{I}(l)}\frac{\overline{x_k}}{r_k}\right) - E\left[\sum_{k\in\mathbf{I}'(l)}\frac{\widetilde{K}_k x_k(t)}{r_k}|\overrightarrow{q}(t)\right]\right] \\
&\qquad + 2E\left[\sum_{l\in\mathcal{L}\setminus\mathcal{L}'(t)}\frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k}{r_k}\right)\right] + C_1 \\
&\leq 2\sum_{l\in\mathcal{L}'(t)} \frac{\widetilde{K}_l q_l(t)}{r_l}\left[-\varepsilon\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k}{r_k}\right)\right] + 2\sum_{l\in\mathcal{L}\setminus\mathcal{L}'(t)}\frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k}{r_k}\right) + C_1 \\
&\qquad \text{using (13), (14), (15)} \\
&\leq 2\sum_{l\in\mathcal{L}'(t)} \frac{\widetilde{K}_l q_l(t)}{r_l}[-\varepsilon y_{min}] + 2\sum_{l\in\mathcal{L}'(t)} \varepsilon y_{min} Q_{init} - 2\sum_{l\in\mathcal{L}\setminus\mathcal{L}'(t)}\frac{\widetilde{K}_l q_l(t)}{r_l}\varepsilon y_{min} \\
&\quad + 2\sum_{l\in\mathcal{L}\setminus\mathcal{L}'(t)}\frac{\widetilde{K}_l q_l(t)}{r_l}\varepsilon y_{min} + 2\sum_{l\in\mathcal{L}\setminus\mathcal{L}'(t)}\frac{\widetilde{K}_l q_l(t)}{r_l}\left(\sum_{k\in\mathbf{I}(l)}\frac{\widetilde{K}_k\lambda_k}{r_k}\right) + C_1
\end{aligned}$$

(compensating for links with $\lambda_l = 0$ by adding the $2\sum_{l\in\mathcal{L}'(t)} \varepsilon y_{min} Q_{init}$ term, and also

subtracting and adding back $2\sum_{l\in\mathcal{L}\setminus\mathcal{L}'(t)}\frac{\widetilde{K}_l q_l(t)}{r_l}\varepsilon y_{min}$ to handle links in $\mathcal{L}\setminus\mathcal{L}'(t)$)

$$\leq 2\sum_{l\in\mathcal{L}}\frac{\widetilde{K}_l q_l(t)}{r_l}(-\varepsilon y_{min})+C_2$$

where $r_{max}=\max_{l\in\mathcal{L}} r_l$, $C_1=\frac{|\mathcal{L}|\widetilde{K}^2\eta I_{max}}{(\min_{l\in\mathcal{L}} r_l)^2}$, and $C_2=C_1+2\varepsilon y_{min}|\mathcal{L}|Q_{init}+2\varepsilon y_{min}\widetilde{K}|\mathcal{L}|+2|\mathcal{L}|\widetilde{K}^2 I_{max}$. Using the above in conjunction with Lemma 2 from [14] suffices to prove stability.

Proof of Theorem 2: Suppose the set of valid priority values that can be assigned to links is $\mathcal{M}$ where $|\mathcal{M}|=m$. Thus, for all l: $1\leq\phi(l)\leq m$.

Let $\mathcal{S}_i=\{l|l\in\mathcal{L},\phi(l)=i\}$. Evidently for a link $l\in\mathcal{S}_i$, $\mathbf{H}(l)\subseteq\bigcup_{j=1}^{i-1}\mathcal{S}_j$, and $\mathbf{Z}(l)\subseteq\mathcal{S}_i$. Conversely, for a link $l\in\mathcal{S}_j$, $\mathbf{I}(l)\setminus(\mathbf{H}(l)\cup\mathbf{Z}(l))\subseteq\bigcup_{i=j+1}^{m}\mathcal{S}_i$.

The queue dynamics are as follows:

$$q_l(t+1)=q_l(t)+\lambda_l(t)-x_l(t) \tag{16}$$

where $x_l(t)$ can be either 0 or r_l depending on whether l was scheduled during slot t or not.

We use the following Lyapunov function:

$$V_q(\overrightarrow{q}(t))=\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{q_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{q_k(t)}{r_k}\right] \tag{17}$$

It can be seen that:

$$\begin{aligned}
&V_q(\overrightarrow{q}(t+1))-V_q(\overrightarrow{q}(t))=\\
&\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t+1)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{q_k(t+1)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{q_k(t+1)}{r_k}\right]\\
&\qquad-\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{q_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{q_k(t)}{r_k}\right]\\
&=\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)+(q_l(t+1)-q_l(t))}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{q_k(t)+(q_k(t+1)-q_k(t))}{r_k}\right.\\
&\qquad\left.+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{q_k(t)+(q_k(t+1)-q_k(t))}{r_k}\right]-\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{q_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{q_k(t)}{r_k}\right]\\
&=\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{q_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{q_k(t)}{r_k}\right]\\
&\qquad+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}\right]\\
&\qquad+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{(q_l(t+1)-q_l(t))}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{q_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{q_k(t)}{r_k}\right]\\
&+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{(q_l(t+1)-q_l(t))}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}\right]\\
&\qquad-\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{q_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{q_k(t)}{r_k}\right]\\
&=\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}\right]\\
&\qquad+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{i+1}\sum_{l\in\mathcal{S}_i}\frac{(q_l(t+1)-q_l(t))}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{q_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{q_k(t)}{r_k}\right]
\end{aligned}$$

