Optimal Link Scheduling in Millimeter Wave Multi-hop Networks with Space Division Multiple Access

Felipe Gomez-Cuba
AtlantTIC, University of Vigo, Spain.
fgomez@gti.uvigo.es

Michele Zorzi
DEI, University of Padua, Italy.
zorzi@dei.unipd.it

Abstract—In this paper, we introduce a model for Multi-Input Multiple-Output (MIMO) Space Division Multiple Access (SDMA) into the analysis of a multi-hop millimeter wave network under the classic Network Utility Maximization (NUM) framework with Maximum Back Pressure scheduling (MBP). We show that the proof of convergence of MBP remains valid when we allow the scheduler to select multiple links to the same receiver in the same frame. Conventional MBP with a single link per receiver is traditionally implemented using the Maximum Weighted Matching (MWM) algorithm over the network graph. Under our modification, the problem becomes a Maximum Weighted Partition of the graph. Message Passing (MP) algorithms are efficient and have been successfully applied to graph partitioning problems in the past, so we use one to approximate the optimal MBP scheduling. Through simulation over a randomized mmWave picocell, we compare the MWM reference without SDMA, the efficient MP approximation, and the exact optimal MBP scheduler with SDMA (obtained by brute force). Simulations show that by leveraging SDMA in multi-hop mmWave network scheduling, a 50% capacity increase is obtained on average.

Index Terms—5G, Millimeter Wave, Beamforming, Space Division Multiple Access, Dynamic Duplexing, Scheduling, Network Utility Maximization

I. INTRODUCTION

Millimeter wave (mmWave) networks are an attractive candidate for beyond 4G and 5G cellular system evolution. The systems offer up to 200× increase in bandwidth along with further gains from highly directional antenna arrays. However, propagation of mmWave signals is affected by at least 20 dB higher free space loss than current cellular microwave systems. Even though this is partially compensated by the higher directivity achieved by large antenna arrays with small form factors, range is still limited to about 200 m [1], [2]. Thus, dense multi-hop architectures are necessary to extend the coverage of the network without recurring to cost-prohibitive hyper-dense wired backhauling deployments. Furthermore, multi-hop relaying is particularly attractive for mmWave systems because some cellular systems already use mmWave frequencies in dedicated backhaul links, and thus the addition of mmWave cellular links naturally gives rise to multi-hop mmWave networks.

Relaying has been introduced in the 3GPP Long Term Evolution (LTE) standard [3], [4], but it achieved limited benefits due to the use of a rigid frame structure for backward compatibility with the earlier, single-hop-only, version of the specification [5]. Steps to make frame structures more flexible have been taken in new versions of the standard [6]–[8]. Due to the fact that mmWave links are very directive, and therefore have fewer interference conflicts with other links in the network, this trend of increasingly flexible frame allocation will continue in mmWave 5G MAC, opening an opportunity for schedulers to optimize which links are active in each time instant or frame [9], [10].

Frequently, single-hop uplink cellular physical layer MIMO techniques achieve great gains from simultaneous reception of multiple transmitters by taking advantage of different users’ channel matrices. These different channels are algebraically treated as projections of the signal over orthogonal or partially-isolated subspaces of the vectorial space determined by the received signal at each antenna of the array. These simultaneous transmissions can take the form of spatial multiplexing, when all signals in different subspaces are desired by the receiver, or interference suppression, when some of the signals do not have to be received and the MIMO processing works to actively reduce the impact of their projection in the subspaces destined to desired signals. However, traditional literature on cellular systems frequently embraces the assumption that wireless communications in the network forms single-hop star topologies in each cell, with the base-station at the center and all users directly connected to it.

Conversely, if one looks at multi-hop networking literature, the body of work on optimal scheduling is dominated by ad hoc and sensor networks, that have been traditional niches for multi-hop architectures. These applications are characterized by low-complexity physical layers, making it extremely difficult to find results in multi-hop literature that are readily compatible with the future mmWave technology [11]. On the contrary, two assumptions about the physical layer that are commonplace in multi-hop scheduling papers are very far from what is expected in mmWave.

1) Power control for fixed rates: In sensor networks, the trade-off between battery and rate is usually resolved in favor of the former over the latter. Nodes adjust transmit...

