This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

Distributed and Local Scheduling Algorithms for mmWave Integrated Access and Backhaul

Swaroop Gopalam, Stephen V. Hanly and Philip Whiting
School of Engineering, Macquarie University, Sydney, Australia
{swaroop.gopalam,stephen.hanly,philip.whiting}@mq.edu.au

***Abstract*—We consider the stability region of a mmWave integrated access and backhaul (IAB) network with stochastic arrivals and time-varying link rates. In the scheduling of links, we consider a limit on the number of RF chains, and the half-duplex constraint which occurs due to the wireless backhaul links. We characterize the stability region, and propose a back-pressure policy for the IAB network under the RF chains and half-duplex constraints. To implement the back-pressure policy, it is required to compute the maximum weighted schedule, which is a complex problem in general. For the IAB network, we present a distributed message passing scheme to compute the maximum weighted schedule, with almost linear complexity. We also investigate a class of local scheduling policies for the IAB network, which have a smaller stability region in general, but require no message passing. We characterize the stability region for the local class, and show that it is same as the global stability region, if the link rates are un-varying. We provide a bound on the gap between local and global regions when the links are time varying. We propose a local max-weight algorithm which achieves the stability region for the local class, and we present numerical results.**

***Index Terms*—Millimeter wave cellular IAB network, RF chains constraint, Local max-weight scheduling algorithm, Distributed back-pressure scheduling algorithm**

I. Introduction

mmWave cellular networks are expected to play a key role in the next generation wireless communications (5G) [1]. They are capable of delivering very high rates, due to the vast amount of spectrum available in the mmWave band. However, wireless communication at mmWave frequencies comes with two major challenges, including 1) high isotropic propagation loss, and 2) sensitivity to blockage by the objects in the environment. To overcome the high propagation losses, directional communication using beam-forming is being considered for mmWave cellular. High beam-forming gains are achievable by implementing large antenna arrays in a tiny area (which is possible due to the small wavelengths). The mmWave cell sizes are expected to be small due to the high propagation loss and blocking, and ultra dense deployments of Next Generation Node Bases (gNBs) are being considered to provide universal coverage.

It is prohibitively expensive to provide fibre backhaul support to all the gNBs under dense deployments. Hence, there has been recent interest in multi-hop relaying (or self backhauling) in mmWave cellular networks as a potential solution. Notably, as part of its standardization efforts, 3GPP has completed a recent study item on the potential solutions for efficient operation of integrated access and wireless backhaul (IAB) for NR [2]. The study emphasizes the joint consideration of radio-access and backhaul for mmWave cellular networks.

In this paper, we consider a multi-hop IAB network, where a fraction of gNBs are deployed with dedicated fiber backhaul links, referred to as IAB donors [2]. The other gNBs (referred to as IAB nodes) relay their backhaul data over wireless mmWave links, possibly in multiple hops to an IAB donor. According to [2], an IAB node establishes a link to a parent node (either another IAB node or a donor) by following the same initial access procedure as a user equipment (UE), and the central unit at the IAB donor establishes a forwarding route to the IAB node via the parent. Therefore, traffic of a UE is forwarded along this established route from the IAB donor to the UE (in downlink). The 3GPP study identified two topologies for the operation of mmWave IAB, 1) spanning tree and 2) directed acyclic graph topologies [2]. We focus on the spanning tree topology, where each IAB node has a unique parent node which forwards the traffic to the IAB node. However, our model does not preclude having, in practice, backup provisioning to deal with link or node failures. In our model, we would consider these failures as a topology change issue, where a forwarding route would need to be established via a new parent node. An example of an IAB network can be seen in Figure 1.

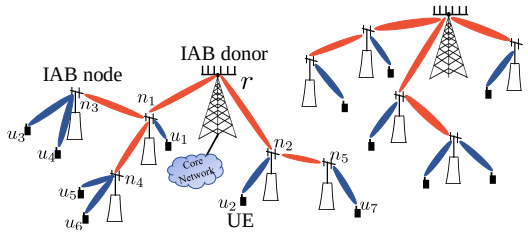


Fig. 1. mmWave IAB network. The red links are mmWave backhaul links and the blue links are mmWave access links

We consider the problem of dynamic resource allocation (or scheduling), which is a key challenge in the control of multi-hop IAB networks [2], [3]. We consider resource allocation in a in-band backhauling scenario (i.e., backhaul and access use the same frequencies), which accommodates tighter interworking access and backhaul links [2]. In an in-band

This research was supported in part by the Australian Research Council under Discovery Project DP180103550. It was also supported by a iMQRTP PhD scholarship from Macquarie University.

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

scenario, the half-duplex constraint imposes restrictions on the links that can be active simultaneously. There are also other constraints on link activation due to beam-forming. Multiple links at a gNB can be scheduled simultaneously using beam-forming. However, there is a limit on the number of beams, given by the number of RF chains at the gNB, which is the maximum number of links that can be activated simultaneously at the gNB.

For this 3GPP mmWave IAB setup, we consider a dynamic scenario with stochastic packet arrivals and time-varying links. The system is stable iff the queues do not buildup indefinitely at any of the gNBs. We provide distributed scheduling algorithms for the mmWave IAB network, and also characterize the stability region, as the set of arrival rate vectors for which stability is possible (under some scheduling algorithm).

Although resource allocation in mmWave multi-hop networks has been considered in the literature, only a few papers have provided distributed solutions. Utility maximization for joint routing and resource allocation for mmWave multi-hop networks was considered in [4]–[10]. Other works considered queue based models [9], [11]–[14]. Utility maximization was used for path selection and scheduling in [9], [13], [14], and for congestion control and scheduling in [11], [12]. Dynamic path selection algorithms for topology management in mmWave networks were studied in [15]–[18]. With the exception of [15], centralized solutions were provided in these works.

In wireless scheduling literature, distributed scheduling algorithms for various networks have been proposed [19]–[25]. Much of the work is focused on networks under a *primary interference constraint* [19], [20], [24], [25]. Under these constraints, any two links sharing a common node (either a transmitter or a receiver) are not allowed to be scheduled simultaneously. The remainder considered more general *conflict constraints* [21]–[23]. Under the conflict constraint, a given link cannot be scheduled with any of the links in a predetermined set. In [19], [20], maximal scheduling based distributed algorithms were proposed, which were shown to achieve only a fraction of capacity in general. In [21], a distributed version of greedy maximal scheduling was proposed for a wireless network with time varying link rates. In [22], [23], Carrier Sense Multiple Access (CSMA) based distributed scheduling algorithms were proposed and shown to achieve full capacity. Pick-and-Compare based distributed algorithms were proposed in [24], [25].

The results of the above mentioned papers cannot be directly applied to the mmWave IAB networks for the following reasons: 1) With the exception of [21], time varying link rates are not considered, which is a key concern. 2) The limit on RF chains imposes a novel form of constraint, which cannot modelled using a conflict constraint. For an example, consider a gNB with 2 RF chains and serving 3 downlinks $\{\ell_1, \ell_2, \ell_3\}$. Even though no two links in $\{\ell_1, \ell_2, \ell_3\}$ conflict with each other, only the links in the following sets $\{\ell_1, \ell_2\}, \{\ell_1, \ell_3\}, \{\ell_2, \ell_3\}$ can be scheduled simultaneously. This form of constraint has not been considered in aforementioned distributed scheduling literature.

A. Contributions

In this paper, we provide distributed scheduling algorithms which can be applied to a mmWave IAB network, with RF chains constraints. Firstly, we propose a back-pressure algorithm for the mmWave IAB network, under the half-duplex and RF chains constraints. The back-pressure algorithm relies on computing the maximum weighted schedule in each slot, which is potentially highly complex. Our first contribution (in the following) addresses this problem.

The proposed message passing scheme for the back-pressure algorithm requires the messages to travel up the tree and back. Due to this messaging overhead, the round-trip delays incurred can be significant for large networks. Hence, we proceed to investigate a class of local scheduling policies, which do not require any exchange of messages other than listening to the parent node. If the parent gNB $p(n)$ of gNB n transmits to n in a slot t, then gNB n does not transmit in slot t. Otherwise, gNB n schedules on its downlinks according to a local scheduling rule, which uses information of queues at n and the rates on the downlinks.

We present our contributions as follows.

1) We provide a distributed message passing scheme to compute the maximum weighted schedule in the IAB network. We use this scheme to implement a novel back-pressure-type algorithm that achieves the full stability region for the IAB network. Computing the maximum weighted schedule is a combinatorial problem but we provide a greedy algorithm to solve it which is of very low complexity, almost linear in the number of links. (See statement 2 of Theorem 1.)
2) We characterize the stability region for a class of local policies, and show that it is the intersection of individual local stability regions at each gNB n. (See Theorem 4.).
3) We present a local version of the max-weight algorithm for mmWave IAB networks, and show that it achieves the stability region for the class of local policies.
4) We show that the stability region of the local class is the same as that of global policies, provided the link rates are unvarying. (See Theorem 5).
5) We provide a bound on the gap between the local and global stability regions, when the links are time-varying. (See Theorem 6).
6) Using numerical simulations, we show that the performance (expected delays) of the proposed local algorithm is very close to that of global policies, such as back-pressure and max-weight algorithms for the considered IAB scenarios.

B. Structure of the paper

The rest of the paper is as follows. In section II, we introduce the IAB system model, including the graph representation, SDMA model and link scheduling constraints. In section III, we introduce the queueing model for the IAB network, and provide the definition of a stationary scheduling algorithm. In section IV, we provide the distributed message passing scheme to compute the maximum weighted schedule in an IAB network. In section V, we characterize the stability

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

region of the IAB network. In section VI, we propose a global back-pressure policy which achieves the stability region characterized in section V. We also discuss a distributed implementation using the message passing scheme developed in section IV. In section VII, we investigate the class of local scheduling policies, and characterize their stability region. We show that the stability region of the local class is same as the global stability region, if the link rates are un-varying. We also provide a bound on the gap between the global and local stability regions, when links are time-varying. In section VIII, we propose a local max-weight algorithm which achieves the stability region of the local class. In section IX, we compare the delay performance of various scheduling policies in an IAB network scenario.

II. System Model

We consider the spanning tree topology and represent the IAB network as the rooted tree graph $G \equiv (\mathcal{U} \bigcup \mathcal{N}, \mathcal{L}, r)$ where $\mathcal{N}$ is the set of all the gNBs, $\mathcal{U}$ is the set of all the UEs and $\mathcal{L}$ is the set of all the wireless links, i.e., the backhaul links and the access links. Here, the single IAB donor is the root node r. For an IAB node $n \in \mathcal{N} - \{r\}$, let $p(n)$ denote the upstream node of n in the path from n to r. We refer to $p(n)$ as the the parent of node n. An IAB node $n \in \mathcal{N} - \{r\}$ gets backhaul data from node $p(n)$ via the backhaul link b_n connecting n and $p(n)$.

We adopt a slotted model with slots $t \in \mathbb{Z}_+$. In a slot t, each link $l \in \mathcal{L}$ is associated with a rate $\mu_l(t)$. $\mu_l(t)$ is the number of packets which can be transmitted over link l, provided l is scheduled in slot t. Let $\boldsymbol{\mu}(t) := [\mu_l(t)]_{l \in \mathcal{L}}$.

Assumption 1.
- *For a mmWave access link ℓ between a gNB and a UE, we assume that $\mu_\ell(t)$ is a random variable taking values from $\{0, 1, \ldots, \mu_{\max}\}$. The effects of fading are modelled using the time-varying link rates. Here, $\mu_\ell(t) = 0$ corresponds to the small-scale outages[1] (due to tracking errors, beam mis-alignment etc.,) which last in the order of milli-seconds.*
- *For a gNB-gNB backhaul link b, we assume that $\mu_b(t) \in \{0, \bar{\mu}_b\}$ is a random variable. The backhaul links are highly directional LOS wireless links between two static gNBs, and hence we do not model fading. Small-scale outages are modelled using the 0 state.*
- *We assume that the link rate process $\{\boldsymbol{\mu}(t)\}_{t=1}^{\infty}$ is stationary and ergodic. The stationary probability of being in a state $\boldsymbol{\mu} \in \mathcal{M}$ is denoted by $\pi_{\boldsymbol{\mu}}$. Here, $\mathcal{M}$ is the set of all possible link rate vectors.*

We use a binary variable $s_l(t) \in \{0, 1\}$ to indicate the scheduled state of a link $l \in \mathcal{L}$. $s_l(t) = 1$ indicates l is scheduled in slot t, and $s_l(t) = 0$ indicates otherwise.

A. Downlink beamforming Model and Half-duplex constraints

Let M_n be the number of RF chains at a gNB $n \in \mathcal{N}$. The gNB n can beamform to up to M_n downstream nodes simultaneously. A downstream node can be a UE or a IAB node receiving backhaul from n. Let $\mathcal{L}_n$ denote the set of all downstream links of a gNB $n \in \mathcal{N}$. The limit on number of RF chains imposes the following scheduling constraint (1).

$$\sum_{\ell \in \mathcal{L}_n} s_\ell(t) \leq M_n, \forall n \in \mathcal{N} \tag{1}$$

Let b_n denote the backhaul link connecting n to $p(n)$. The half-duplex constraint on link scheduling is given in (2).

$$s_{b_n}(t) \times s_\ell(t) = 0, \forall \ell \in \mathcal{L}_n, n \in \mathcal{N} - \{r\} \tag{2}$$

III. Queuing Model and Scheduling Policies

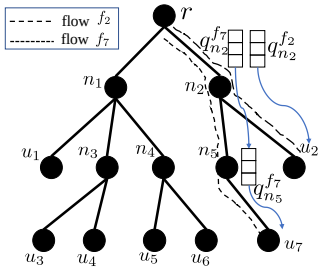


Fig. 2. Graph representation of the left IAB network in Figure 1.

The queuing model is as follows. Each source-destination pair $r - u$ is associated with a flow f, which is the set of nodes in G in the path from r to u including both. The packets destined for u have to routed along flow f through the network (see Figure 2). Let $\mathcal{F}$ denote the set of all the flows in the network. A gNB n maintains a queue q_n^f corresponding to each flow f that passes through n, i.e., $n \in f$. Let $q_n^f(t)$ denote the number of packets in the queue of flow f, at node n, in slot t. Note that $q_n^f(t) = 0, \forall t \in \mathbb{Z}_+$ if $n \notin f$.

The packet arrivals of each flow $f \in \mathcal{F}$ occur as an exogenous process at the root r. Let $a_r^f(t) \in \mathbb{Z}_+$ denote the number of packets of flow f arriving during slot t at node r, and let $d_n^f(t)$ denote the number of departures in slot t from flow f's queue at gNB n. For an IAB node $n \in \mathcal{N} - \{r\}$, packets arrive over the backhaul link b_n from node $p(n)$ (see Figure 2). Hence, arrivals into the queue q_n^f are the departures from $q_{p(n)}^f$. The queue recursion equations can be written as follows

$$q_r^f(t+1) = q_r^f(t) + a_r^f(t) - d_r^f(t)$$
$$q_n^f(t+1) = q_n^f(t) + d_{p(n)}^f(t) - d_n^f(t), \ \forall \ n \in \mathcal{N} - \{r\}$$

A. Scheduling Policy for the IAB Network

Let $\mathcal{S}$ denote the set of $\boldsymbol{s} = \{s_l\}_{l \in \mathcal{L}} \in \{0, 1\}^{|\mathcal{L}|}$ such that,

$$\sum_{l \in \mathcal{L}_n} s_l \leq M_n, \forall n \in \mathcal{N} \tag{3}$$

$$s_{b_n} s_l = 0, \forall l \in \mathcal{L}_n, n \in \mathcal{N} \tag{4}$$

$\mathcal{S}$ is the set of all the feasible schedules. In each slot t, a scheduling policy has to choose a schedule $\boldsymbol{s}(t) \in \mathcal{S}$. In this paper, we consider the class of stationary scheduling policies.