$$
\begin{aligned}
&+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{(q_l(t+1)-q_l(t))}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}\right]\\
&\leq\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}\right]\\
&\quad+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{(q_l(t+1)-q_l(t))}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{q_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{q_k(t)}{r_k}\right]\\
&\qquad+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{\lambda_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{\lambda_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{\lambda_k(t)}{r_k}\right]\\
&\leq\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}\right]\\
&\quad+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{I}(l)\setminus(\mathbf{H}(l)\cup\mathbf{Z}(l))}\frac{(q_k(t+1)-q_k(t))}{r_k}\right]\\
&\qquad+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{\lambda_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{\lambda_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{\lambda_k(t)}{r_k}\right]\\
&=\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}+\sum_{k\in\mathbf{Z}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}\right]\\
&\quad+\sum_{i=1}^{m-1}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\sum_{k\in\mathbf{I}(l)\setminus(\mathbf{H}(l)\cup\mathbf{Z}(l))}\frac{(q_k(t+1)-q_k(t))}{r_k}\right]\\
&\qquad+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{\lambda_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{\lambda_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{\lambda_k(t)}{r_k}\right] \qquad (18)
\end{aligned}
$$

since $k\in\mathbf{H}(l)\implies l\in\mathbf{I}(k)\setminus(\mathbf{H}(k)\cup\mathbf{Z}(k))$ and $k\in\mathbf{Z}(l)\implies l\in\mathbf{Z}(k)$ and $l\in\mathcal{S}_m\implies\mathbf{I}(l)\setminus(\mathbf{H}(l)\cup\mathbf{Z}(l))=\phi$

Since we have a prioritized maximal scheduler with thresholds, it follows that if $q_l(t)\geq r_l$ then either l is scheduled in slot t or else some $k\in(\mathbf{H}(l)\cup\mathbf{Z}(l))\setminus\{l\}$ must be scheduled in slot t. Therefore for all participating links l:

$$\sum_{k\in\mathbf{H}(l)}\frac{x_k(t)}{r_k}+\sum_{k\in\mathbf{Z}(l)}\frac{x_k(t)}{r_k}\geq 1 \qquad (19)$$

Since the vector $\overrightarrow{\lambda}+\varepsilon_o\overrightarrow{r}$ lies within the $[\overrightarrow{\gamma}]$ reduced rate region, there is some positive ε such that $(1+\varepsilon)\overrightarrow{\gamma}^{-1}\bullet(\overrightarrow{\lambda}+\varepsilon_o\overrightarrow{r})$ is stabilizable by some scheduling algorithm. Noting that $\gamma_l=\frac{1}{H_l}$ in the current case, it follows that:

$$\sum_{k\in\mathbf{H}(l)}\left((1+\varepsilon)\frac{H_k(\lambda_k+\varepsilon_o r_k)}{r_k}\right)+\sum_{k\in\mathbf{Z}(l)}\left((1+\varepsilon)\frac{H_k(\lambda_k+\varepsilon_o r_k)}{r_k}\right)\leq\widetilde{h}_l \qquad (20)$$

Hence:

$$
\begin{aligned}
&\left[\sum_{k\in\mathbf{H}(l)}\frac{\lambda_k+\varepsilon_o r_k}{r_k}+\sum_{k\in\mathbf{Z}(l)}\frac{\lambda_k+\varepsilon_o r_k}{r_k}\right]\\
&\qquad=\left(\sum_{k\in\mathbf{H}(l)}\frac{1}{H_k(1+\varepsilon)}\frac{H_k(1+\varepsilon)(\lambda_k+\varepsilon_o r_k)}{r_k}+\sum_{k\in\mathbf{Z}(l)}\frac{1}{H_k(1+\varepsilon)}\frac{H_k(1+\varepsilon)(\lambda_k+\varepsilon_o r_k)}{r_k}\right)\\
&\qquad\leq\frac{1}{\widetilde{h}_l(1+\varepsilon)}\left(\sum_{k\in\mathbf{H}(l)}\frac{H_k(1+\varepsilon)(\lambda_k+\varepsilon_o r_k)}{r_k}+\sum_{k\in\mathbf{Z}(l)}\frac{H_k(1+\varepsilon)(\lambda_k+\varepsilon_o r_k)}{r_k}\right)\quad\text{(using (3))} \qquad (21)\\
&\qquad<\left(\frac{1}{\widetilde{h}_l}\right)\widetilde{h}_l=1\quad\text{(using (20))}\\
&\therefore\sum_{k\in\mathbf{H}(l)}\frac{\lambda_k}{r_k}+\sum_{k\in\mathbf{Z}(l)}\frac{\lambda_k}{r_k}<1-\varepsilon_o(\min_{l\in\mathcal{L}}|\mathbf{H}(l)\cup\mathbf{Z}(l)|)\leq 1-\varepsilon_o\\
&\quad(\forall l\in\mathcal{L},l\in\mathbf{Z}(l)\text{ and hence }|\mathbf{Z}(l)|\geq 1)
\end{aligned}
$$