F. Gomez-Cuba’s work is supported in part by FPU2012/01319. M. Zorzi’s work was supported in part by NYU-Wireless.
power to the minimum necessary to reach the receiver, making received power constant in all links. Classic results on scheduling analysis exploit this property by modeling networks with equal capacity for all links. With this assumption, calculating capacity is equivalent to counting the number of active links, and results on fundamental graph theoretic quantities have a direct correspondence to capacity measures. However, mmWave and cellular literature emphasizes high rates, whereas BSs usually have access to the power network. Received power and link capacities vary as power allocation achieves a balance between maximizing the cell throughput and offering fair rates to the users in the worst percentile.

2) **Destructive collision model:** Since the earliest ALOHA protocols, interference has been canonically represented in most multi-hop models as the impossibility to recover either signal when two or more transmitters are active within range of the receiver. More recent “physical layer aware” models allow the capture of the strongest colliding packet if its total interference power is under a threshold; but the paradigm still views collisions as fundamentally destructive events to be avoided by schedulers. In contrast, modern physical layer techniques can receive multiple packets simultaneously and cancel the interference from undesired signals. This requires new network schedulers that model collisions as beneficial up to some processing capability limit, and actively increase the number of simultaneous transmissions to maximize parallelism and spatial multiplexing gains.

In this paper we revisit one of the strongest frameworks in multi-hop network scheduling analysis, Network Utility Maximization (NUM) with Maximum Back Pressure (MBP) scheduling [10], [12], and add to the mmWave scheduling in [10] the necessary adaptions to accommodate multi-user MIMO receiver techniques, which are expected to play a critical role in the capacity of mmWave 5G cellular networks.

The rest of this paper is organized as follows. Section II describes our mmWave channel, link, network and traffic models. Section III describes the NUM problem and the scheduling algorithm. Section IV provides numeric examples of the solution to the problem and further discusses the properties of the model. Section V concludes the paper and discusses future work for developing practical algorithms and integrating the results in the cellular research framework.

### A. Notation

Matrices (vectors) are bold uppercase (lowercase) letters, $\mathbf{X}$ (x). We denote by $|\mathbf{X}|_k$ the $k$-th norm of a matrix $(\sum_{i,j} |X_{i,j}|^k)^{\frac{1}{k}}$, particularly the 1-norm represents the sum of the elements and the 2-norm is the Frobenius norm. $\mathbf{1}_{n,m}$ is the all-ones matrix with $n$ rows and $m$ columns.

### II. SYSTEM MODEL

#### A. mmWave SDMA Links

All mmWave nodes are assumed to have arrays of $N$ antennas that make use of analog processing techniques due to the high power consumption that would be required by having dedicated power amplifiers and analog/digital converters on each antenna to perform full-digital MIMO. We consider the hybrid analog/digital Space Division Multiple Access (SDMA) scheme represented in Fig 1. We assume all devices can transmit using a single power amplifier and transmission radio chain connected to an analog beamforming unit and antenna array, whereas all devices have a single receiver array connected to a hybrid architecture with up to $K \ll N$ parallel independent analog beamforming units, each connected to one analog/digital signal port. The receiver can process signals from $K$ transmitters simultaneously, by assigning each to one of the $K$ ports and adjusting the corresponding analog beamforming vector to the channel of the desired transmitter. Each of these $K$ connections forms the following equivalent point-to-point channel between the transmitter $n$ and the receiver $m$, which is simultaneously receiving from a set of transmitters $\mathcal{T}(m)$.

$$y_{n,m}(t) = w_{n,m}^r \mathbf{H}_{n,m} w_{n,m}^t g_{n,m} x_n(t)$$

$$+ \sum_{i \in \mathcal{T}(m) \setminus n} w_{i,m}^r \mathbf{H}_{i,m} w_{i,m}^t g_{i,m} x_i(t)$$

$$+ \sum_{i,j \notin n,m} w_{i,m}^r \mathbf{H}_{i,m} w_{i,j}^t g_{i,m} x_i(t) + z(t)$$

Here the first term is the desired transmission from $n$ to $m$. The second term represents the residual signal by other transmitters towards $m$ that leaks into the ADC port that $m$ has assigned to $n$. We consider these mismatched SDMA leaked signals as interference coming from the other transmitters $i \neq n, i \in \mathcal{T}(m)$ that also transmit to $m$ and meant to be received by its other ports. The third term represents any other interference by links not involving $m$ that are active in the network at the same time. The fourth term is
Additive White Gaussian Noise (AWGN) with variance $N_0$. We denote by $g_{n,m}$ the macroscopic pathloss, and by $H_{n,m}$ the normalized channel small scale fading matrix. Vector $w_{n,m}^t$ denotes the receiver beamforming at $m$’s port assigned to $n$; $w_{n,m}^t$ denotes the transmitter beamforming at $n$ when it is transmitting towards $m$. Equivalent definitions apply to the quantities with subindex $i$ or $j$, denoting other transmitters causing interference and their intended receivers.