Definition 1. *In each slot t, a stationary scheduling policy chooses a schedule $s(t) \in \mathcal{S}$, such that $s(t)$ only depends on*

[1]Note that this does not include blocking. The link outages caused due to blocking can last in the order of seconds, and a change in topology is necessary to address it. Once the new topology is established, the algorithm proposed in the paper can be applied to stabilize the queues.

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

the current state $[\mathcal{Q}(t), \boldsymbol{\mu}(t)]$*, Here* $\boldsymbol{s}(t) := [s_\ell(t)]_{\ell\in\mathcal{L}}$*,* $\boldsymbol{\mu}(t) = [\mu_l(t)]_{l\in\mathcal{L}}$ *and* $\mathcal{Q}(t) = [q_n^f(t)]_{[f,n]\in\mathcal{F}\times\mathcal{N}}$*.*

- *A deterministic stationary policy chooses according to a mapping from the* $[\boldsymbol{q}, \boldsymbol{\mu}] \in \mathbb{Z}_+^{|\mathcal{F}|\times|\mathcal{N}|} \times \mathcal{M}$ *to a schedule* $\boldsymbol{s} \in \mathcal{S}$*.*
- *Under a randomized stationary policy, given a state* $[\mathcal{Q}, \boldsymbol{\mu}] \in \mathbb{Z}_+^{|\mathcal{F}|\times|\mathcal{N}|} \times \mathcal{M}$*, the schedule is the output of a random variable* $Y_{\mathcal{Q},\boldsymbol{\mu}}$ *with the probability distribution* $\mathcal{P}_{\mathcal{Q},\boldsymbol{\mu}}$ *on* $\mathcal{S}$*. The probability distribution* $\mathcal{P}_{\mathcal{Q},\boldsymbol{\mu}}$ *only depends on the state* $\{\mathcal{Q}, \boldsymbol{\mu}\}$*. At each time* t*, the schedule* $s(t)$ *is chosen independently according to the distribution* $\mathcal{P}_{\mathcal{Q}(t),\boldsymbol{\mu}(t)}$*.*

IV. Finding the maximum weighted schedule in IAB network

Several scheduling policies in the literature consider the maximum weight optimization (5) for choosing the schedule. Here, $w_l(t)$ is a function of $[\mathcal{Q}(t), \boldsymbol{\mu}(t)]$. For example, $w_l(t) := \mu_l(t)$ corresponds to the maximum rate scheduling policy. In later sections, we introduce the global back-pressure policy for the IAB network, which considers maximizing the back-pressure objective, as a maximum weight optimization.

$$\boldsymbol{s}(t) = \arg\max_{\boldsymbol{s}\in\mathcal{S}} w_l(t)s_l(t) \tag{5}$$

Finding a maximum weighted schedule, i.e., a solution to (5) is integral to these scheduling policies. In this section, we obtain a dynamic programming based forward-backward algorithm for obtaining a maximum weighted schedule in almost linear time.

A similar dynamic programming (DP) approach for finding a maximum weighted independent set on a tree is given in [26]. However, there are key differences between our maximum weighted schedule problem for an IAB tree network, and the independent set problem in [26], due to multiple RF chains. The DP sub-problem in [26] is to choose the maximum of two values, whereas in our case with multiple RF chains, we will show that the DP sub-problem at each gNB is a combinatorial optimization problem. For our problem, the schedule at each DP step, has to choose among all the sets (of links) with cardinality less than or equal to the number of RF chains. For this combinatorial problem, we provide an optimal greedy algorithm with complexity almost linear in the number of RF chains.

Given a weight w_l on each link $l \in \mathcal{L}$, we consider the following maximum weight optimization (6).

$$\max_{\boldsymbol{s}\in\mathcal{S}} \sum_{l\in\mathcal{L}} w_l s_l \tag{6}$$

For each $n \in \mathcal{N}\bigcup\mathcal{U}$, let G_n be the induced sub-graph of G formed using the vertex set containing n and its descendants in G. Let E_n denote the set of links of G_n. Consider the maximum weight optimization on the sub-graph G_n as the following integer linear program ILP (7)-(9).

$$\max_{[s_l]_{l\in E_n}\in\{0,1\}^{|E_n|}} \sum_{l\in E_n} w_l s_l \quad \text{s.t.} \tag{7}$$

$$\sum_{l\in\mathcal{L}_m} s_l \le M_m, \forall m \in \mathcal{N}\cap G_n \tag{8}$$

$$s_{b_m} s_l = 0, \forall l \in \mathcal{L}_m, m \in \mathcal{N}\cap G_n - \{n\} \tag{9}$$

Here, (8) is the RF chains constraint at each gNB m in G_n, and (9) is the half-duplex constraint at each gNB m in $G_n - \{n\}$. Let v_n denote the optimal value of ILP (7)-(9). Note, by definition, v_r is the optimal value of ILP (6).

A. Reduced complexity via an Inductive approach

Consider a set $\psi \subseteq \mathcal{L}_n$ such that $|\psi| \le M_n$. Let $w(\psi)$ be the maximum value of (7) subject to the additional constraints (10), (11) along with (8), (9).

$$s_l = 1, \forall l \in \psi \tag{10}$$

$$s_l = 0, \forall l \in \mathcal{L}_n - \psi \tag{11}$$

Note that $w(\psi)$ is the maximum weight on G_n given that the links in ψ are scheduled at gNB n. It follows that

$$v_n = \max_{\psi\subseteq\mathcal{L}_n} w(\psi) \text{ s.t. } |\psi| \le M_n \tag{12}$$

To obtain $w(\psi)$, we use the following Lemma 1. Refer to Appendix B, for a proof.

Lemma 1. *For each* $\psi \subseteq \mathcal{L}_n$ *such that* $|\psi| \le M_n$*,*

$$w(\psi) = \sum_{l\in\psi} w_l + \sum_{m\in R(\psi)} \sum_{o\in C(m)} v_o + \sum_{m\in C(n)-R(\psi)} v_m \tag{13}$$

where, $C(m)$ *is the set of receiving nodes of links in* $\mathcal{L}_m$ *for* $m \in \mathcal{N}$*, and* $C(m) = \phi$ *for* $m \in \mathcal{U}$*.* $R(\psi) := \{m \in C(n) : (n, m) \in \psi\}$ *is the set of receiving nodes of links in* ψ*.*

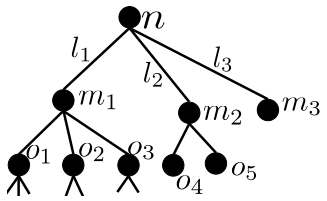


Fig. 3. Graph G_n for illustration of inductive solution. In this example, $w(\{l_1, l_3\}) = \sum_{i=1,3} w_{l_i} + v_2 + \sum_{i=1}^{3} v_{o_i}$. Suppose $M_n = 3$, then $v_n = \max_{\psi\in 2^{\{l_1,l_2,l_3\}}} w(\psi)$, where 2^A represents the power set of a set A.

(12) combined with Lemma 1 provides an inductive solution for finding the maximum weight v_n. This approach requires the knowledge of values $\{v_m\}_{m\in C(n)}, \{\sum_{o\in C(m)} v_o\}_{m\in C(n)}$, i.e, values from two generations of descendants from n. Hence, dynamic programming can be applied to compute the maximum weight, by solving (12) at each gNB n.

This approach has much reduced complexity compared to directly solving (6), since the size of $\mathcal{S}$ grows exponentially in the total number of links $\sum_{n\in\mathcal{N}} \mathcal{L}_n$, whereas $\{\psi \subseteq \mathcal{L}_n : |\psi| \le M_n\}$ of (12) grows exponentially only in number of links $|\mathcal{L}_n|$ at that level, in general.

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

B. Complexity of optimization (12)

As briefly mentioned earlier, the possibilities for set ψ in (12) can grow exponentially in $|\mathcal{L}_n|$, in general. There are $\sum_{i=0}^{M_n} \binom{|\mathcal{L}_n|}{i}$ possibilities for set ψ. Hence, the direct approach to solve (12) leads to a high dimensional search when then values of $M_n, |\mathcal{L}_n|$ are large. In the following, we present Algorithm 1 to solve (12), and show that it has a computational complexity of $O(|\mathcal{L}_n| \log |\mathcal{L}_n|)$.

Algorithm 1 Solver for sub-problem (12)

function LOCAL SOLVER($n, \{v_m, \sum_{o\in C(m)} v_o\}_{m\in C(n)}$)
  $w'_{(n,m)} := w_{(n,m)} - v_m + \sum_{o\in C(m)} v_o$ for $m \in C(n)$.
  Sort the links $l_i \in \mathcal{L}_n$ such that $w'_{l_1} \geq \ldots \geq w'_{l_{|\mathcal{L}_n|}}$.
  $k^* := \max\{k \leq M_n : w'_{l_k} \geq 0\}$ & $\psi^* := \{l_1, \ldots, l_{k^*}\}$.
  $v_n := \sum_{l\in\psi^*} w'_l + \sum_{m\in C(n)} v_m$.
  return v_n, ψ^*
end function

Theorem 1. *1) Algorithm 1 solves optimization (12) and has a computational complexity of $O(|\mathcal{L}_n| \log |\mathcal{L}_n|)$, where $|\mathcal{L}_n|$ is the set of downlinks at gNB n.*
2) Algorithm 2 solves the maximum weight optimization (6), and has a computational complexity of $O(\sum_{n\in\mathcal{N}} |\mathcal{L}_n| \log |\mathcal{L}_n|)$.

Proof. Since $\sum_{m\in C(n)-R(\psi)} = \sum_{m\in C(n)} - \sum_{m\in R(\psi)}$, (13) can be re-written as

$$w(\psi) = \sum_{m\in R(\psi)} (w_{(n,m)} + \sum_{o\in C(m)} v_o - v_m) + \sum_{m\in C(n)} v_m$$
$$= \sum_{l\in\psi} w'_l + \sum_{m\in C(n)} v_m \quad (14)$$

where $w'_{(n,m)} := w_{(n,m)} - v_m + \sum_{o\in C(m)} v_o$.

Hence, (12) is equivalent to maximizing $\sum_{l\in\psi} w'_l$ subject to $|\psi| \leq M_n$. To maximize $\sum_{l\in\psi} w'_l$, Algorithm 1 chooses ψ^* as the links $l \in \mathcal{L}_n$ with the highest non-negative weights w'_l such that $|\psi^*| \leq M_n$. It follows that ψ^* also solves (12).

With the exception of sorting, all the operations in Algorithm 1 are of linear computational complexity. The sort has a complexity of $O(|\mathcal{L}_n| \log(|\mathcal{L}_n|))$.

Recall that v_r at root r is the optimal value of (6). Computation in Algorithm 2 only involves executing Algorithm 1 at each gNB $n \in \mathcal{N}$. Hence, 2) of Theorem 1 is immediate. □

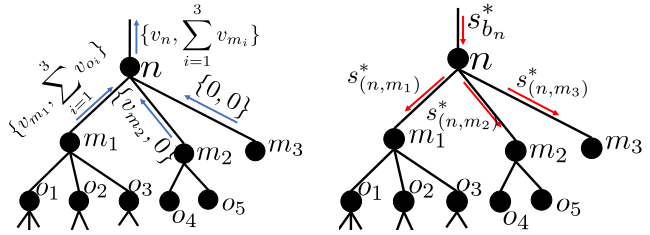
C. Implementation using Message Passing

Algorithm 2 presents the DP algorithm to compute the maximum weighted schedule, by locally applying Algorithm 1 to compute the solutions to the DP sub-problem in (12). In phase 1 of Algorithm 2, the solution to (12) (i.e., the maximum weight v_n and the corresponding local schedule ψ_n) at each gNB n is computed by applying Algorithm 1 in lines 8-9. The necessary information $\{v_n, \sum_{o\in C(m)} v_o\}_{m\in C(n)}$ which is required for the local computation at the parent node $p(n)$ is sent upstream as a message in line 10. This message passing is illustrated in Fig. 4(a).

In phase 2, the maximum weighted schedule is decided using the local schedules $\{\psi_n\}$ computed during phase 1. The root node starts the process by computing $[s_l]_{l\in\mathcal{L}_r}$ according to ψ_r. The other nodes n decide $[s_l]_{l\in\mathcal{L}_n}$ to be **0** if $s_{b_n} = 0$, i.e., backhaul link b_n is part of the max-weight schedule, in line 18. Otherwise, $[s_l]_{l\in\mathcal{L}_n}$ is chosen according to ψ_n in line 17. This process is illustrated in Fig. 4(b).

Algorithm 2 DP Algorithm for calculation of maximum weighted schedule

1: **Input** $\{w_l\}_{l\in\mathcal{L}}$
2: **Output** $\boldsymbol{s}^*$
Phase 1 – Upstream weight computation, starting from the leaves and going to the root.
3: **for** each leaf node n of G **do**
4: $v_n = 0$.
5: Send the values $\{0, 0\}$ to parent $p(n)$.
6: **end for**
7: **for** each non-leaf node n of G **do**
8: After receiving the messages from each child in $C(n)$, run Algorithm 1 with Arg$_n$:=$(n, \{v_m, \sum_{o\in C(m)} v_o\}_{m\in C(n)})$ as the input.
9: (v_n, ψ_n) = LOCAL SOLVER(Arg$_n$).
10: Send the values $\{v_n, \sum_{m\in C(n)} v_m\}$ to parent $p(n)$.
11: **end for**
Phase 2 – Downstream schedule computation, starting from the root and going to the leaves.
12: $s^*_l = 1,\ \forall l \in \psi_r$ & $s^*_l = 0,\ \forall l \in \mathcal{L}_r - \psi_r$.
13: Send the value $s^*_{(r,m)}$ to each child $m \in C(r)$.
14: **for** $n \in \mathcal{N} - \{r\}$ **do**
15: **if** $s^*_{b_n} = 0$ **then** // b_n is the backhaul link connecting n to its parent $p(n)$.
16: $s^*_l = 1,\ \forall l \in \psi_n$ & $s^*_l = 0,\ \forall l \in \mathcal{L}_n - \psi_n$.
17: **else**
18: $s^*_l = 0, \forall l \in \mathcal{L}_n$
19: **end if**
20: Send the value $s^*_{(n,m)}$ to each child $m \in C(n)$.
21: **end for**



(a) Upstream message passing for weight computation

(b) Downstream message passing for schedule computation

Fig. 4. An illustration of the forward-backward message passing for computing maximum weighted schedule in Algorithm 2.

V. Stability Region of the IAB Network

In this section, we define and characterize the stability region of the IAB network. Following other works [27]–[29], we assume that the process for exogenous packet arrivals $\{a^f_r(t)\}_{t=0}^{\infty}$ of each flow $f \in \mathcal{F}$ is a stationary and ergodic process, with a mean $\nu^f := \mathbb{E}[a^f_r(1)]$. We also assume that $\mathbb{E}[(a^f_r(1))^2] < \infty$ for each $f \in \mathcal{F}$.

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

Definition 2. *We consider the system to be stable under a scheduling policy if and only if* $\limsup_{t\to\infty}\sum_{\tau=0}^{t-1}\mathbb{E}[\mathcal{Q}(\tau)]/t<\infty$.

Definition 3. *We consider the system to be stabilizable if and only if there exists a stationary scheduling policy under which the system is stable.*

We introduce the necessary terminology for characterizing the stability region. Let $\boldsymbol{\nu}:=[\nu^f]_{f\in\mathcal{F}}$ denote the arrival rate vector. For $\ell\in\mathcal{L}$, Let $\mathcal{F}_\ell\subseteq\mathcal{F}$ denote the set of flows whose path includes ℓ, and $\nu_\ell:=\sum_{f\in\mathcal{F}_\ell}\nu^f$ denotes the average arrival rate of packets into link ℓ.