Therefore:

$$
\begin{aligned}
&E[V_q(\overrightarrow{q}(t+1)) - V_q(\overrightarrow{q}(t))|\overrightarrow{q}(t)] \\
&\leq \sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}E\left[\sum_{k\in\mathbf{H}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}+\sum_{k\in\mathbf{Z}(l)}\frac{(q_k(t+1)-q_k(t))}{r_k}\Big|\overrightarrow{q}(t)\right] \\
&\quad+\sum_{i=1}^{m-1}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}E\left[\sum_{k\in\mathbf{I}(l)\setminus(\mathbf{H}(l)\cup\mathbf{Z}(l))}\frac{(q_k(t+1)-q_k(t))}{r_k}\Big|\overrightarrow{q}(t)\right] \\
&\qquad+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}E\left[\sum_{l\in\mathcal{S}_i}\frac{\lambda_l(t)}{r_l}\left[\sum_{k\in\mathbf{H}(l)}\frac{\lambda_k(t)}{r_k}+\frac{1}{2}\sum_{k\in\mathbf{Z}(l)}\frac{\lambda_k(t)}{r_k}\right]\right]\text{ using (18))} \\
&\leq \sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}E\left[\sum_{k\in\mathbf{H}(l)}\frac{\lambda_k(t)-x_k(t)}{r_k}+\sum_{k\in\mathbf{Z}(l)}\frac{\lambda_k(t)-x_k(t)}{r_k}\Big|\overrightarrow{q}(t)\right] \\
&\quad+\sum_{i=1}^{m-1}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}E\left[\sum_{k\in\mathbf{I}(l)\setminus(\mathbf{H}(l)\cup\mathbf{Z}(l))}\frac{\lambda_k(t)-x_k(t)}{r_k}|\overrightarrow{q}(t)\right]+C_1\text{ where }C_1=\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\frac{\eta I_{max}|\mathcal{L}|}{(\min_{l\in\mathcal{L}}r_l)^2} \\
&=\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\left(\sum_{k\in\mathbf{H}(l)}\frac{\lambda_k}{r_k}+\sum_{k\in\mathbf{Z}(l)}\frac{\lambda_k}{r_k}\right)-E\left[\left(\sum_{k\in\mathbf{H}(l)}\frac{x_k(t)}{r_k}+\sum_{k\in\mathbf{Z}(l)}\frac{x_k(t)}{r_k}\right)\Big|\overrightarrow{q}(t)\right]\right] \\
&\quad+\sum_{i=1}^{m-1}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\left(\sum_{k\in\mathbf{I}(l)\setminus(\mathbf{H}(l)\cup\mathbf{Z}(l))}\frac{\lambda_k}{r_k}\right)-E\left[\left(\sum_{k\in\mathbf{I}(l)\setminus(\mathbf{H}(l)\cup\mathbf{Z}(l))}\frac{x_k(t)}{r_k}\right)\Big|\overrightarrow{q}(t)\right]\right]+C_1 \\
&\leq \sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}(-\varepsilon_o)+\sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\varepsilon_o+\sum_{i=1}^{m-1}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}|\mathbf{I}(l)\setminus(\mathbf{H}(l)\cup\mathbf{Z}(l))|+C_1 \\
&\qquad\text{using (19) and (21), and compensating for links } l\notin\mathcal{L}'(t) \text{ by adding } \sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\varepsilon_o \\
&\leq \sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}(-\varepsilon_o)+\sum_{i=1}^{m-1}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}(I_{max}-1)+C_2 \\
&\leq \sum_{l\in\mathcal{S}_m}\frac{q_l(t)}{r_l}(-I_{max})+\sum_{i=1}^{m-1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}(-\varepsilon_o)+\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i}(I_{max}-1)\right]+C_2 \\
&= \sum_{l\in\mathcal{S}_m}\frac{q_l(t)}{r_l}(-I_{max})+\sum_{i=1}^{m-1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[-\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i}\right]+C_2 \\
&\leq \sum_{l\in\mathcal{S}_m}\frac{q_l(t)}{r_l}(-I_{max})+\sum_{i=1}^{m-1}\sum_{l\in\mathcal{S}_i}\frac{q_l(t)}{r_l}\left[-\left(\frac{I_{max}}{\varepsilon_o}\right)\right]+C_2 \quad (\because I_{max}\geq 1,\varepsilon_o<1)
\end{aligned}
$$

where $C_2 = C_1 + \sum_{i=1}^{m}\left(\frac{I_{max}}{\varepsilon_o}\right)^{m-i+1}\varepsilon_o|\mathcal{S}_i|$.
Using the above in conjunction with Lemma 2 from [14] suffices to prove stability.