We consider independent processing on each signal port of the receiver, and hence the receiver observes a series of $K$ scalar values $y_{n,m}$ for each transmitter $n \in \mathcal{T}(m)$. An evolution of this scheme could include the use of $K \times K$ multi-user digital MIMO interference suppression techniques to remove terms 2 and 3. However, we will defer advanced interference representation to later works and in this paper instead focus on the discussion of the generalization of backpressure algorithms with multiple links at the receiver. For this purpose, we will assume that nodes are sufficiently separated and antenna arrays are sufficiently large that miss-matched beamforming vectors have very small gain and the interfering signal is buried in noise. Put formally

$$\sum_{i \neq n} |w_{n,m}^t H_{i,m} w_{i,j}^t g_{i,j,m}|^2 \ll |z(t)|^2$$

(2)

$$y_{n,m}(t) \simeq w_{n,m}^t H_{n,m} w_{n,m}^t g_{n,m} x_{n,m}(t) + z(t)$$

The simplification of mmWave PHY assuming interference can be ignored because array gains are so directive is often called the “pseudo-wired” model [13]. In [10], this assumption is questioned: two models named Interference Free (IF) and Actual Interference (AI) are discussed, where the first follows pseudo-wired assumption whereas the latter takes the interference rigorously into account. The comparison showed that it can not be said that the pseudo-wired assumption is true for all links under all schedules: some links were shown to suffer non-negligible interference in some realizations of AI simulations in [10]. However, the result also showed that in the big picture, i.e., in the throughput optimality analysis, the scheduler tends to avoid the selection of links with interference. Thus, even though this pseudo-wired assumption is not true globally for all possible link combinations, it is satisfied locally for each set of links that are likely to be selected together by the optimal scheduler. In conclusion, the throughput optimality analyses using IF and AI in practice behave very similarly. In this paper we take advantage of this phenomenon and use the IF pseudo-wired approximation to focus on the discussion of scheduling and leave more accurate interference models for future work.

For the calculation of the beamforming vectors, we assume that channel matrices remain constant for the duration of a scheduling interval, and that transmitters design the beamforming vectors to maximize the Signal to Noise Ratio (SNR) in the absence of interference.

$$w_{n,m}^t, w_{n,m}^t = \arg \max |w^t H_{n,m} w|^2$$

(3)

Since the channel is essentially static and beamforming does not depend on interference, a node $n$ can obtain the set of neighbors connected to it ($\Omega(n)$), and compute all the necessary beamforming vectors during the attachment procedure ($w_{n,n}^t, w_{m,m}^t, m \in \Omega(n)$). An example of mmWave neighbor detection scheme is provided in [14].

For an accurate mmWave propagation, we compute the macroscopic pathloss

$$g_{n,m}(dB) = 75.85 + 37.3 \log_{10}(d(n,m)) + \log N_0(0, 8.36)$$

(4)

and the small scale fading matrix $H$ following the simulation algorithms described in [15, Sec. III-E]:

We compute the beamforming gain in the direction of the desired link $n, m$, denoted as

$$G_{n,m} = |w_{n,m}^t H_{n,m} w_{n,m}^t|^2$$

With these data, the capacity of link $n, m$ in a frame $t$ with duration $T_t$ is modeled in bits per frame as

$$c_{n,m}(t) = \alpha_1 T_t W \log \left(1 + \alpha_2 \frac{P_{n,m}(t) G_{n,m} g_{n,m}}{W N_0}\right)$$

(5)

where $W$ is the system bandwidth, $P_{n,m}(t)$ is the power allocated by $n$ to transmit towards $m$, $N_0$ is the noise power spectral density and the pathloss and beamforming gains are defined above. The two coefficients $\alpha_1, \alpha_2$ are power and bandwidth penalty factors introduced to fit to the Shannon capacity curve any specific practical physical layer of interest, and are often obtained from empirical data. For illustration purposes, in our simulations we set these values to $-3$ dB SNR penalty and no bandwidth penalty, i.e., $\alpha_1 = 1, \alpha_2 = 0.5$.