For a given $\boldsymbol{\mu}\in\mathcal{M},\boldsymbol{s}\in\mathcal{S}$, we denote the corresponding rate vector as $\boldsymbol{\mu}\odot\boldsymbol{s}$, where $\odot$ is element-wise product. For a given link state $\boldsymbol{\mu}$, let $\mathcal{C}_{\boldsymbol{\mu}}:=\{\boldsymbol{\mu}\odot\boldsymbol{s}\}_{\boldsymbol{s}\in\mathcal{S}}$ denote the set of the rate vectors corresponding to the feasible schedules.

The stability region is given by the set Λ as stated in Theorem 2.

$$\Lambda:=\left\{\boldsymbol{\nu}=[\nu^f]_{f\in\mathcal{F}}:[\nu_\ell]_{\ell\in\mathcal{L}}\in\sum_{\boldsymbol{\mu}\in\mathcal{M}}\pi_{\boldsymbol{\mu}}Conv(\mathcal{C}_{\boldsymbol{\mu}})\right\}\quad(15)$$

where $Conv(.)$ is the convex hull of the given set.

A point $\boldsymbol{\nu}$ is in interior of Λ if and only if there exists a $\delta>0$ such that $\boldsymbol{\nu}+\delta\mathbf{1}\in\Lambda$, where $\mathbf{1}=[1]_{f\in\mathcal{F}}$.

Theorem 2. *1) The system is not stabilizable if* $\boldsymbol{\nu}\notin\Lambda$.
2) Given a $\boldsymbol{\nu}$ *in interior of* Λ*, there exists a stationary scheduling policy which which will stabilize the system.*

Proof. The proof follows from standard arguments such as in the proof of Theorem 5 in [28]. □

VI. Global back-pressure policy

In this section, we present a back-pressure policy which stabilizes the system for any arrival vector within Λ. The construction of the policy is based on [28]–[30]. First, we introduce the necessary notation. Let $\mathcal{L}^a\subseteq\mathcal{L}$ represent the set of all the gNB-UE access links.

Under the back-pressure policy, we define the following weights on links. For an access link $l\in\mathcal{L}^a\bigcap\mathcal{L}_n$, define

$$w_l^{bp}(t):=\mu_l(t)\max_{f\in\mathcal{F}_l}q_n^f(t)\quad(16)$$

For a backhaul link b_n, connecting gNB $p(n)$ to gNB n, we define the weight as

$$w_{b_n}^{bp}(t):=\mu_{b_n}(t)\max_{f\in\mathcal{F}_{b_n}}\left(q_{p(n)}^f(t)-q_n^f(t)\right)\quad(17)$$

In each slot t, the back-pressure policy chooses $\boldsymbol{s}(t)$ as the optimal solution of following optimization (18)

$$\max_{\boldsymbol{s}(t)\in\mathcal{S}}\sum_{l\in\mathcal{L}}w_l^{bp}(t)s_l(t)\quad(18)$$

Under the back-pressure policy, if a backhaul link b_n is scheduled in slot t, then only the packets corresponding to flow $f^*:=\arg\max_{f\in\mathcal{F}_{b_n}}(q_{p(n)}^f(t)-q_n^f(t))$ are sent over link b_n in slot t. In other words, $d_{p(n)}^f(t)=0$ for each $f\in\mathcal{F}_{b_n}-\{f^*\}$.

Theorem 3. *The back-pressure policy, as defined in (18), stabilizes the network for any arrival rate vector* $\boldsymbol{\nu}$ *interior of* Λ.

Proof. Refer to [28], [29] for proof. The DRPC policy in [28], [29] is equivalent to the back-pressure policy in this paper. □

A. Finding the maximum back-pressure schedule

Firstly, to evaluate the weight $w_l^{bp}(t)$ for $l\in\mathcal{L}_n$, the only queue information required is the queue lengths at n, and the children gNBs of n, in the case where l is a backhaul link. Hence, $w_l^{bp}(t)$ can be evaluated locally at n for each $l\in\mathcal{L}_n$, given that the children gNBs of n communicate the necessary queue length information.

To implement the back-pressure algorithm, as in (18), it is required to find a schedule $\boldsymbol{s}(t)\in\mathcal{S}$ such that $\sum_{l\in\mathcal{L}}w_l^{bp}(t)s_l(t)$ is maximized. Note that this is a maximum weight optimization, and Algorithm 2 can be implemented to solve it in a distributed manner. The message passing scheme depicted in Fig. 4 can be adopted for such an implementation.

At this point, we have obtained the first two contributions listed in page 2. In what follows, we address the final four.

VII. Local policies and their stability region

In the previous section, we have provided an optimal back-pressure scheduling policy which achieves any arrival rate vector within the stability region. We have also shown that a distributed forward-backward message passing algorithm can be used for its implementation. However, such a scheme involves communicating the queue lengths upstream for the calculation of weights on links in the first step. Once the weights are calculated, the messages need to passed up and down the tree for computing the schedule. This round-trip communication can result in a severe message overhead when the network gets large.

Hence, in this section, we consider a class of local scheduling policies where the scheduling decisions are made locally at the gNBs. For this class, scheduling does not require any exchange of messages other than listening to the parent node. If the parent gNB $p(n)$ of gNB n transmits to n in a slot t, then gNB n does not transmit in slot t. Otherwise, gNB n schedules on its downlinks according to a local scheduling rule, which uses information of queues at n and the rates on the downlinks. We characterize the stability region of this class of policies in Theorem 4. We propose a local algorithm, which we show achieves any arrival rate vector in the stability region characterized in Theorem 4.

We make the following stronger Assumption 2 on arrival and link processes for the analysis of local policies. Parts 1) and 2) of Assumption 2 are required for Lyapunov stability arguments by considering 1-step conditional drift. We use independence of the link processes in part 3) to derive a *local characterization* of stability region for local policies in (21) in the following section.

Note that under these assumptions, the state process $[\mathcal{Q}(t),\boldsymbol{\mu}(t)]_{t\in\mathbb{Z}_+}$ is a time homogeneous Markov chain under any stationary policy.

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

Assumption 2. *1) For each $f \in \mathcal{F}$, the arrival process $\{a_r^f(t)\}_{t=0}^{\infty}$ is an i.i.d sequence of random variables, satisfying $\mathbb{E}[(a_r^f(t))^2] < \infty$.*

2) Let $\boldsymbol{\mu}_n(t) := [\mu_\ell(t)]_{\ell \in \mathcal{L}_n}$. For each $n \in \mathcal{N}$, $\{\boldsymbol{\mu}_n(t)\}_{t=0}^{\infty}$ is an i.i.d sequence of random variables.

3) The arrival processes are independent across $f \in \mathcal{F}$, and independent of the links process $\boldsymbol{\mu}(t)$. The link processes $\boldsymbol{\mu}_n(t)$ are independent across $n \in \mathcal{N}$.

Consider the class $\mathcal{P}$ of local stationary scheduling policies which make scheduling decisions as follows. The decision process starts at the root, the root r makes a decision $\boldsymbol{s}_r(t) := [s_l(t)]_{l \in \mathcal{L}_r}$ based on the local information $\boldsymbol{\mu}_r(t), \mathcal{Q}_r(t) := [q_r^f(t)]_{f \in \mathcal{F}}$. For every other node n, the decision is made as follows

1) If $s_{b_n}(t) = 1$ (i.e., parent node $p(n)$ has decided to schedule backhaul link b_n), then the links in $\mathcal{L}_n$ are not scheduled i.e., $\boldsymbol{s}_n(t) = \mathbf{0}$
2) If $s_{b_n}(t) = 0$ (i.e., parent node $p(n)$ has decided to not schedule backhaul link b_n), then $\boldsymbol{s}_n(t) := [s_l(t)]_{l \in \mathcal{L}_n}$ is chosen such that $\sum_{l \in \mathcal{L}_n} s_l(t) \leq M_n$ based on the local information $\boldsymbol{\mu}_n(t), \mathcal{Q}_n(t) := [q_n^f(t)]_{f \in \mathcal{F}}$.

It is clear that the the scheduling policies in $\mathcal{P}$ do not violate the half-duplex or the RF chains constraints, i.e., they are feasible.

Due to point 3) of Assumption 2, it is clear the policies in $\mathcal{P}$ satisfy

$$\mathbb{E}[s_{b_n}(t) | \boldsymbol{\mu}_n(t)] = \mathbb{E}[s_{b_n}(t)], \forall n \in \mathcal{N} - \{r\} \tag{19}$$

where $\boldsymbol{\mu}_n(t) := [\mu_l(t)]_{l \in \mathcal{L}_n}$. Since $s_{b_n}(t) \in \{0, 1\}$, (19) is equivalent to the statement that the scheduling decision of backhaul link b_n is made independently of the link states of the downstream links of gNB n. Due to this property, the local class only achieves a subset of the stability region Λ. We illustrate this in section VII-C, and provide an upper-bound on the gap between the stability regions in section VII-D.

A. Stability region of $\mathcal{P}$

The property (19) along with point 3) of Assumption 2, leads to a decomposition of stability region of $\mathcal{P}$ into individual local stability regions corresponding to each gNB. The system is stable when the arrival rate vector is interior to each local stability region. We will now show this decomposition.

Consider the sub-network formed using node n and its set of downstream links $\mathcal{L}_n$. We say that a local schedule $\boldsymbol{s}_n := [s_l]_{l \in \mathcal{L}_n} \in \{0, 1\}^{|\mathcal{L}_n|}$ of the sub-network is *feasible* if and only if $\sum_{l \in \mathcal{L}_n} s_l \leq M_n$, i.e., number of activated links is less than the number of RF chains at gNB n. Let $\mathcal{S}_n$ denote the set of all the feasible local schedules $\boldsymbol{s}_n$. For a given link state $\boldsymbol{\mu}_n \in \mathcal{M}_n \subseteq \{0, \ldots, \mu_{\max}\}^{|\mathcal{L}_n|}$, we denote the rate vector corresponding to feasible schedule as $\boldsymbol{\mu}_n \odot \boldsymbol{s}_n$, where $\odot$ represents element-wise product. For a given link state $\boldsymbol{\mu}_n$, $\mathcal{C}_{\boldsymbol{\mu}_n} := \{\boldsymbol{\mu}_n \odot \boldsymbol{s}_n\}_{\boldsymbol{s}_n \in \mathcal{S}_n}$ is the set of rate vectors corresponding to the feasible schedules. We define the local stability region Λ_n as

$$\Lambda_n := \left\{ [\nu^f]_{f \in \mathcal{F}} : \frac{[\nu_l]_{l \in \mathcal{L}_n}}{1 - \nu_{b_n}/\bar{\mu}_{b_n}} \in \sum_{\boldsymbol{\mu}_n \in \mathcal{M}_n} \pi_{\boldsymbol{\mu}_n} Conv(\mathcal{C}_{\boldsymbol{\mu}_n}) \right\} \tag{20}$$

where $\bar{\mu}_{b_n}$ is defined in Assumption 1. Recall that b_n is the backhaul link connecting n and its parent. Since the root node r has wired backhaul (and hence no parent), treat $\nu_{b_r}/\bar{\mu}_{b_r}$ as zero in the expression for Λ_r. We define $\Lambda_{\mathcal{P}}$ as follows.

$$\Lambda_{\mathcal{P}} := \bigcap_n \Lambda_n \tag{21}$$

The stability region of the scheduling policies in class $\mathcal{P}$ is characterized by $\Lambda_{\mathcal{P}}$ as stated in Theorem 4.

Theorem 4. *Suppose Assumption 2 holds, then*

1) If $\boldsymbol{\nu} \notin \Lambda_{\mathcal{P}}$, then the system is unstable under any policy in $\mathcal{P}$.

2) If $\boldsymbol{\nu} + \delta[1]_{f \in \mathcal{F}} \in \Lambda_{\mathcal{P}}$ for some $\delta > 0$, then system is stable under some policy in $\mathcal{P}$.

Proof. For 1), we provide a proof by showing transience of the underlying state Markov chain $\{\mathcal{Q}(t), \boldsymbol{\mu}(t)\}_{t \in \mathbb{N}}$. For a detailed proof, see Lemma 3 in Appendix C.

For 2), we follow a standard setup. We construct a randomized stationary policy corresponding to the rate vector $\boldsymbol{\nu} + \delta[1]_{f \in \mathcal{F}}$. Stability under the constructed policy follows from standard Lyapunov drift arguments. For a detailed proof, see Lemma 4 in Appendix C. □

B. Optimality of class $\mathcal{P}$ for un-varying link states

Clearly, $\Lambda_{\mathcal{P}} \subseteq \Lambda$ since class $\mathcal{P}$ is a subset of class of all the stationary policies. However, when the link state $\boldsymbol{\mu}(t)$ is constant for all t, the two stability regions are the same. The result is stated in the following Theorem 5.

An intuitive explanation for this result is the following observation. The stability characterization of $\Lambda_{\mathcal{P}}$ results from property (19), (which is satisfied by all the policies in $\mathcal{P}$, under Assumption 2). If the link state is un-varying, then (19) holds for every stationary policy (and not just policies in class $\mathcal{P}$).

Theorem 5. *Suppose that the link rates are unvarying, i.e., $\boldsymbol{\mu}(t) = \boldsymbol{\mu}^d > \mathbf{0}, \forall t \in \mathbb{Z}_+$. Then, the stability region for the local class, $\Lambda_{\mathcal{P}}$ is same as the global stability region Λ, i.e.,*

$$\Lambda_{\mathcal{P}} = \Lambda \tag{22}$$

Proof. See Appendix D. □

C. Example with varying link states, where $\Lambda_{\mathcal{P}} \subset \Lambda$

We now demonstrate the other case, where $\Lambda_{\mathcal{P}}$ is a strict subset of Λ with an example.

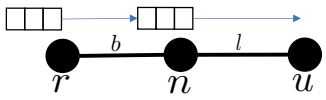


Fig. 5. Local policies stability region example

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

Consider a network with two links, one backhaul link b from r to n, and an access link l from n to u (see Fig. 5). In this network, there is one flow f going from r to u. In each slot t, a packet arrives independently at r with probability (w.p.) ν_f. For the link states, $\mu_b(t) = 1, \forall t \in \mathbb{Z}_+$. For l, in each slot t, $\mu_l(t)$ is either 0 or 1, independently w.p. 0.5. There are two queues in this network q_r^f at r and q_n^f at n. First, we present an optimal policy for this setup. Consider the policy which schedules link l in each slot t such that $\mu_l(t) = 1$, and schedules link b otherwise. It is immediate that $\mathbb{E}[\mu_l(t)s_l(t)] = \mathbb{E}[\mu_b(t)s_b(t)] = 0.5$. This policy stabilizes the network for any $\nu_f < 0.5$

Now, consider any policy in $\mathcal{P}$. Here, in each slot t, $s_b(t)$ is decided independently of $\mu_l(t)$. Suppose $s_b(t) = 1$ in a slot t, due to the half-duplex constraint, $s_l(t) = 0$. Hence, $\mathbb{E}[\mu_l(t)s_l(t)|s_b(t) = 1] = 0$. For the other case, suppose $s_b(t) = 0$ and $s_l(t) = 1$. Since $\mu_l(t) = 1$ w.p. 0.5, it follows that $\mathbb{E}[\mu_l(t)s_l(t)|s_b(t) = 0] = 0.5$. W.l.o.g. we assume that link b is scheduled in a slot t only if $q_r^f(t) > 0$. Hence, $\mathbb{E}[q_n^f(t+1) - q_n^f(t)|s_b(t) = 1] = 1$ and $\mathbb{E}[q_n^f(t+1) - q_n^f(t)|s_b(t) = 0] \geq -0.5$. Also, $\mathbb{E}[q_r^f(t+1) - q_r^f(t)|s_b(t) = 1] \geq \nu_f - 1$ and $\mathbb{E}[q_r^f(t+1) - q_r^f(t)|s_b(t) = 0] = \nu_f$. Let $P[s_b(t) = 1] = p_1(t)$. It follows that $\mathbb{E}[q_n^f(t+1) - q_n^f(t)] \geq 1.5p_1(t) - 0.5$ and $\mathbb{E}[q_r^f(t+1) - q_r^f(t)] = \nu_f - p_1(t)$. Hence,

$$3\mathbb{E}[q_r^f(t+1) - q_r^f(t)] + 2\mathbb{E}[q_n^f(t+1) - q_n^f(t)] \geq 3\nu_f - 1$$

It is clear that any policy in $\mathcal{P}$ cannot stabilize the network for $\nu_f > 1/3$. This demonstrates that the stability region of class $\mathcal{P}$ is a strict subset of the stability region Λ, for this example.