B. Network and Scheduling Model

We represent the wireless network by the directed graph $G(N, L)$, where $N$ is the set of nodes (Base Stations – BS –, Relay Nodes – RN – and User Equipment – UE –), $L$ is the set of links, and $F$ is the set of traffic flows in the network, indexed by $n, f$ and $g$ respectively. We denote the cardinalities of these sets as $N, L$ and $F$.

UEs can attach to as many RNs or BSs as they wish, we call the set of these two Access Points (APs), but no UE-UE connections are allowed. RNs, on the other hand, can communicate arbitrarily with any RN or BS. And BS are always connected to a wired backhaul which means that they do not need to connect wirelessly between them. Each node $n$ is aware of the set of neighbors connected to it $\Omega(n)$, and the maximum degree of the graph is $\Omega_{\text{max}}$. All devices have $K \geq \Omega_{\text{max}}$ receive radio chains and a single transmit power amplifier. However, radio stages are half-duplex in nature, and thus each device can either receive from (some subset of) all its neighbors $\Omega(n)$ at once, or transmit to only one of them. A very interesting extension that we will leave for future work is the case $1 < K \leq \Omega_{\text{max}}$, which would imply not all neighbors can be received at the same time. This could be modeled as a special case of link interference and easily assimilated in a generalization of our results to the AI link model.

For each node $n$ we define the boolean transmission indicator $s_n(t) = 1$ if node $n$ transmits at time $t$, and 0 otherwise.
Moreover for each pair of nodes that form a link \( \ell = (n,m) \), \( n,m \in \mathcal{N} \), we define the receiver selection binary variable \( p_{n,m}(t) \in \{0,1\} \) if \( n \) transmits towards \( m \). It is clear that to satisfy the half-duplex constraint \( p_{n,m}(t) = 0 \) if either \( s_n = 0 \) or \( s_m = 1 \). Moreover \( \sum_m p_{n,m}(t) = 1 \) to satisfy total transmit power constraints at \( n \). We constrain \( p_{n,m}(t) \) to be either 0 or 1 because in our SDMA model transmitters must select only one receiver to transmit to with full power; however, we have deliberately chosen this notation to remain applicable in future works when we expect to generalize our results to multiple links per transmitter, allowing \( p_{n,m}(t) \) to represent a real-valued fraction of power allocated by the transmitter to multiple receivers at once.

We represent the state of all nodes in frame \( t \) by the binary vector \( \mathbf{s}(t) \) and we denote the power allocations at all links by the binary vector \( \mathbf{p}(t) \). We call the pair \( (\mathbf{s}(t),\mathbf{p}(t)) \) a schedule on the network. Note that in our terminology a schedule \( (\mathbf{s}(t),\mathbf{p}(t)) \) is the allocation for one frame \( t \), and a scheduling policy is the method that chooses all schedules \( (\mathbf{s}(t),\mathbf{p}(t)) \forall t \). For each state vector \( \mathbf{s}(t) \), the set of all power allocations possible in this state \( \Pi(\mathbf{s}(t)) \) is continuous and convex and constrained by the half-duplex and power constraints. The set of all states is countable and contains all \( 2^N \) binary vectors of \( N \) elements. We denote the set of all possible schedules in the network by \( \Pi = \bigcup_{\mathbf{s}(t)} \Pi(\mathbf{s}(t)) \).

Notice that when \( p_{n,m}(t) = p_{m,n}(t) = 0 \forall m \in \Omega(n) \) the state of \( s_n(t) \) is irrelevant to network capacity, and in fact the only relevant ones in \( s(t) \) can be inferred by the nonzero elements in \( p(t) \). Thus the notation of the schedule with dual terms \( (\mathbf{s}(t),\mathbf{p}(t)) \) is redundant, but it is convenient to highlight that our solution separates the problem in two: the optimal power allocation towards receivers over \( \Pi(\mathbf{s}(t)) \) by pre-selected transmitters, and the selection of the optimal set of active transmitters determined by \( \mathbf{s}(t) \) alone.

Next, we define the traffic features in the network. As we said above, there are \( F \) flows. Each node \( n \) maintains a separate queue for each flow \( f \) and we denote the number of packets in it by \( q_n^f \). We denote by vectors \( \mathbf{q}_n, \mathbf{q}^f \) and \( \mathbf{q} \) the queue lengths of all flows at node \( n \), the queue lengths dedicated to flow \( f \) at all nodes, and all the queues of the network, respectively. For each flow \( f \in \mathcal{F} \), we denote by \( \mathcal{S}_f \) and \( \mathcal{D}_f \) the sets of sources and destinations of packets in \( f \). In each time frame \( t \), the sources produce \( a_f^f(t) \) packets of flow \( f \). When a packet of \( f \) reaches a destination, it is removed from the network. We use the following definition to characterize the average packet arrival rate.

**Definition 1.** An elastic packet arrival process by flow \( f \) in source node \( s \in \mathcal{S}_f \) is a stochastic process with a controllable time-varying mean arrival rate injected into the network \( \lambda_s^f(t) = \mathbb{E}[a_s^f(t)] \), with a long-term mean arrival rate \( \bar{x}_s^f = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \lambda_s^f(t) \).

We denote using vectors \( \mathbf{a}(t), \mathbf{\lambda}(t), \mathbf{x} \) the stacked packet arrival realizations, time-varying distribution mean, and long-term averages of the packet arrival process, respectively.

We recall that when \( (\mathbf{s}(t),\mathbf{p}(t)) \) is fixed, then the link capacities between any pair of nodes \( c_{n,m}(t) \) are determined by \( (5) \). Each transmitter node can only have one active link \( (n,m) \) and is allowed to use its rate to serve the queues of all flows. We denote the service rate at queue \( q_n^f(t) \) as \( c_{n,m}^f(t) \), satisfying \( \sum_{f \in \mathcal{F}} c_{n,m}^f(t) \leq c_{n,m}(t) \) and \( c_{n,m}^f(t) \leq q_n^f(t) \).

Finally, the evolution of each queue in the network follows

\[
q_n^f(t+1) = \begin{cases} \sum_m [c_{n,m}^f(t) - c_{n,m,n}^f(t)] + a_n^f(t) & n \notin \mathcal{D}_f \\ 0 & n \in \mathcal{D}_f \end{cases} \quad n \in \mathcal{D}_f \tag{6}
\]

We ignore flow destination queues (always zero), and compact the evolution of all system queues in vector notation by denoting all link capacities in a matrix with proper dimensions \( C(t) : \{c_{n+N(f-1),m+N(f-1)} = c_{n,m}^f(t)\} \)

\[
\mathbf{q}(t+1) = \mathbf{q}(t) + (C(t) - C^T(t))1_{\mathcal{N}F,1} + \mathbf{a}(t) \tag{7}
\]

### III. Throughput and NUM Optimality

#### A. Problem Statement

We formulate the scheduling problem as a Network Utility Maximization with constraints to guarantee network stability. For each flow, we define its utility function as a continuous non-decreasing function that attributes a utility value \( U_f(R^f) \) to the successful delivery of an average data rate \( R^f \) bits of flow \( f \) to its destinations.

We say a queue is stable if it does not grow unbounded, i.e., \( \lim_{t \to \infty} q_u^f(t) < \infty \), and the network is stable if all queues are stable \( \lim_{t \to \infty} |\mathbf{q}| < \infty \). We define the stability rate region of the network, \( \mathcal{X} \in \Lambda \), as the set of long-term average rate vectors for which there exists a scheduling policy such that the network is stable. When the network is stable, the long-term average rates of packets leaving the network equal the long term-average rates of exogenous traffic entering the network at the sources, \( R^f = \sum_{n \in \mathcal{S}(f)} x_n^f \), and the NUM problem takes the form

\[
\max_{x \in \Lambda} \sum_{f=1}^{F} U_f(\sum_{n=1}^{N} x_n^f) \tag{8}
\]

Note that \( \Lambda \) defines a notion of capacity region, because for any \( x \not\in \Lambda \) the network is unstable and by definition a positive fraction of the incoming rates \( x \) will stall in the queues for an infinite time, never reaching its destination. Therefore, the NUM problem is equivalent to estimating a point on the edge of the capacity region of the wireless network, for some defined function relating the rates of the flows. Particularly, using linear utility maximizes the sum rate, using diminishing returns laws such as \( U(r) = \frac{1}{2} \log(r) \) produces throughput-fairness results, and weighted functions can be employed to establish traffic priorities and QoS constraints.