D. The gap between Λ and $\Lambda_{\mathcal{P}}$

In the previous section, we have illustrated that the global stability region Λ is larger than the $\Lambda_{\mathcal{P}}$ with an example. We now characterize an upper bound on the gap between the two stability regions Λ and $\Lambda_{\mathcal{P}}$, in Theorem 6.

Let $\bar{\boldsymbol{\mu}} := \sum_{\boldsymbol{\mu} \in \mathcal{M}} \pi_{\boldsymbol{\mu}} \boldsymbol{\mu}$ be the expected link rate vector. We define $\Lambda_n^{\bar{\boldsymbol{\mu}}}$ for $n \in \mathcal{N}$ as follows.

$$\Lambda_n^{\bar{\boldsymbol{\mu}}} = \{\boldsymbol{\nu} = [\nu^f]_{f \in \mathcal{F}} : \frac{[\nu_l]_{l \in \mathcal{L}_n}}{1 - \nu_{b_n}/\bar{\mu}_{b_n}} \in Conv(\mathcal{C}_{\bar{\boldsymbol{\mu}}_n})\} \quad (23)$$

where $\mathcal{C}_{\bar{\boldsymbol{\mu}}_n} := \{\bar{\boldsymbol{\mu}}_n \odot \boldsymbol{s}_n\}_{\boldsymbol{s}_n \in \mathcal{S}_n}$, and $\nu_{b_r}/\bar{\mu}_{b_r} = 0$.

Note that $\sum_{\boldsymbol{\mu}_n \in \mathcal{M}_n} \pi_{\boldsymbol{\mu}_n} \mathcal{C}_{\boldsymbol{\mu}_n} \supseteq \mathcal{C}_{\bar{\boldsymbol{\mu}}_n}$, where $\sum$ here represents the Minkowski addition. Hence,

$$\sum_{\boldsymbol{\mu}_n \in \mathcal{M}_n} \pi_{\boldsymbol{\mu}_n} Conv(\mathcal{C}_{\boldsymbol{\mu}_n}) \supseteq Conv(\mathcal{C}_{\bar{\boldsymbol{\mu}}_n}) \quad (24)$$

Hence from (20) and (23), we obtain $\Lambda_n \supseteq \Lambda_n^{\bar{\boldsymbol{\mu}}}$ for each $n \in \mathcal{N}$. Now from (21), we obtain $\Lambda \supseteq \Lambda_{\mathcal{P}} \supseteq \bigcap_n \Lambda_n^{\bar{\boldsymbol{\mu}}}$.

By applying Theorem 5 with $\boldsymbol{\mu}^d = \bar{\boldsymbol{\mu}}$, we obtain $\bigcap_n \Lambda_n^{\bar{\boldsymbol{\mu}}}$ is equal to $\Lambda^{\bar{\mu}}$ defined as follows.

$$\Lambda^{\bar{\boldsymbol{\mu}}} := \{\boldsymbol{\nu} = [\nu^f]_{f \in \mathcal{F}} : [\nu_l]_{l \in \mathcal{L}} \in Conv(\mathcal{C}_{\bar{\boldsymbol{\mu}}})\} \quad (25)$$

Hence, we obtain

$$\Lambda \supseteq \Lambda_{\mathcal{P}} \supseteq \Lambda^{\bar{\mu}} \quad (26)$$

The *relative gap* $G(\boldsymbol{a}, \boldsymbol{b})$ between two rate vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}_+^{|\mathcal{F}|}$ w.r.t $\boldsymbol{a} = [a^f]_{f \in \mathcal{F}}$ is defined as (27). Recall that $a_l = \sum_{f \in \mathcal{F}_l} a^f$ and $b_l = \sum_{f \in \mathcal{F}_l} b^f$ for $l \in \mathcal{L}$, where $\mathcal{F}_l \subseteq \mathcal{F}$ is the set flows whose path includes l.

$$G(\boldsymbol{a}, \boldsymbol{b}) = \max_{l \in \mathcal{L}} |(b_l - a_l)//a_l| \quad (27)$$

where, $(b_l - a_l)//a_l = 0$, if $a_l = 0, b_l = 0$, $(b_l - a_l)//a_l = \infty$, if $a_l = 0, b_l > 0$ and $(b_l - a_l)//a_l = (b_l - a_l)/a_l$ o.w.

We define the relative gap $G_{\mathcal{P}}$ between the stability regions Λ and $\Lambda_{\mathcal{P}}$ as

$$G_{\mathcal{P}} := \sup_{\boldsymbol{b} \in \Lambda} \inf_{\boldsymbol{a} \in \Lambda_{\mathcal{P}}} G(\boldsymbol{a}, \boldsymbol{b}) \quad (28)$$

Note that by definition, for each $\boldsymbol{b} \in \Lambda$, there exists a $\boldsymbol{a} \in \Lambda_{\mathcal{P}}$ such that b_l less than or equal to $(1 + G_{\mathcal{P}})a_l$, for each $l \in \mathcal{L}$. Hence, Λ is *larger* than $\Lambda_{\mathcal{P}}$, at most by a proportion of $G_{\mathcal{P}}$.

Theorem 6. *The gap $G_{\mathcal{P}}$ between the stability regions Λ and $\Lambda_{\mathcal{P}}$ is less than or equal to* $\max_{l \in \mathcal{L}} \frac{(E[(\mu_l(t) - \bar{\mu}_l)^2])^{1/2}}{\bar{\mu}_l}$.

Proof. Consider any $\boldsymbol{b} \in \Lambda$. By definition of convex hull, there exist vectors $[p_{\boldsymbol{s}|\boldsymbol{\mu}}] \in \mathbb{R}_+^{|\mathcal{S}|}$ for each $\boldsymbol{\mu} \in \mathcal{M}$ such that $\sum_{\boldsymbol{s} \in \mathcal{S}} p_{\boldsymbol{s}|\boldsymbol{\mu}} = 1$ and $\boldsymbol{b} = \sum_{\boldsymbol{\mu} \in \mathcal{M}} \sum_{\boldsymbol{s} \in \mathcal{S}} \pi_{\boldsymbol{\mu}} p_{\boldsymbol{s}|\boldsymbol{\mu}} \boldsymbol{\mu} \odot \boldsymbol{s}$.

Let $p_s := \sum_{\boldsymbol{\mu} \in \mathcal{M}} \pi_{\boldsymbol{\mu}} p_{\boldsymbol{s}|\boldsymbol{\mu}}$. Define $\boldsymbol{a} := \sum_{\boldsymbol{s} \in \mathcal{S}} p_s \bar{\boldsymbol{\mu}} \odot \boldsymbol{s}$. Clearly, $\boldsymbol{a} \in \Lambda_1^{\bar{\boldsymbol{\mu}}}$. From (26), $\boldsymbol{a} \in \Lambda_{\mathcal{P}}$. Note,

$$\begin{aligned}
G(\boldsymbol{a}, \boldsymbol{b}) &= \max_l |b_l - a_l|//a_l && (29)\\
&\leq \max_l \sum_{\boldsymbol{s} \in \mathcal{S}} \sum_{\boldsymbol{\mu} \in \mathcal{M}} \pi_{\boldsymbol{\mu}} p_{\boldsymbol{s}|\boldsymbol{\mu}} |\mu_l - \bar{\mu}_l| s_l//\bar{\mu}_l s_l && (30)\\
&\leq \max_l \sum_{\boldsymbol{s} \in \mathcal{S}} \sum_{\boldsymbol{\mu} \in \mathcal{M}} \pi_{\boldsymbol{\mu}} p_{\boldsymbol{s}|\boldsymbol{\mu}} |\mu_l - \bar{\mu}_l|/\bar{\mu}_l && (31)\\
&= \max_l \sum_{\boldsymbol{\mu} \in \mathcal{M}} \pi_{\boldsymbol{\mu}} |\mu_l - \bar{\mu}_l|/\bar{\mu}_l && (32)\\
&= \max_l \frac{\mathbb{E}[|\mu_l(t) - \bar{\mu}_l|]}{\bar{\mu}_l} && (33)\\
&\leq \max_l \frac{(\mathbb{E}[(\mu_l(t) - \bar{\mu}_l)^2])^{1/2}}{\bar{\mu}_l} && (34)
\end{aligned}$$

(34) follows from (33) by Jensen's inequality. Since the choice of $\boldsymbol{b}$ was arbitrary, the Theorem follows. □

Theorem 6 says that if the standard deviations of the link rates are small compared to their means, then the gap will be small.

VIII. A Local Max-weight Scheduling Algorithm

We present a local version of the max-weight scheduling algorithm which lies in class $\mathcal{P}$, and achieves the stability region of $\Lambda_{\mathcal{P}}$. This algorithm does not require any message passing, since it is in $\mathcal{P}$, and also acheives the global stability region when the link rates are un-varying, due to Theorem 5.

For a link $l \in \mathcal{L}_n$, let $Q_n^l(t) := \sum_{f \in \mathcal{F}_l} q_n^f(t)$ be the total number of packets queued at n to be sent over link l. Consider the set of links $\mathcal{L}_n'(t) \subseteq \mathcal{L}_n$ defined as follows; the set $\mathcal{L}_n'(t)$ contains a link $\ell \in \mathcal{L}_n$ iff either 1) ℓ is an access (gNB-UE) link such that $Q_n^\ell(t) > 0$ or 2) ℓ is a backhaul (gNB-gNB) link

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

such that $Q_n^\ell(t) \geq \mu_\ell(t) > 0$. We propose the local scheduling rule as the following optimization

$$\begin{aligned} \max \quad & \sum_{l \in \mathcal{L}_n} s_l(t)\mu_l(t)Q_n^l(t) \\ \text{s.t.} \quad & \sum_{l \in \mathcal{L}'_n(t)} s_l(t) \leq M_n \\ & s_l(t) \in \{0,1\}, \forall l \in \mathcal{L}'_n(t) \\ & s_l(t) = 0, \forall l \in \mathcal{L}_n - \mathcal{L}'_n(t) \end{aligned} \tag{35}$$

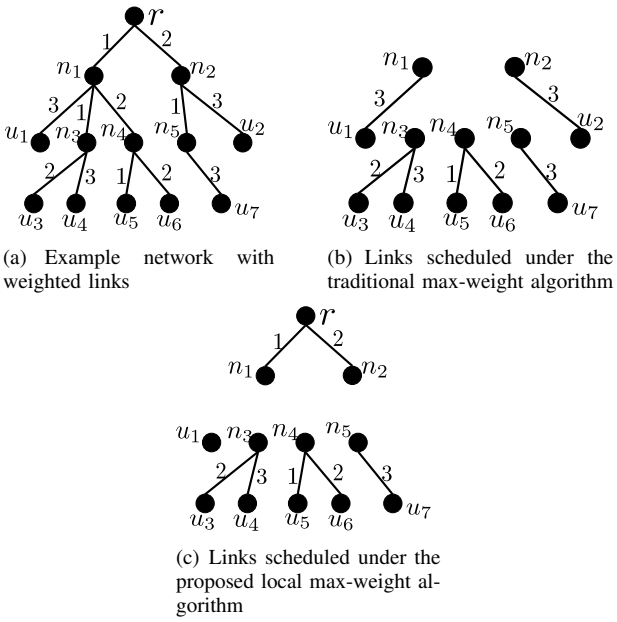
i.e., schedule the links with largest weights $w_l(t) := \mu_l(t)Q_n^l(t)$ from the set $\mathcal{L}'_n(t)$ subject to the limit on RF chains.

A. Local Max-weight Scheduling Policy

Recall $\boldsymbol{s}_n(t) := [s_l(t)]_{l \in \mathcal{L}_n}$. In each slot t, the root r starts the decision process by choosing $\boldsymbol{s}_r(t)$ according to the local scheduling rule (35). Other nodes $n \in \mathcal{N} - \{r\}$ choose $\boldsymbol{s}_n(t)$ depending on the value of $s_{b_n}(t)$ as follows

1) If the backhaul link into n, (i.e., b_n) is scheduled in slot t and $s_{b_n}(t) = 1$, then gNB n does not transmit and $\boldsymbol{s}_n(t) = \mathbf{0}$.
2) Otherwise, $\boldsymbol{s}_n(t)$ is chosen according to the local scheduling rule (35).

It can be noted that the scheduling decision at $n \in \mathcal{N} - \{r\}$ depends only on the local information $\boldsymbol{\mu}_n(t), Q_n^l(t)$ and the one bit of information $s_{b_n}(t)$ (which is a part of $\boldsymbol{s}_{p(n)}(t)$) from the parent $p(n)$. Hence, it can be implemented in a distributed manner by down-stream passing on the tree G. It is clear that the algorithm is feasible, since it does not violate constraints (1)-(2).



(a) Example network with weighted links

(b) Links scheduled under the traditional max-weight algorithm

(c) Links scheduled under the proposed local max-weight algorithm

Fig. 6. Numerical example

We illustrate the difference between the traditional max-weight algorithm and the proposed algorithm with the example in Figure 6. Consider the network and link weights $Q_n^l(t)\mu_l(t)$ given in Figure 6(a). Suppose each gNB node has 2 RF chains, i.e., $M_n = 2, \forall n \in \mathcal{N}$ and that $Q_n^l(t) \geq \mu_l(t), \forall l \in \mathcal{L}_n, n \in \mathcal{N}$. The links shown in Figure 6(b) are scheduled under the traditional max-weight algorithm, with a total weight of 17. The links shown in Figure 6(c) are scheduled under the proposed local max-weight algorithm, with a total weight of 14.

Theorem 7. *Given Assumption 2 holds, the system is stable under the proposed local max-weight algorithm for any $\boldsymbol{\nu}$ in the interior of $\Lambda_{\mathcal{P}}$.*

Proof. See Appendix E. □

We now provide an overview of the proof of Theorem 7. Using the standard 1-step conditional drift arguments for the quadratic Lyapunov function $\sum_{l \in \mathcal{L}_r}(Q_r^l(t))^2$, we show that the queues at gNB r are stable, i.e., $\limsup_{T\to\infty} \sum_{t=0}^{T} \sum_{f \in \mathcal{F}} \mathbb{E}[q_r^f(t)]/T < \infty$, in Lemma 5 in Appendix E.

For the rest of the gNBs $n \in \mathcal{N} - \{r\}$, we use an inductive argument to prove stability. We use the queue stability result for gNB r as the anchor for the induction. Using 1-step conditional drift arguments, we show that the long-term average queue length $\limsup_{T\to\infty} \sum_{t=0}^{T} \sum_{f \in \mathcal{F}} \mathbb{E}[q_n^f(t)]/T$ at a gNB n is bounded by a function of the long-term average queue lengths at gNBs n' in the path from n to r (not including n). We finish the proof by applying strong induction.

The induction step result for a gNB $n \in \mathcal{N} - \{r\}$ is derived by considering an appropriate Lyapunov function, which is not the standard quadratic function of queue lengths.

At this point, we have obtained the first 5 contributions listed on page 2. In the following numerical section, we address the final contribution.