#### B. Proposed Solution

We say a scheduling policy is throughput optimal if it makes the network stable for all \( x \in \Lambda \). From the definition, it follows
that the solution to the NUM problem can be achieved in a network decoupling the selection of the arrival rates, $x$, and the scheduling in the network, operated by a throughput optimal scheduler to guarantee stability independently of $x$.

**Proposition 1.** The Maximum Back Pressure Scheduling algorithm (Alg 1) is throughput optimal.

**Algorithm 1 MBP**

\[
\text{for all } t \text{ do } \\
(s(t), p(t)) = \arg \max_{(s(t), p(t))} \sum_{n=1}^{N} \sum_{m=1}^{N} \max_{f} c_{n,m}^{f}(q_{n}^{f} - q_{m}^{f}) \tag{9}
\]

\[
c_{n,m}^{f} = \begin{cases} 
\min(c_{n,m}, q_{n}^{f}) & f = \arg \max_{f} (q_{n}^{f} - q_{m}^{f}) \\
0 & \text{otherwise}
\end{cases}
\]

**Proposition 2.** In a network with MBP scheduling and rates controlled by the Adaptive NUM CC algorithm (Alg 2), long term rates converge to the solution of the approximate problem $x^{V}$ and this solution is arbitrarily close to $x^{*}$ as $V \to \infty$.

**Algorithm 2 Adaptive NUM CC**

\[
C_{\text{max}} = \max_{n,m} \{c_{n,m} | p_{n,m} = 1\}
\]

\[
\lambda_{n}^{f}(t) = \begin{cases} 
\max(\min(\tilde{u}^{-1}(\frac{q_{n}^{f}}{C_{\text{max}}}), 0), 0) & n \in S_{f} \\
0 & \text{otherwise}
\end{cases} \tag{11}
\]

C. Discussion of the Propositions

The idea behind the approximate problem is to introduce a penalty for queue length in the maximization objective, Alg 2 then reacts to queue lengths by reducing $\lambda_{n}^{f}(t)$ at the source nodes of each flow $f$ when the local queue grows, and increases the rate when the queue is small and the network is likely able to accept a higher rate. The inverse derivative of the utility function governs the direction of the response to queue length, leading the rates to climb the gradient increasing rates minus queue penalties.

The large scalar $V$ governs the strength of the reaction to changes in queue length. This means that larger values of $V$ to better approach the optimal rate come associated with allowing for much greater queue lengths and bringing the network much closer to unstability. In turn, being closer to unstability causes the MBP scheduling policy to need many more frame realizations to exhibit its long-term stochastic properties. In other words, the algorithm serves all queues eventually, but takes a much longer time to do so.

This is the main idea behind the concept of “throughput optimality”. The proof is an adaptation of the classic proof of optimality for MBP using a Lyapunov function that upper bounds queue lengths defined as $L(q) \triangleq |q|_{2} \geq |q|_{1}$. The proof of NUM near-optimality is a modification of the throughput-optimality proof with a Lyapunov function that measures queue length stability in relation to $x^{V}$, instead of raw inelastic traffics. Under the condition that there is a finite highest link capacity, $C_{\text{max}} \triangleq \max_{n,m} c_{n,m} < \infty$, it can be proved that the average Lyapunov drift along $T$ frames $\Delta(T) = E_{s}[L(q(t + T) - L(q(t)))]$ is negative for large values of $|q|_{1}$. This means that all the states of the system with sufficiently long queues have a stochastic tendency to drift to lower queue length states, and the network stabilizes.

The methods to derive the proofs for traditional ad hoc network models are outlined in Appendix A and well documented in the literature, so we will focus the discussion in the differences in the implementation of the algorithms for our SDMA mmWave model with multiple links per receiver.

D. Discussion of Implementation

The major difference with previous works that allowed only a single incoming link at each receiver is that we implement (9) by separating the selection of node roles as transmitters or receivers $(s(t))$, and the selection of a single destination as a form of power allocation at each node that has taken the role of transmitter. In maximizing the sum pressure weights associated with each link and flow $w_{n,m}^{f} = c_{n,m}^{f}(q_{n}^{f} - q_{m}^{f})$, we have that the single destination per transmitter constraint on $p(t)$ causes that if $c_{n,m}^{f} > 0$ then $\forall m' \neq m$, $c_{n,m'}^{f} = 0$. Separating the two problems, we note that for a fixed $s(t)$ and choosing $Q_{n,m} = \max_{f} (q_{n}^{f} - q_{m}^{f})$, then for any other flow $f'$ and power allocation $p'(t)$ that maximizes $w_{n,m}^{f'}$, the relation $w_{n,m}^{f'} < w_{n,m}^{f}$ is satisfied. Thus, we can always replace $f'$ by $\arg \max_{f} (q_{n}^{f} - q_{m}^{f})$ and increase the objective function. Conversely, if we fix $f$ and start with any other power allocation $p'(t)$, then by definition its weight is less than or equal to the weight contributed by the optimum $p^{*}(t)$. Thus, we have that the maximization is equivalent to

\[
\max_{s(t)} \max_{p(t) \in H(s(t))} \sum_{n=1}^{N} \sum_{m=1}^{N} c_{n,m}(t)Q_{n,m} \tag{12}
\]

\[
\sum_{n=1}^{N} \sum_{m=1}^{N} c_{n,m}(t)Q_{n,m} \tag{12}
\]
where the selection of the flows with the highest queue pressure can be separated from the capacity calculation. Due to the fact that we have considered a directive antenna model with negligible interference, the selection of the best destination is independent at each active transmitter for a fixed set of potential receivers (neighbors that are not transmitters themselves, given by the zeros in \( s(t) \)). Finally, the “outer” optimization over \( s \) is a graph partition problem where the weight of assigning each node to the set of transmitters or potential receivers is given by the optimal pressure weights calculated locally by each node. Message Passing (MP) is a framework that has been often used in graph partition problems with good results. Because the outer optimization is a graph partition, we implement a belief propagation approximation to the solution of (9). The convergence of a similar MP scheme is studied in [16]. However, our scheme does not satisfy the sufficient conditions for optimality of the analysis, and we have observed by simulation that the MP scheme sometimes converges to local maxima. In order to compare this MP implementation, which we consider as a heuristic approximation to the optimal, we have implemented an exhaustive brute-force search among the state-space of \( s \). For this, the fact that in our model we divide the graph partition and receiver selection problems has proven useful in that the size of the binary state space to be covered is reduced from \( 2^{N^2} \) to \( 2^N \).

IV. NUMERICAL EXAMPLES

We simulate the picocell network represented in Fig. 2, with 10 UEs randomly distributed in a disk of radius 200 m, and a BS at the center. Moreover, another four wireless RNs are placed at fixed locations at 115 m from the BS with a 90° separation. We define the minimum connectivity requirement as a maximum omnidirectional (i.e., without beamforming) pathloss of 164 dB. This threshold, inspired by [14], is selected for a reasonable rate of 10 Mbps when the BS transmits towards a UE, both obtain 30 dB beamforming gains at their arrays, and the radio hardware parameters are those on Table I.

Finally, we add two traffic flows for each UE: one uplink with source at the UE and destination at the BS, and one with source at the BS and destination at the UE. All exogenous arrivals apply the congestion control algorithm specified in (11). To select values of \( V \) (the congestion control tuning) we set \( V = 10 \lambda_{\text{max}}^2 \) as per discussion in Appendix A.

We develop a side-by-side comparison of three scheduling techniques in the same random network realization. In the first protocol, denoted MWM, the set of valid schedules \( \Pi \) only allows a single link per receiver and the general MBP in (9) is solved using the MWM algorithm, as in [10]. In contrast, in our proposed scheme \( \Pi \) allows multiple links per receiver using SDMA, while MBP scheduling is separated in two problems as in (12). The problem of selecting a destination receiver at each active transmitter, \( p(t) \in \Pi(s(t)) \), is a trivial power allocation that each transmitter may solve locally. We believe that this framework can be extended in future works to replace single destination selection with a complete power-allocation algorithm allowing multiple destinations at once by means of Spatial Division Multiplexing (SDM).

A. Comparison of Algorithms

In each frame the NUM method selects a set of links to be active. When the network stabilizes, the throughput optimal scheduler enters a steady state that repeats these patterns. We can discuss the properties of each scheduler by counting how often each different association pattern occurs. Figure 3 illustrates the stability of the three scheduling algorithms, showing how queue lengths and user throughput converge stochastically to stable values along a large number of frames.