IX. Numerical Results

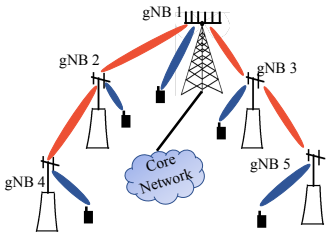


Fig. 7. Simulated IAB Network Topology

We consider the gNB setup shown in Fig. 7. Here, gNB 1 is the IAB donor, and the other gNBs $2-5$ are IAB nodes with the IAB topology shown in Fig. 7. The parameters for simulation are chosen as follows. The number of UEs associated at each gNB is chosen uniformly randomly between 4 and 11. For the gNB-gNB backhaul links, the distance is uniformly chosen between 340 and 440m. For the access gNB-UE links, the gNB-UE distance is chosen uniformly randomly between 0 and 200m. The distances for all the links are provided in section A of Appendix A. For the arrival process, the number of packet arrivals in each slot, corresponding to each UE (or flow) is a i.i.d Poisson random variable. The mean

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

is chosen to be the same for each UE. Other parameters are given in the following Table I.

Following [31], we model outage (due to tracking errors and beam misalignment etc.,) of each link as an alternating renewal process. For access links, the outage periods are geometrically distributed with mean 5.56 slots, and the non-outage periods are geometrically distributed with mean 50 slots. Therefore, the stationary probability of an access link being in outage is 0.1. For backhaul links, the outage periods are are geometrically distributed with mean 1.01 slots, and the non-outage periods are geometrically distributed with mean 100 slots.Therefore, the stationary probability of a backhaul link being in outage is 0.01.

Given that the link is not in outage, we model the fading process as follows. Following [32], we consider Rician fading for access links with K factor as 13 dB for LOS links and 6 dB for NLOS links. The fading realizations are generated independently in each slot. We assume that adaptive modulation and coding is used at the physical layer, and that the Shannon rate is achieved based on the SNR determined by fading.

TABLE I
SIMULATION PARAMETERS

Paramter	Value
Carrier frequency	23 GHz
Bandwidth	1 GHz
Propagation model	3GPP Urban Micro
Slot duration	125 μs
Packet size	100 Kb
RF chains	4
Noise spectral density	-174 dBm/Hz
gNB transmit power	24 dBm per RF chain
Beamforming gain	30 dB (for access), 40 dB (for backhaul)
Noise figure	5 dB (for gNB), 7 dB (for UE)
Number of UEs at gNBs 1-5	10, 5, 9, 10, 8

A. Scheduling policies under comparison

For comparison, we consider the following five scheduling policies.

1) *Local Maxweight:* We consider the local algorithm proposed in section VIII.
2) *Maxweight:* We consider the traditional max weight algorithm which requires global information. The max weight algorithm maximizes the objective $\sum_{l\in\mathcal{L}} s_l(t)\mu_l(t)Q_n^l(t)$ subject to the half-duplex and RF chains constraints (1-2).
3) *Backpressure:* We consider the back pressure algorithm proposed in section VI.
4) *Local PropFair:* This is another algorithm from class $\mathcal{P}$. The proportional fairness algorithm is implemented locally at a gNB n, provided backhaul link b_n is not scheduled. For scheduling, a gNB n chooses 4 links (since there are 4 RF chains) with the highest ratios of of instantaneous rate to average rate.
5) *Local Maxweight 2:* This is another algorithm from class $\mathcal{P}$. Here, the max weight algorithm implemented at a gNB n maximizes $\sum_{l\in\mathcal{L}_n} \mu_l(t)Q_n^l(t)s_l(t)$ provided the backhaul link b_n is not scheduled. There is a crucial difference between the proposed Local Maxweight algorithm and this scheme. Here, all the links in $\mathcal{L}_n$ are considered for scheduling at a node n, whereas in the proposed scheme scheduling of the backhaul links in $\mathcal{L}_n$ with $Q_n^l(t) < \mu_l(t)$ (i.e., small queue sizes) were avoided in favour of scheduling links at the downstream gNBs.

There are three types of UEs in the network in Fig. 7. 1) The UEs of gNB 1 are served by the root gNB 1. Hence packets of these UEs are not relayed over backhaul links. Here, the end-to-end delay for a UE is same as the scheduling delay (at gNB 1). 2) The UEs of gNB 2 and gNB 3. The packets of these UEs have to be relayed over 1 backhaul link. Here, the end-to-end delay is the sum of scheduling delay at gNB and backhaul delay (over 1 link). 3) The UEs of gNB 4 and gNB 5. The packets of these UEs have to be relayed over 2 backhaul links. Here, the end-to-end delay is the sum of scheduling delay at gNB and 2 backhaul delays (i.e., 2 hop). We present the average end-to-end delays of UEs of various gNBs w.r.t arrival rate in Fig. 8-Fig. 10.

Arrival rate (in packets/slot) is the expected rate of packet requests corresponding to each UE (or flow). The packet size (in Kb) and slot length (in μs) are presented in Table I. In what follows, arrival rate is the expected rate of traffic (in Mb/s) corresponding to each UE.

B. Results

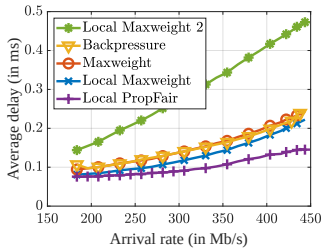
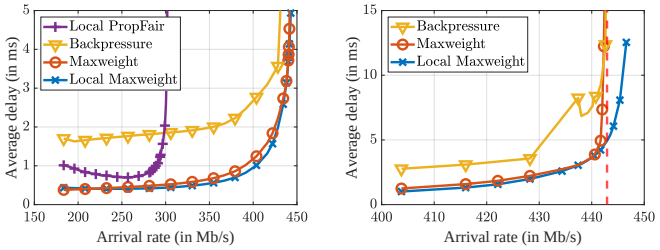


Fig. 8. Average delays of UEs at the root gNB 1



(a) Average end-to-end delay vs. arrival rate

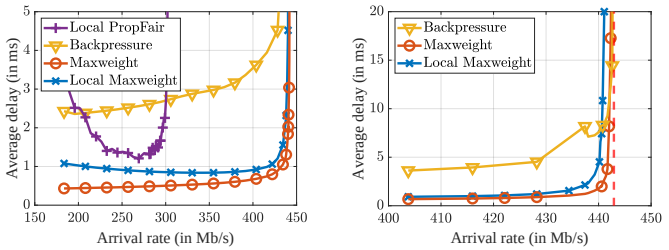
(b) Asymptotic behaviour at high arrival rates

Fig. 9. Average end-to-end delays of UEs at gNBs 2&3

The results for UEs at the root node gNB 1 are presented in Fig. 8. It can be observed that the average delays here are much smaller (compared to the delays of UEs at other gNBs, given in Fig. 9 and Fig. 10) for all the considered algorithms. The Local PropFair algorithm has the best performance of all the schemes, with the difference being more significant at higher arrival rates. However, the following results will show that the Local PropFair algorithm has a smaller stability region compared to the other schemes.

The results for the UEs of gNBs 2&3 are presented in Fig. 9. Local Maxweight 2 is unstable for the considered arrival rates,

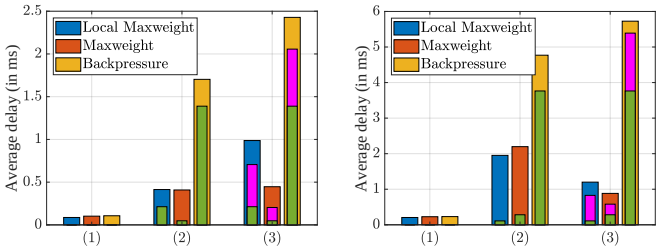
This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367



(a) Average end-to-end delay vs. arrival rate

(b) Asymptotic behaviour at high arrival rates

Fig. 10. Average end-to-end delays of UEs at gNBs 4&5



(a) For arrival rate of 214 Mb/s

(b) For arrival rate of 428 Mb/s

Fig. 11. Average end-to-end, queueing & scheduling delays. Here, (1) corresponds to UEs of gNB 1, (2) corresponds to UEs of gNBs 2 & 3, and (3) corresponds to UEs of gNBs 4 & 5.

and the end-to-end delays are unbounded. Hence, it is not plotted. It can be observed that the Local PropFair does not stabilize the network for arrival rates higher than 310 Mb/s. It can also be noted that the Local Maxweight algorithm has a comparable performance to the global scheme Maxweight and better performance than Backpressure. The asymptotic behaviour of the schemes is shown in Fig. 9(b). The dashed line is an upper-bound on the stability region. Here, it can be noted that Local Maxweight algorithm has bounded delays for arrival rates beyond the stability region. However for the UEs of gNBs 4&5 (given in Fig. 10), the delays start blowing up at lower rates than the global schemes, i.e., Maxweight and Backpressure. The system is unstable under the proposed algorithm at arrival rates beyond 442 Mb/s (see Fig. 10), even though the delays are bounded for UEs at gNBs 1, 2&3. A possible explanation for this is the hierarchical nature of the proposed algorithm; the scheduling at an upstream node is given priority over the downstream nodes.

The results for the UEs of gNBs 4&5 are presented in Fig. 10. Local Maxweight 2 is unstable for the considered arrival rates, and hence not plotted. It can be observed that the delays under the PropFair algorithm blow up at approximately 300 Mb/s. It can be noted again that Local Maxweight algorithm has a comparable performance to the Maxweight algorithm. The asymptotic behaviour of the schemes is shown in Fig. 10(b). The dashed line is an upper-bound on the stability region. Here, it can be noted that the delays (under Local Maxweight) start blowing up at lower rates than the Maxweight and Backpressure algorithms. This is the gap between the capacity achieved by the Local Maxweight algorithm and the global schemes.

For the considered simulation, the gap in capacity (between Backpressure and Local Maxweight algorithms) is small. Theorem 6 provides an explanation. The variation (over time) in the mmWave link states is small in the considered IAB scenario. Hence, the proposed Local Maxweight scheduling algorithm can be applied in such scenarios (i.e., where the link variations are small) without a significant loss in capacity.

Fig. 11 presents the average delays of UEs at arrival rates 214 Mb/s (in Figure 11(a)) and 428 Mb/s (in Fig. 11(b)). As mentioned earlier, the end-to-end delays to UEs at gNBs $2-5$ are the sum of scheduling delay and backhaul delay. For the UEs of gNBs $i = 2, 3$, the green bar represents the backhaul delay on the link connecting gNB 1 and gNB i. For the UEs of gNBs 4&5, the packets are routed along two backhaul links. The green bar represents the delay on the first hop, and the pink bar represents the delay on the second hop.

For both the arrival rates, the Maxweight has the best performance, Backpressure has the worst, and Local Maxweight has a performance closer to Maxweight. A notable observation is that, at low arrival rate 214 Mb/s, Local Maxweight algorithm has much higher backhaul delays relative to Maxweight (see (2),(3) in Fig. 11(a)). This is because the the backhaul links b are only considered for scheduling in Local Maxweight when the criterion $Q_n^b(t) \geq \mu_b(t)$ holds. Hence under Local Maxweight, the packets are queued until the backhaul link capacity is reached before transmission is attempted (even though a scheduling resource, i.e., RF chain, might be available earlier). This leads to idling under the Local Maxweight algorithm at low arrival rates. At higher arrival rates such as 428 Mb/s, it can be observed that this phenomenon does not have a significant impact on end-to-end delay, since the queues build up quicker at higher arrival rates.

Another observation is that Backpressure has higher delays than Local Maxweight in the considered scenario in Fig. 11. A possible explanation is when scheduling a backhaul link, under Backpressure, packets of only one flow (with largest differential backlog) are transmitted. Hence, to fully utilize a backhaul link (which must happen at reasonably high arrival rates since back-pressure achieves capacity), the flow queue sizes at each node in the path have to be be higher than or equal to the backhaul link rates, which are large.

We now present the cumulative distribution of end-to-end delays under various schemes at arrival rates 214 and 428 Mb/s.

Fig. 12(a) presents the distribution of end-to-end delays for UEs of gNB 1. Here, it can be observed that the results of considered algorithms are very similar.

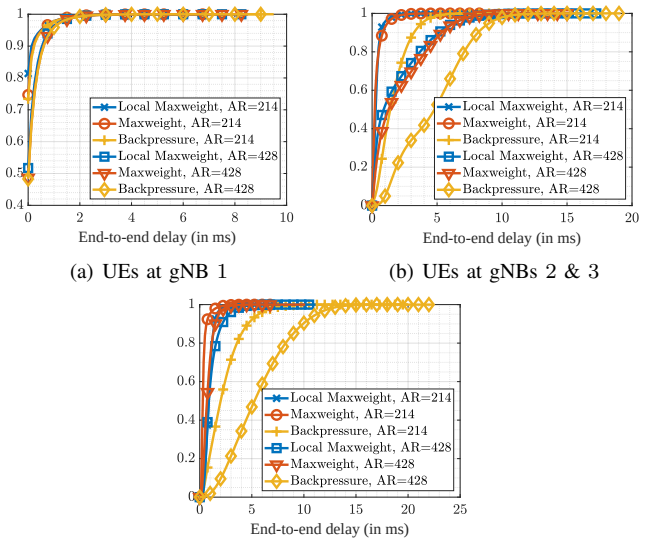
Fig. 12(b) presents the distribution for UEs of gNBs 2 and 3. The results of the proposed algorithm are comparable to Maxweight algorithm.

Fig. 12(c) presents the distribution for UEs of gNBs 4 and 5. The results of Local Maxweight algorithm are comparable to Maxweight.

X. Conclusions and Future Work

In the paper, we have provided new distributed and local scheduling algorithms for mmWave IAB networks, under practical constraints including half duplex constraints for each IAB node. We also include beamforming constraints that arise from the limited number of RF chains at each IAB node. A

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367



(a) UEs at gNB 1

(b) UEs at gNBs 2 & 3

(c) UEs at gNBs 4 & 5

Fig. 12. Cumulative distribution of end-to-end delays of UEs

fixed number of RF chains at a node puts a cap on the number of beams that can be simultaneously activated.

We have provided an efficient dynamic programming algorithm for computing the maximum weighted schedule, which can be implemented using message passing. The message passing scheme was used to implement an optimal global back-pressure policy for IAB networks in a distributed manner.

We have also proposed a local max-weight policy which only requires local information at gNBs for scheduling. We have shown that there is no gap when the link rates are unvarying. We have characterized a bound on the gap between the stability regions of the local policy and the global back-pressure policy, when links are time-varying. The gap is small when the standard-deviation of the link rate is small relative to the mean. Using numerical simulation, we have shown that the gap is indeed small in practical IAB scenarios. In simulation, the local max-weight algorithm has comparable delay performance to the global back-pressure algorithm.

In this paper, we have focused on the spanning tree topology described by 3GPP [2]. Distributed scheduling algorithms for other topologies such as forests and directed acyclic graphs can be considered for future work. Another interesting topic is to jointly consider scheduling and topology management under blocking.

Appendix A: Supporting Material

We use the notation $\Delta_t(\cdot)$ to mean the time difference of a process at $t+1$. For example, $\Delta_t(Q_n^l) := Q_n^l(t+1) - Q_n^l(t)$.

Consider P_n as the set of nodes in the path from n to r including n and r. We now derive telescoping equations of the aggregate queues flowing along this path. The results will be used in the proofs that follow. Let $A_r^\ell(t) := \sum_{f\in\mathcal{F}_\ell} a_r^f(t), \forall \ell \in \mathcal{L}$ be the total number of arrivals at the root r which will use link ℓ. Let $D_n^\ell(t) := \sum_{f\in\mathcal{F}_\ell} d_n^f(t), \forall \ell \in \mathcal{L}, n \in \mathcal{N}$ be the number of departures from node n in slot t, which will go over link ℓ. Also recall $Q_n^\ell(t) := \sum_{f\in\mathcal{F}_\ell} q_n^f(t), \forall \ell \in \mathcal{L}, n \in \mathcal{N}$ as the queue length in slot t at node n, of packets which will use link ℓ.

Consider a link $\ell \in \mathcal{L}_n$. For any $n' \in P_n - \{r\}$,

$$Q_{n'}^\ell(t+1) - Q_{n'}^\ell(t) = D_{p(n')}^\ell(t) - D_{n'}^\ell(t) \quad (36)$$

$$Q_r^\ell(t+1) - Q_r^\ell(t) = A_r^\ell(t) - D_r^\ell(t) \quad (37)$$

Summing over (37) and (36) for $n' \in P_n - \{r\}$ yields

$$V_\ell(t+1) - V_\ell(t) = A_r^\ell(t) - D_n^\ell(t) \quad (38)$$

where $V_\ell(t) := \sum_{n'\in P_n} Q_{n'}^\ell(t)$ is the aggregated queue length of ℓ along path P_n, for each node $n \in \mathcal{N}$ and $\ell \in \mathcal{L}_n$.

Lemma 2. *Suppose $D_n^l(t) = \min\{\mu_l(t)s_l(t), Q_n^l(t)\}$ for some $l \in \mathcal{L}_n$ under some scheduling policy. Then,*

$$Q_n^l(t)\left(\mu_l(t)s_l(t) - D_n^l(t)\right) \le \mu_{\max}^2 \quad (39)$$

for each $l \in \mathcal{L}_n$, $n \in \mathcal{N}$, $t \in \mathbb{Z}_+$.

Proof. Suppose $Q_n^l(t) > \mu_{\max}$, then $\mu_l(t)s_l(t) - D_n^l(t) = 0$. Hence, $Q_n^l(t)(\mu_l(t)s_l(t) - D_n^l(t)) = 0$. For the other case, suppose $Q_n^l(t) \le \mu_{\max}$. Note that $0 \le \mu_l(t)s_l(t) - D_n^l(t) \le \mu_{\max}$. The result is immediate □

A. Additional simulation parameters

The gNB i - gNB j distances (in m) for $(i,j) = (1,2),(1,3),(2,4),(3,5)$ are $396.5, 387.5, 369.3, 346.4$. The vector of gNB i to UE distances (in m) for $i = 1,\ldots,5$ are $[94.8, 89.0, 174.4, 82.2, 59.4, 165.5, 195.3, 12.6, 143.1, 180.3]$; $[156.5, 169.9, 108.0, 191.6, 169.0]$; $[147.3, 75.4, 122.2, 164.2, 132.9, 131.8, 157.2, 143.3, 161.3]$; $[155.0, 179.5, 144.4, 190.6, 113.0, 60.1, 109.7, 67.5, 182.1, 43.3]$ and $[158.3, 148.0, 181.0, 89.2, 185.1, 118.6, 173.7, 108.8]$ respectively.

Appendix B: Proof of Lemma 1

Proof. Note the following identity, for each node $m \in G$,

$$\sum_{l\in E_m} w_l s_l = \sum_{l\in\mathcal{L}_m} w_l s_l + \sum_{o\in C(m)} \sum_{l\in E_o} w_l s_l \quad (40)$$

since $E_m = \mathcal{L}_m \bigcup_{o\in C(m)} E_o$.

From (40), we obtain that $\sum_{l\in E_n} w_l s_l$ equals

$$\sum_{l\in\mathcal{L}_n} w_l s_l + \sum_{m\in R(\psi)} \sum_{l\in E_m} w_l s_l + \sum_{m\in C(n)-R(\psi)} \sum_{l\in E_m} w_l s_l \quad (41)$$

Consider any $m \in R(\psi)$. Since $s_{(n,m)} = 1$, from the half-duplex constraint, $s_l = 0$ for each $l \in \mathcal{L}_m$. Hence from (40), $\sum_{l\in E_m} w_l s_l = \sum_{o\in C(m)} \sum_{l\in E_o} w_l s_l$. Hence, from (7) and (41), we obtain

$$w(\psi) = \sum_{l\in\psi} w_l + \sum_{m\in R(\psi)} \sum_{o\in C(m)} \max_{[s_l]_{l\in E_o}} \sum_{l\in E_o} w_l s_l + \sum_{m\in C(n)-R(\psi)} \max_{[s_l]_{l\in E_m}} \sum_{l\in E_m} w_l s_l$$

where $[s_l]_{l\in E_o}$ is subject to the scheduling constraints on graph G_o for each $o \in C(m), m \in R(\psi)$, and $[s_l]_{l\in E_m}$ is subject to the scheduling constraints on graph G_m for each $m \in C(n) - R(\psi)$.

By the definition in (7)-(9), v_m equals the maximum value of $\sum_{l\in E_m} w_l s_l$ such that $[s_l]_{l\in E_m}$ is subject to the scheduling constraints on graph G_m. Hence, we obtain (13). □

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

APPENDIX C: PROOF OF THEOREM 4

Lemma 3. *If $\boldsymbol{\nu} \notin \Lambda_n$ for any $n \in \mathcal{N}$, then the system is unstable under any policy in $\mathcal{P}$.*

Proof. Firstly, note that under Assumption 2, the state process $\{\mathcal{Q}(t), \boldsymbol{\mu}(t)\}_{t=0}^{\infty}$ is a time homogeneous Markov chain.

Suppose that the system is stable under a stationary policy in $\mathcal{P}$ for some $[\nu_l]_{l\in\mathcal{L}_n} \notin \Lambda_n$. By definition, for a stable system $\limsup_{t\to\infty} \sum_{\tau=0}^{t-1} \mathbb{E}[\mathcal{Q}(\tau)]/t < \infty$. It follows that the Markov chain $\{\mathcal{Q}(t), \boldsymbol{\mu}(t)\}_{t=0}^{\infty}$ must be positive recurrent. Assume that the Markov chain $\{\mathcal{Q}(t)\}_{t=0}^{\infty}$ starts with the stationary distribution, i.e., the initial distribution is same as the stationary distribution of the Markov chain. We now provide a proof by contradiction.

Since the Markov chain is in stationary distribution and the system is stable, $\mathbb{E}[\mu_{b_n}(t)s_{b_n}(t)] \geq \nu_{b_n}$. Hence, $\mathbb{P}[s_{b_n}(t) = 1] \geq \nu_{b_n}/\bar{\mu}_{b_n}$ and

$$\mathbb{P}[s_{b_n}(t) = 0] \leq 1 - \nu_{b_n}/\bar{\mu}_{b_n} \tag{42}$$

Consider a link $l \in \mathcal{L}_n$, $\forall t \geq 0$. From (38) in Appendix A and (42), we obtain

$$\mathbb{E}[\Delta_t(V_l)] \geq \nu_l - (1 - \frac{\nu_{b_n}}{\bar{\mu}_{b_n}})\mathbb{E}[\mu_l(t)s_l(t)|s_{b_n}(t) = 0] \tag{43}$$

Note that $\mathbb{P}[\boldsymbol{\mu}_n(t)|s_{b_n}(t)] = \pi_{\boldsymbol{\mu}_n}$, for any policy in class $\mathcal{P}$. Hence, we obtain

$$[\mathbb{E}[\mu_l(t)s_l(t)|s_{b_n}(t) = 0]]_{l\in\mathcal{L}_n} = \sum_{\boldsymbol{\mu}_n\in\mathcal{M}_n} \pi_{\boldsymbol{\mu}_n} \sum_{\boldsymbol{s}_n\in\mathcal{S}_n} p_{\boldsymbol{s}_n|0,\boldsymbol{\mu}_n}\ \boldsymbol{\mu}_n \odot \boldsymbol{s}_n \tag{44}$$

where $p_{\boldsymbol{s}_n|0,\boldsymbol{\mu}_n} := \mathbb{P}[\boldsymbol{s}_n(t) = \boldsymbol{s}_n|s_{b_n}(t) = 0, \boldsymbol{\mu}_n(t) = \boldsymbol{\mu}_n]$. Hence from (44), it follows that

$$[\mathbb{E}[\mu_l(t)s_l(t)|s_{b_n}(t) = 0]]_{l\in\mathcal{L}_n} \in \sum_{\boldsymbol{\mu}_n\in\mathcal{M}_n} \pi_{\boldsymbol{\mu}_n} Conv(\mathcal{C}_{\boldsymbol{\mu}_n}) \tag{45}$$

Suppose $\frac{\nu_l}{1-\nu_{b_n}/\bar{\mu}_{b_n}} \leq \mathbb{E}[\mu_l(t)s_l(t)|s_{b_n}(t) = 0]$ for each $l \in \mathcal{L}_n$, then from (45), $\boldsymbol{\nu} \in \Lambda_n$, which is a contradiction since it is given that $\boldsymbol{\nu} \notin \Lambda_n$ for some n.

Hence, there must exist a $\ell \in \mathcal{L}_n$ such that $\nu_\ell > (1 - \nu_{b_n}/\bar{\mu}_{b_n})\mathbb{E}[\mu_\ell(t)s_\ell(t)|s_{b_n}(t) = 0]$. Then, $\mathbb{E}[V_\ell(t+1) - V_\ell(t)] > 0$ from (43). This is also a contradiction since the Markov chain is in stationary distribution, which completes the proof. □

Lemma 4. *If $\boldsymbol{\nu} + \delta\mathbf{1} \in \Lambda_{\mathcal{P}}$ for some $\delta > 0$, then the system is stable under a policy in $\mathcal{P}$.*

Proof. Since $\boldsymbol{\nu}^{(1)} := \boldsymbol{\nu} + \delta\mathbf{1} \in \Lambda_{\mathcal{P}}$, it follows that $\boldsymbol{\nu}^{(1)} \in \bigcap_n \Lambda_n$. We will use the vector $\boldsymbol{\nu}^{(1)}$ to construct a stationary randomized policy $\hat{\boldsymbol{s}}$ in the following. By definition of Λ_n,

$$\frac{[\nu_l^{(1)}]_{l\in\mathcal{L}_n}}{1 - \nu_{b_n}^{(1)}/\bar{\mu}_{b_n}} \in \sum_{\boldsymbol{\mu}_n\in\mathcal{M}_n} \pi_{\boldsymbol{\mu}_n} Conv(\mathcal{C}_{\boldsymbol{\mu}_n}) \tag{46}$$

Hence, there must exist $\{\boldsymbol{c}_{\boldsymbol{\mu}_n}\}_{\boldsymbol{\mu}_n\in\mathcal{M}_n}$ such that $\boldsymbol{c}_{\mu_n} \in Conv(\mathcal{C}_{\boldsymbol{\mu}_n})$, which satisfy

$$\frac{[\nu_l^{(1)}]_{l\in\mathcal{L}_n}}{1 - \nu_{b_n}^{(1)}/\bar{\mu}_{b_n}} = \sum_{\boldsymbol{\mu}_n\in\mathcal{M}_n} \pi_{\boldsymbol{\mu}_n} \boldsymbol{c}_{\boldsymbol{\mu}_n} \tag{47}$$

Since $\boldsymbol{c}_{\boldsymbol{\mu}_n} \in Conv(\mathcal{C}_{\boldsymbol{\mu}_n})$, it can be expressed as

$$\boldsymbol{c}_{\boldsymbol{\mu}_n} = \sum_{\boldsymbol{s}_n\in\mathcal{S}_n} p_{\boldsymbol{s}_n|\boldsymbol{\mu}_n}\ \boldsymbol{\mu}_n \odot \boldsymbol{s}_n \text{ such that } \sum_{\boldsymbol{s}_n\in\mathcal{S}_n} p_{\boldsymbol{s}_n|\boldsymbol{\mu}_n} = 1 \tag{48}$$

Now, consider a stationary randomized policy $\hat{\boldsymbol{s}}$ in $\mathcal{P}$ which makes decisions $\hat{\boldsymbol{s}}(t)$ based on $\boldsymbol{\mu}(t)$ defined as follows. The decision process starts at the root, the root r chooses $\hat{\boldsymbol{s}}_r(t) = \boldsymbol{s}_r \in \mathcal{S}_r$ w.p. $p_{\boldsymbol{s}_r|\boldsymbol{\mu}_r(t)}$ given the current link state $\boldsymbol{\mu}_r(t) \in \mathcal{M}_r$. For every other node n, the decision is made as follows

1) If $\hat{s}_{b_n}(t) = 1$ (i.e., parent node $p(n)$ has decided to schedule backhaul link b_n), then the links in $\mathcal{L}_n$ are not scheduled i.e., $\hat{\boldsymbol{s}}_n(t) = \mathbf{0}$.
2) If $\hat{s}_{b_n}(t) = 0$ (i.e., parent node $p(n)$ has decided to not schedule backhaul link b_n), then gNB n chooses $\hat{\boldsymbol{s}}_n(t) = \boldsymbol{s}_n \in \mathcal{S}_n$ w.p. $p_{\boldsymbol{s}_n|\boldsymbol{\mu}_n(t)}$ given the current link state $\boldsymbol{\mu}_n(t) \in \mathcal{M}_n$.

We also assume that a link l is not scheduled if $\mu_l(t) = 0$.

We will now show that $\mathbb{E}[\mu_l(t)\hat{s}_l(t)] = \nu_l^{(1)}$ for each $l \in \mathcal{L}$, using induction. Firstly, by construction of $\hat{\boldsymbol{s}}$ and from (47)-(48), $\mathbb{E}[\mu_l(t)\hat{s}_l(t)] = \nu_l^{(1)}$ for each $l \in \mathcal{L}_r$, ($\because \nu_{b_r}/\bar{\mu}_{b_r} = 0$).

Now suppose $\mathbb{E}[\mu_l(t)\hat{s}_l(t)] = \nu_l^{(1)}$ for each $l \in \mathcal{L}_n$ for some $n \in \mathcal{N}$. Consider a backhaul link b_m from gNB n to gNB m. Since $b_m \in \mathcal{L}_n$, $\mathbb{E}[\mu_{b_m}(t)\hat{s}_{b_m}(t)] = \nu_{b_m}^{(1)}$. Since $\mu_{b_m}(t) \in \{0, \bar{\mu}_{b_m}\}$, and since $\hat{s}_{b_m}(t) = 1$ only if $\mu_{b_m}(t) > 0$, we obtain

$$\mathbb{P}[\hat{s}_{b_m}(t) = 1] = \frac{\nu_{b_m}^{(1)}}{\bar{\mu}_{b_m}} \tag{49}$$

Hence, for each $l \in \mathcal{L}_m$,

$$\mathbb{E}[\mu_l(t)\hat{s}_l(t)] = \mathbb{P}[s_{b_m}(t) = 0]\mathbb{E}[\mu_l(t)\hat{s}_l(t)|\hat{s}_{b_m}(t) = 0] \tag{50}$$

$$= (1 - \frac{\nu_{b_m}^{(1)}}{\bar{\mu}_{b_m}})\mathbb{E}[\mu_l(t)\hat{s}_l(t)|\hat{s}_{b_m}(t) = 0] \tag{51}$$

Note that (52) holds, since $\mathbb{P}[\boldsymbol{\mu}_m(t)|\hat{s}_{b_m}(t)] = \pi_{\boldsymbol{\mu}_m}$. Here, $p_{\boldsymbol{s}_m|\boldsymbol{\mu}_m} := \mathbb{P}[\hat{\boldsymbol{s}}_m(t) = \boldsymbol{s}_m|s_{b_m}(t) = 0, \boldsymbol{\mu}_m(t) = \boldsymbol{\mu}_m]$.

$$[\mathbb{E}[\mu_l(t)\hat{s}_l(t)|\hat{s}_{b_m}(t) = 0]]_{l\in\mathcal{L}_m} = \sum_{\boldsymbol{\mu}_m\in\mathcal{M}_m} \pi_{\boldsymbol{\mu}_m} \sum_{\boldsymbol{s}_m\in\mathcal{S}_m} p_{\boldsymbol{s}_m|\boldsymbol{\mu}_m}\ \boldsymbol{\mu}_m \odot \boldsymbol{s}_m \tag{52}$$

Hence, from (51), (47) and (48), we obtain $\mathbb{E}[\mu_l(t)\hat{s}_l(t)] = \nu_l^{(1)}$ for each $l \in \mathcal{L}_m$, which completes the induction.

Since $\mathbb{E}[\mu_l(t)\hat{s}_l(t)] = \nu_l^{(1)} > \nu_l$ for each $l \in \mathcal{L}$, using standard Lyapunov drift arguments, such as in Theorem 6 of [28], it can be shown that the stationary randomized policy stabilizes the network. □

APPENDIX D: PROOF OF THEOREM 5

Proof of Theorem 5. Clearly $\Lambda \supseteq \Lambda_{\mathcal{P}}$, since local class $\mathcal{P}$ is a subset of all the stationary policies. Here, we will show that $\Lambda_{\mathcal{P}} \supseteq \Lambda$, which completes the proof.

Consider a $\boldsymbol{\nu} \in \Lambda$. Since it is given that $\boldsymbol{\mu}(t) = \boldsymbol{\mu}^d, \forall t$, it follows from the definition of Λ that $[\nu_l]_{l\in\mathcal{L}} \in Conv(\mathcal{C}_{\boldsymbol{\mu}^d})$. Hence,

$$[\nu_l]_{l\in\mathcal{L}} \in Conv([\boldsymbol{\mu}^d \odot \boldsymbol{s}]_{\boldsymbol{s}\in\mathcal{S}}) \tag{53}$$

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

where $\mathcal{S}$ is the set of all feasible schedules and $\odot$ is the component-wise product. Therefore, $\exists [p_{\boldsymbol{s}}]_{\boldsymbol{s}\in\mathcal{S}} \geq 0$ such that $[\nu_l]_{l\in\mathcal{L}} = \sum_{\boldsymbol{s}\in\mathcal{S}} p_{\boldsymbol{s}} \boldsymbol{\mu}^d \odot \boldsymbol{s}$ and $\sum_{\boldsymbol{s}\in\mathcal{S}} p_{\boldsymbol{s}} = 1$.

Consider some $n \in \mathcal{N}$. The set of feasible states $\mathcal{S}$ can be divided into two disjoint sets $A_1 := \{\boldsymbol{s} \in \mathcal{S} : s_{b_n} = 0\}$ and $A_2 := \{\boldsymbol{s} \in \mathcal{S} : s_{b_n} = 1\}$

Note that $\sum_{\boldsymbol{s}\in\mathcal{S}} p_{\boldsymbol{s}} \mu^d_{b_n} s_{b_n} = \nu_{b_n}$. It follows that $\sum_{\boldsymbol{s}\in A_2} p_{\boldsymbol{s}} \mu^d_{b_n} = \nu_{b_n}$. Therefore,

$$\sum_{\boldsymbol{s}\in A_2} p_{\boldsymbol{s}} = \nu_{b_n}/\mu^d_{b_n}; \quad \sum_{\boldsymbol{s}\in A_1} p_{\boldsymbol{s}} = 1 - \nu_{b_n}/\mu^d_{b_n} \tag{54}$$

Consider the links $l \in \mathcal{L}_n$, we have

$$[\nu_l]_{l\in\mathcal{L}_n} = \sum_{\boldsymbol{s}\in\mathcal{S}} p_{\boldsymbol{s}} [\mu^d_l]_{l\in\mathcal{L}_n} \odot [s_l]_{l\in\mathcal{L}_n} \tag{55}$$

Let $\alpha_n(\boldsymbol{s})$ be the local feasible schedule $[s_l]_{l\in\mathcal{L}_n}$ corresponding to the feasible schedule $\boldsymbol{s}$. We define for each local schedule $\boldsymbol{s}_n \in \mathcal{S}_n$,

$$p'_{\boldsymbol{s}_n} = \sum_{\boldsymbol{s}\in A_1 : \alpha_n(\boldsymbol{s}) = \boldsymbol{s}_n} p_{\boldsymbol{s}} \tag{56}$$

Note that $\alpha_n(\boldsymbol{s}) = [0]_{l\in\mathcal{L}_n}, \forall \boldsymbol{s} \in A_2$. Therefore, it follows from (55) and (56) that

$$[\nu_l]_{l\in\mathcal{L}_n} = \sum_{\boldsymbol{s}_n\in\mathcal{S}_n} p'_{\boldsymbol{s}_n} [\mu^d_l]_{l\in\mathcal{L}_n} \odot \boldsymbol{s}_n \tag{57}$$

By construction, $\sum_{\boldsymbol{s}_n\in\mathcal{S}_n} p'_{\boldsymbol{s}_n} = \sum_{\boldsymbol{s}\in A_1} p_{\boldsymbol{s}}$. Therefore, $\sum_{\boldsymbol{s}_n\in\mathcal{S}_n} p'_{\boldsymbol{s}_n} = 1 - \frac{\nu_{b_n}}{\mu^d_{b_n}}$ (from (54)). Diving (57) on both sides by $(1 - \nu_{b_n}/\mu^d_{b_n})$, we obtain

$$\frac{[\nu_l]_{l\in\mathcal{L}_n}}{(1-\nu_{b_n}/\mu^d_{b_n})} = \sum_{\boldsymbol{s}_n\in\mathcal{S}_n} \frac{p'_{\boldsymbol{s}_n}}{(1-\nu_{b_n}/\mu^d_{b_n})} [\mu^d_l]_{l\in\mathcal{L}_n} \odot \boldsymbol{s}_n \tag{58}$$

Since, $\sum_{\boldsymbol{s}_n\in\mathcal{S}_n} \frac{p'_{\boldsymbol{s}_n}}{(1-\nu_{b_n}/\mu^d_{b_n})} = 1$, it follows that

$$\frac{[\nu_l]_{l\in\mathcal{L}_n}}{(1-\nu_{b_n}/\mu^d_{b_n})} \in Conv(\boldsymbol{\mu}^d_n \odot \boldsymbol{s}_n) \tag{59}$$

where $\boldsymbol{\mu}^d_n = [\mu^d_l]_{l\in\mathcal{L}_n}$. Hence, $\boldsymbol{\nu} \in \Lambda_n$. Since the choice of n was arbitrary, it follows that $\boldsymbol{\nu} \in \Lambda_{\mathcal{P}}$. Hence, $\Lambda_{\mathcal{P}} \supseteq \Lambda$. □

Appendix E: Stability under the Local Max-weight algorithm

Lemma 5. *Given $\boldsymbol{\nu}$ is interior of $\Lambda_{\mathcal{P}}$, the queues at node r are stable under the proposed scheduling policy, i.e.,* $\limsup_{T\to\infty} \sum_{t=0}^{T} \sum_{f\in\mathcal{F}} \mathbb{E}[q^f_r(t)]/T < \infty$

Proof of Lemma 5. In this proof, $\boldsymbol{s}(t)$ refers to the the local max-weight policy, and $\hat{\boldsymbol{s}}(t)$ refers to the randomized policy of Lemma 4. Define $V^{loc}_r(t) := \sum_{l\in\mathcal{L}_r} (Q^l_r(t))^2$.

$$\mathbb{E}[\Delta_t(V^{loc}_r)|\mathcal{Q}(t)] = \sum_{l\in\mathcal{L}_r} \mathbb{E}[\big(\Delta_t(Q^l_r)\big)^2 + 2Q^l_r(t)\Delta_t(Q^l_r)|\mathcal{Q}(t)]$$
$$\leq K + 2\sum_{l\in\mathcal{L}_r} Q^l_r(t)\mathbb{E}[A^l_r(t) - D^l_r(t)|\mathcal{Q}(t)] \tag{60}$$

where $K = \sum_{l\in\mathcal{L}_r} \mathbb{E}[(A^l_r(t))^2] + |\mathcal{L}_r|\mu^2_{\max}$. (60) follows since $(\Delta_t(Q^l_r))^2 \leq (A^l_r(t))^2 + \mu^2_{\max}$ for each $l \in \mathcal{L}_r$.

Under the proposed local max-weight algorithm, $s_l(t) = 0, \forall l \in \mathcal{L}_r - \mathcal{L}'_r(t)$. Hence, from Lemma 2 and (60), we obtain

$$\mathbb{E}[\Delta_t(V^{loc}_r)|\mathcal{Q}(t)] \leq K_1 + 2\sum_{l\in\mathcal{L}_r} Q^l_r(t)\nu_l - 2\mathbb{E}[\sum_{l\in\mathcal{L}'_r(t)} Q^l_r(t)\mu_l(t)s_l(t)|\mathcal{Q}(t)] \tag{61}$$

where $K_1 := K + 2|\mathcal{L}_r|\mu^2_{\max}$. Since the proposed local max-weight algorithm maximizes the final term in (61), it must greater than $2\mathbb{E}[\sum_{l\in\mathcal{L}'_r(t)} Q^l_r(t)\mu_l(t)\hat{s}_l(t)|\mathcal{Q}(t)]$. Hence, from (61), and since $\hat{\boldsymbol{s}}(t)$ in Lemma 4 only makes decisions based on the channel state $\boldsymbol{\mu}(t)$, we obtain

$$\mathbb{E}[\Delta_t(V^{loc}_r)|\mathcal{Q}(t)] \leq K_1 + 2\sum_{l\in\mathcal{L}_r} Q^l_r(t)\nu_l - 2\mathbb{E}[\sum_{l\in\mathcal{L}'_r(t)} Q^l_r(t)\mu_l(t)\hat{s}_l(t)] \tag{62}$$

Note that $\forall l \in \mathcal{L}_r - \mathcal{L}'_r(t)$, either $\mu_l(t) = 0$ or $Q^l_r(t) \leq \mu_l(t)$. Therefore, $Q^l_r(t)\mu_l(t)\hat{s}_l(t) \leq \mu^2_{\max}$. Hence,

$$0 \leq 2(|\mathcal{L}_r|\mu^2_{\max} - \mathbb{E}[\sum_{l\in\mathcal{L}_r - \mathcal{L}'_r(t)} Q^l_r(t)\mu_l(t)\hat{s}_l(t)]) \tag{63}$$

Adding (63) to (62), we obtain

$$\mathbb{E}[\Delta_t(V^{loc}_r)|\mathcal{Q}(t)] \leq K_r + 2\sum_{l\in\mathcal{L}_r} Q^l_r(t)(\nu_l - \mathbb{E}[\mu_l(t)\hat{s}_l(t)])$$

where $K_r = K_1 + 2|\mathcal{L}_r|\mu^2_{\max}$. Since $\hat{\boldsymbol{s}}(t)$ is a stabilizing policy, $\exists \delta_r > 0$ such that $\mathbb{E}[\mu_l(t)\hat{s}_l(t)] \geq \nu_l + \delta_r, \forall l \in \mathcal{L}_r$. Hence,

$$\mathbb{E}[\Delta_t(V^{loc}_r)|\mathcal{Q}(t)] \leq K_r - 2\delta_r \sum_{l\in\mathcal{L}_r} Q^l_r(t) \tag{64}$$

Proceeding similarly as in the proof of Lemma 4.1 in [29], we obtain the result $\limsup_{T\to\infty} \sum_{t=0}^{T-1} \sum_{l\in\mathcal{L}_r} \mathbb{E}[Q^l_r(t)]/T \leq \frac{K_r}{2\delta_r}$. □

Lemma 6. *Given $\boldsymbol{\nu}$ is interior of $\Lambda_{\mathcal{P}}$, $\exists$ $\{W_l(t)\}_{l\in\mathcal{L}_n}$, non-negative functions of $\mathcal{Q}(t)$, and $K_n, \epsilon_n > 0$ such that*

$$\sum_{l\in\mathcal{L}_n} Q^l_n(t)\mathbb{E}[\Delta_t(W_l)|\mathcal{Q}(t)] \leq K_n - \epsilon_n \sum_{l\in\mathcal{L}_n} Q^l_n(t)$$

for each $n \in \mathcal{N} - \{r\}$, under the proposed scheduling policy.

Proof of Lemma 6. First, we introduce a local randomized policy $\hat{\boldsymbol{s}}^{loc}(t)$ and derive necessary properties. In the proof, $\boldsymbol{s}(t)$ refers to the proposed local max-weight policy.

Consider a local randomized policy which operates at n, and makes decisions $\hat{\boldsymbol{s}}^{loc}(t) := [\hat{s}^{loc}_\ell(t)]_{\ell\in\mathcal{L}_n}$, based on $\boldsymbol{\mu}_n(t) := [\mu_l(t)]_{l\in\mathcal{L}_n}$ and $s_{b_n}(t)$. We make use of the stabilizing policy $\hat{\boldsymbol{s}}(t)$ of Lemma 4 for the following construction. We define the policy as follows; $\forall l \in \mathcal{L}_n$

$$\mathbb{P}[\hat{s}^{loc}_l(t) = 1|\boldsymbol{\mu}_n(t), s_{b_n}(t) = 1] = 0 \tag{65}$$
$$\mathbb{P}[\hat{s}^{loc}_l(t) = 1|\boldsymbol{\mu}_n(t), s_{b_n}(t) = 0] = \mathbb{P}[\hat{s}_l(t) = 1|\boldsymbol{\mu}_n(t), \hat{s}_{b_n}(t) = 0] \tag{66}$$

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and the content may change prior to final publication. Citation information: DOI 10.1109/TNET.2022.3154367

Note that $s_{b_n}(t)$ is a function of $[\{q^f_{n'}(t)\}_{f\in\mathcal{F}}, \boldsymbol{\mu}_{n'}(t)]_{n'\in P_n-\{n\}}$. Hence from Assumption 2, $\boldsymbol{\mu}_n(t)$ is independent of $s_{b_n}(t)$. Now using (66), we obtain

$$\mathbb{E}[\mu_l(t)\hat{s}^{loc}_l(t)|s_{b_n}(t)=0] = \mathbb{E}[\mu_l(t)\hat{s}_l(t)|\hat{s}_{b_n}(t)=0] \quad (67)$$

Since the system is stable under the policy $\hat{\boldsymbol{s}}(t)$,

$$\mathbb{E}[\mu_l(t)\hat{s}_l(t)] > \nu_l \quad (68)$$

$$\mathbb{E}[\mu_l(t)\hat{s}_l(t)|\hat{s}_{b_n}(t)=0] > \nu_l/\mathbb{P}[\hat{s}_{b_n}(t)=0] \quad (69)$$

Since the queue at backhaul link b_n is stable under $\hat{\boldsymbol{s}}(t)$, we have $\mathbb{E}[\mu_{b_n}(t)s_{b_n}(t)] > \nu_{b_n}$, which implies $\mathbb{P}[\hat{s}_{b_n}(t)=1] > \nu_{b_n}/\bar{\mu}_{b_n}$, and $\mathbb{P}[\hat{s}_{b_n}(t)=0] < 1-\nu_{b_n}/\bar{\mu}_{b_n}$. Hence, it follows from (67) and (69), for each $l\in\mathcal{L}_n$, $\exists\delta>0$ such that

$$\mathbb{E}[\mu_l(t)\hat{s}^{loc}_l(t)|s_{b_n}(t)=0] = (1-\nu_{b_n}/\bar{\mu}_{b_n})^{-1}\nu_l + \delta \quad (70)$$

For $l\in\mathcal{L}_n$, define $W_l(t) := V_l(t) + \alpha_l V_{b_n}(t)$, where $\alpha_l := \delta/2\nu_{b_n} + \nu_l/(\bar{\mu}_{b_n}-\nu_{b_n})$. From (38) for $l\in\mathcal{L}_n$ and b_n,

$$\Delta_t(W_l) = A^l_r(t) - D^l_n(t) + \alpha_l(A^{b_n}_r(t) - D^{b_n}_{p(n)}(t)) \quad (71)$$

Note that whenever $s_{b_n}(t)=1$, we have $s_l(t)=0, \forall l\in\mathcal{L}_n$. Moreover, b_n is scheduled only when $\mu_{b_n}(t)=\bar{\mu}_{b_n}$ and $Q^l_n(t)\geq\bar{\mu}_{b_n}$. Hence from (71), for any $l\in\mathcal{L}_n$

$$\mathbb{E}[\Delta_t(W_l)|\mathcal{Q}(t), s_{b_n}(t)=1] = \nu_l + \alpha_l(\nu_{b_n}-\bar{\mu}_{b_n}) \quad (72)$$

$$= -(\bar{\mu}_{b_n}/\nu_{b_n}-1)\delta/2 =: -\delta' < 0 \quad (73)$$

Let E_1 denote the event $\{\mathcal{Q}(t), s_{b_n}(t)=1\}$. Hence,

$$\mathbb{E}[\sum_{l\in\mathcal{L}_n} Q^l_n(t)\Delta_t(W_l)|E_1] = -\delta'\sum_{l\in\mathcal{L}_n} Q^l_n(t) \quad (74)$$

For the other case $s_{b_n}(t)=0$, let E_0 denote the event $\{\mathcal{Q}(t), s_{b_n}(t)=0\}$. Consider

$$\mathbb{E}[\sum_{l\in\mathcal{L}_n} Q^l_n(t)\Delta_t(W_l)|E_0] \quad (75)$$

$$= \sum_{l\in\mathcal{L}_n} Q^l_n(t)(\nu_l+\alpha_l\nu_{b_n}) - \sum_{l\in\mathcal{L}'_n(t)} Q^l_n(t)\mathbb{E}[D^l_n(t)|E_0] \quad (76)$$

$$\leq \sum_{l\in\mathcal{L}_n} Q^l_n(t)(\nu_l+\alpha_l\nu_{b_n}) - \sum_{l\in\mathcal{L}'_n(t)} Q^l_n(t)\mathbb{E}[\mu_l(t)s_l(t)|E_0] + |\mathcal{L}_n|\mu^2_{\max} \quad (77)$$

(76) follows since $s_l(t)=0, l\in\mathcal{L}_n-\mathcal{L}'_n(t)$, and (77) is due to Lemma 2.

Since the proposed local max weight policy maximizes $\sum_{l\in\mathcal{L}'_n(t)} Q^l_n(t)\mu_l(t)s_l(t)$ whenever $s_{b_n}(t)=0$, we obtain $\mathbb{E}[\sum_{l\in\mathcal{L}_n} Q^l_n(t)\Delta_t(W_l)|E_0]$

$$\leq |\mathcal{L}_n|\mu^2_{\max} + \sum_{l\in\mathcal{L}_n} Q^l_n(t)(\nu_l+\alpha_l\nu_{b_n}) - \sum_{l\in\mathcal{L}'_n(t)} Q^l_n(t)\mathbb{E}[\mu_l(t)\hat{s}^{loc}_l(t)|s_{b_n}(t)=0] \quad (78)$$

$\forall l\in\mathcal{L}_n-\mathcal{L}'_n(t)$, either $\mu_l(t)=0$ or $Q^l_n(t)\leq\mu_l(t)$. This implies $Q^l_n(t)\mu_l(t)\hat{s}_l(t)\leq\mu^2_{\max}, \forall l\in\mathcal{L}_n-\mathcal{L}'_n(t)$. Therefore,

$$0 \leq |\mathcal{L}_n|\mu^2_{\max} - \mathbb{E}[\sum_{l\in\mathcal{L}_n-\mathcal{L}'_n(t)} Q^l_n(t)\mu_l(t)\hat{s}^{loc}_l(t)|s_{b_n}(t)=0] \quad (79)$$

Adding (79) to (78), we obtain $\mathbb{E}[\sum_{l\in\mathcal{L}_n} Q^l_n(t)\Delta_t(W_l)|E_0]$

$$\leq 2|\mathcal{L}_n|\mu^2_{\max} + \sum_{l\in\mathcal{L}_n} Q^l_n(t)(\nu_l+\alpha_l\nu_{b_n}) - \sum_{l\in\mathcal{L}_n} Q^l_n(t)\mathbb{E}[\mu_l(t)\hat{s}^{loc}_l(t)|s_{b_n}(t)=0] \quad (80)$$

$$= 2|\mathcal{L}_n|\mu^2_{\max} - \delta/2\sum_{l\in\mathcal{L}_n} Q^l_n(t) \quad (81)$$

(81) follows from (70), by substituting $\alpha_l := \delta/2\nu_{b_n} + \nu_l/(\bar{\mu}_{b_n}-\nu_{b_n})$.

Using law of total expectations, from (81) and (74), we obtain

$$\sum_{l\in\mathcal{L}_n} Q^l_n(t)\mathbb{E}[\Delta_t(W_l)|\mathcal{Q}(t)] \leq K_n - \epsilon_n\sum_{l\in\mathcal{L}_n} Q^l_n(t) \quad (82)$$

where $K_n = 2|\mathcal{L}_n|\mu^2_{\max}$ and $\epsilon_n = \min\{\delta/2, \delta'\}$ □

Proof of Theorem 7. We use proof by induction. Firstly, $\limsup_{t\to\infty}\sum^t_{\tau=0}\sum_{f\in\mathcal{F}}\mathbb{E}[q^f_m(\tau)]/t<\infty$ is true for $m=r$ from Lemma 5.

Now suppose $\limsup_{t\to\infty}\sum^t_{\tau=0}\sum_{f\in\mathcal{F}}\mathbb{E}[q^f_m(\tau)]/t<\infty$ for all the nodes m in the path from n to r excluding n, i.e., $P_n-\{n\}$. In the following, we will show the same is true for $m=n$ and complete the induction.

Consider the function $V^{loc}_n(t) := \sum_{l\in L_n} W^2_l(t)$, where $W_l(t)$ is defined in the proof of Lemma 6. We consider the conditional drift $\mathbb{E}[\Delta_t(V^{loc}_n)|\mathcal{Q}(t)]$, which equals

$$\sum_{l\in L_n}\mathbb{E}[(\Delta_t(W_l))^2 + 2W_l(t)\Delta_t(W_l)|\mathcal{Q}(t)] \leq K'_n + 2\sum_{l\in\mathcal{L}_n}\{W_l(t)-Q^l_n(t)\}\mathbb{E}[\Delta_t(W_l)|\mathcal{Q}(t)] + 2\sum_{l\in\mathcal{L}_n} Q^l_n(t)\mathbb{E}[\Delta_t(W_l)|\mathcal{Q}(t)] \quad (83)$$

where, $K'_n = \sum_{l\in\mathcal{L}_n}\mathbb{E}[(A^l_r(t)+\alpha_l A^{b_n}_r(t))^2] + (1+\alpha^2_l)\mu^2_{\max}$. (83) follows since $(\Delta_t(W_l))^2 \leq (A^l_r(t)+\alpha_l A^{b_n}_r(t))^2 + (1+\alpha^2_l)\mu^2_{\max}$.

Since $W_l(t+1)-W_l(t)\leq A^l_r(t)+\alpha_l A^{b_n}_r(t)$, the middle term in (83) is $\leq 2\sum_{l\in\mathcal{L}_n}\{W_l(t)-Q^l_n(t)\}(\nu_l+\alpha_l\nu_{b_n})$, which is a linear function of $\mathcal{Q}_{p(n)}(t) := \{q^f_m(t)\}_{f\in\mathcal{F}, m\in P_n-\{n\}}$. Let $g(\mathcal{Q}_{p(n)}(t)) := 2\sum_{l\in\mathcal{L}_n}\{W_l(t)-Q^l_n(t)\}(\nu_l+\alpha_l\nu_{b_n})$. The final term is $\leq 2K_n - 2\epsilon_n\sum_{l\in\mathcal{L}_n} Q^l_n(t)$ from Lemma 6. Hence,

$$\mathbb{E}[\Delta_t(V^{loc}_n)|\mathcal{Q}(t)] \leq K'_n + 2K_n + g(\mathcal{Q}_{p(n)}(t)) - 2\epsilon_n\sum_{l\in\mathcal{L}_n} Q^l_n(t) \quad (84)$$

Let $K''_n := K'_n + 2K_n$. Proceeding similarly as in the proof of Lemma 4.1 in [29], we obtain

$$\limsup_{T\to\infty}\frac{\sum^{T-1}_{t=0}\sum_{l\in\mathcal{L}_n}\mathbb{E}[Q^l_n(t)]}{T} \leq \frac{K''_n}{2\epsilon_n} + \limsup_{T\to\infty}\frac{\sum^{T-1}_{t=0}\mathbb{E}[g(\mathcal{Q}_{p(n)}(t))]}{2T\epsilon_n} \quad (85)$$

By supposition, the term on RHS of (85) is less than infinity. The induction is complete. □

REFERENCES

[1] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. Soong, and J. C. Zhang, "What will 5G be?" *IEEE Journal on selected areas in communications*, vol. 32, no. 6, pp. 1065–1082, 2014.

[2] 3GPP, "Study on integrated access and backhaul," 3rd Generation Partnership Project (3GPP), TR 38.874 (Rel 16), 2019.

[3] M. Polese, M. Giordani, T. Zugno, A. Roy, S. Goyal, D. Castor, and M. Zorzi, "Integrated access and backhaul in 5G mmWave networks: Potentials and challenges," *arXiv preprint arXiv:1906.01099v1*, 06 2019.

[4] R. Ford, F. Gómez-Cuba, M. Mezzavilla, and S. Rangan, "Dynamic time-domain duplexing for self-backhauled millimeter wave cellular networks," in *2015 IEEE International Conference on Communication Workshop (ICCW)*. IEEE, 2015, pp. 13–18.

[5] D. Yuan, H.-Y. Lin, J. Widmer, and M. Hollick, "Optimal joint routing and scheduling in millimeter-wave cellular networks," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 1205–1213.

[6] M. Eslami Rasekh, D. Guo, and U. Madhow, "Joint routing and resource allocation for millimeter wave picocellular backhaul," *IEEE Transactions on Wireless Communications*, vol. 19, no. 2, pp. 783–794, Feb 2020.

[7] Y. Li, J. Luo, R. Stirling-Gallacher, and G. Caire, "Integrated access and backhaul optimization for millimeter wave heterogeneous networks," *arXiv preprint arXiv:1901.04959v1*, 01 2019.

[8] Y. Niu, "Exploiting Multi-Hop relaying to overcome blockage in directional mmWave small cells," *Journal of Communications and Networks*, 04 2015.

[9] B. Sahoo, C.-H. Yao, and H.-Y. Wei, "Millimeter-wave multi-hop wireless backhauling for 5G cellular networks," in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*. IEEE, 06 2017, pp. 1–5.

[10] Y. Yao, H. Tian, G. Nie, H. Wu, and J. Jin, "Multi-path routing based QoS-aware fairness backhaul-access scheduling in mmWave UDN," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 09 2018, pp. 1–7.

[11] J. García-Rois, F. Gómez-Cuba, M. R. Akdeniz, F. J. González-Castaño, J. C. Burguillo, S. Rangan, and B. Lorenzo, "On the analysis of scheduling in dynamic duplex multihop mmWave cellular systems," *IEEE Transactions on Wireless Communications*, vol. 14, no. 11, pp. 6028–6042, 2015.

[12] F. Gomez-Cuba and M. Zorzi, "Optimal link scheduling in millimeter wave multi-hop networks with space division multiple access," in *2016 Information Theory and Applications Workshop (ITA)*. IEEE, 2016, pp. 1–9.

[13] J. Garcia-Rois, R. Banirazi, F. Gonzalez-Castano, B. Lorenzo, and J. Burguillo, "Delay-aware optimization framework for proportional flow delay differentiation in millimeter-wave backhaul cellular networks," *IEEE Transactions on Communications*, vol. PP, pp. 1–1, 01 2018.

[14] K. Vu, C.-F. Liu, M. Bennis, M. Debbah, and M. Latva-aho, "Path selection and rate allocation in self-backhauled mmWave networks," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 04 2018, pp. 1–6.

[15] M. Polese, M. Giordani, A. Roy, D. Castor, and M. Zorzi, "Distributed path selection strategies for integrated access and backhaul at mmWaves," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–7.

[16] Q. Hu and D. Blough, "Relay selection and scheduling for millimeter wave backhaul in urban environments," in *2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 10 2017, pp. 206–214.

[17] Y. Liu, Q. Hu, and D. M. Blough, "Joint link-level and network-level reconfiguration for MmWave backhaul survivability in urban environments," in *Proceedings of the 22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWIM '19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 143–151. [Online]. Available: https://doi.org/10.1145/3345768.3355913

[18] Y.-H. Chiang and W. Liao, "Mw-hierback: A cost-effective and robust millimeter wave hierarchical backhaul solution for hetnets," *IEEE Transactions on Mobile Computing*, vol. PP, pp. 1–1, 04 2017.

[19] G. Sharma, C. Joo, N. B. Shroff, and R. R. Mazumdar, "Joint congestion control and distributed scheduling for throughput guarantees in wireless networks," *ACM Trans. Model. Comput. Simul.*, vol. 21, no. 1, Dec. 2010. [Online]. Available: https://doi.org/10.1145/1870085.1870090

[20] A. Gupta, X. Lin, and R. Srikant, "Low-complexity distributed scheduling algorithms for wireless networks," *IEEE/ACM Transactions on Networking*, vol. 17, no. 6, pp. 1846–1859, 2009.

[21] C. Joo, X. Lin, J. Ryu, and N. B. Shroff, "Distributed greedy approximation to maximum weighted independent set for scheduling with fading channels," *IEEE/ACM Transactions on Networking*, vol. 24, no. 3, pp. 1476–1488, 2016.

[22] L. Jiang and J. Walrand, "A distributed csma algorithm for throughput and utility maximization in wireless networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 3, pp. 960–972, 2010.

[23] J. Ni, B. Tan, and R. Srikant, "Q-csma: Queue-length based csma/ca algorithms for achieving maximum throughput and low delay in wireless networks," in *2010 Proceedings IEEE INFOCOM*, 2010, pp. 1–5.

[24] L. X. Bui, S. Sanghavi, and R. Srikant, "Distributed link scheduling with constant overhead," *IEEE/ACM Transactions on Networking*, vol. 17, no. 5, pp. 1467–1480, 2009.

[25] E. Modiano, D. Shah, and G. Zussman, "Maximizing throughput in wireless networks via gossiping," in *Proceedings of the Joint International Conference on Measurement and Modeling of Computer Systems*, ser. SIGMETRICS '06/Performance '06. New York, NY, USA: Association for Computing Machinery, 2006, pp. 27–38. [Online]. Available: https://doi.org/10.1145/1140277.1140283

[26] G. Chen, M. Kuo, and J.-P. Sheu, "An optimal time algorithm for finding a maximum weight independent set in a tree," *BIT*, vol. 28, 06 1988.

[27] A. Eryilmaz, R. Srikant, and J. R. Perkins, "Stable scheduling policies for fading wireless channels," *IEEE/ACM Transactions on Networking*, vol. 13, no. 2, pp. 411–424, Apr. 2005. [Online]. Available: https://doi.org/10.1109/TNET.2004.842226

[28] M. J. Neely, "Dynamic power allocation and routing for satellite and wireless networks with time varying channels," Ph.D. dissertation, Massachusetts Institute of Technology, 2003.

[29] L. Georgiadis, M. J. Neely, and L. Tassiulas, *Resource allocation and cross-layer control in wireless networks*. Now Publishers Inc, 2006.

[30] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," in *29th IEEE Conference on Decision and Control*, 1990, pp. 2130–2132 vol.4.

[31] M. Gapeyenko, A. Samuylov, M. Gerasimenko, D. Moltchanov, S. Singh, M. R. Akdeniz, E. Aryafar, N. Himayat, S. Andreev, and Y. Koucheryavy, "On the temporal effects of mobile blockers in urban millimeter-wave cellular scenarios," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 11, pp. 10 124–10 138, 2017.

[32] M. K. Samimi, G. R. MacCartney, S. Sun, and T. S. Rappaport, "28 ghz millimeter-wave ultrawideband small-scale fading models in wireless channels," in *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, 2016, pp. 1–6.