Figure 4 represents the most used pattern by each algorithm and the histogram of all link activation patterns, ordered by number of appearances. Comparing the most frequent schedule in the reference MWM algorithm (Fig 4(a)) and either implementation of our proposal (Fig 4(b) or 4(c)), we can observe that the main difference resides in the fact that the traditional model in the literature only allows one link per receiver, whereas our model takes advantage of SDMA to enable multiple receptions at once, improving spatial multiplexing. The main goal of this paper is to characterize the advantages of transitioning to this new paradigm.

The analysis of the throughput-optimal schedulers can provide insights for the design of practical MAC and routing protocols that operate deterministically on much shorter frame durations. The scheduling length, or number of frames that a deterministic algorithm needs to serve all nodes, affects the delay in wireless communications. We can see in the histograms that the MWM algorithm used a total of 571 different frame patterns, with the most frequent not exceeding
2% of the time, and with 95% of the time represented by 265 different frames. This suggests that practical schedulers designed with a single-link-per-receiver constraint like MWM would have a very long scheduling length and delay.

On the other hand, both SDMA algorithms exhibit a few dominant patterns with a high probability, with the most used schedule appearing in approximately 20% of the frames and 95% of frames having only 92 and 232 different patterns, respectively. This suggests that deterministic practical schedulers designed to exploit SDMA to enable multiple links per receiver would have a much shorter scheduling length and delay.

**B. Average Result over Randomized Networks**

We generalize the observations above, specific to one network topology, by repeating the study over 50 randomly-generated node locations (“drops”) and averaging the results.

Due to the fact that the BF solution has exponential computation time, we perform the full comparison over 50 drops between the MWM and MP algorithms. To compare BF, we randomly select a sample of four random drops to assess the differences with MP and MWM.

In Figure 5 we can see the long-term sum utility function, long term sum throughput, and CDF achieved by the traffic flows in the network using MWM, MP and BF. The scatter plot in Fig 5(a) shows the utility values for each of the 50 random network drops, whereas the indicator to the left represents the average and standard deviation over all simulations. Due to the fact that the utility function is \(0.5 \log(r)\) and rates are in the order of Gbps, small utility changes may translate into large rate changes. The important result in Fig. 5(a) is not the raw utility gain, but rather the fact that its proportional-fair metric is very consistent in all algorithms. We represent the sum throughput for each protocol for each of the 50 random network drops in the bar plot in Fig 5(b). Finally, we represent the CDF of rates across all users and all simulations in Fig. 5(c). As we can see, the MP approximation algorithm delivers an average 50% increase in rate above the single-link-per-receiver MWM reference. To understand the imperfect approximation of the MP algorithm of the exact solution (9), we perform four representative drops of the BF algorithm, as its computational complexity prevents obtaining the solution in all 50 realizations. In the first drop, MP achieves 95% of the optimal BF solution, whereas MWM is below 50%. In the second and third drops, the difference is more moderate and the computationally-tractable protocols achieve 85% and 70% of the BF optimum, respectively. In the fourth drop, both protocols behave similarly at 75% of the BF gain. But it must be noted that this event only occurs sporadically in networks with very bad conditioning for MP, otherwise the gains are considerable. The MP heuristic is a step in the right direction,
but there is still interesting potential for future work to develop better techniques to solve (9).

V. CONCLUSIONS AND FUTURE WORK

Future mmWave 5G networks will require a combined framework drawing from existing models in both single-hop cellular and multi-hop network paradigms. In the past, one driver of cellular rate increase has been the spatial multiplexing gain derived from the reception of multiple wireless signals from different transmitters, using SDMA techniques. The physical layer assumptions contained in traditional multi-hop network scheduling incorporating the simultaneous processing of multiple signals at the same time. Our future plans include the implementation of power allocation algorithms enabling transmitters to select multiple destinations at once, reciprocating the introduction of multiple sources at each receiver in this paper. In addition, for fixed sets of transmitters and receivers the cellular literature contains many solutions to power allocation in the presence of interference. The next step after the implementation of a computationally tractable power allocation model under IF would be the extension of the PHY power allocation technique to incorporate interference avoidance with the Actual Interference link model.

REFERENCES


**APPENDIX A**

**PROOF OF NUM AND THROUGHPUT OPTIMALITY**

We define the Lyapunov function of the queue length as

\[ \mathcal{L}(q(t)) = \sum_{n,f} q_n^f(t)^2 = |q(t)|^2 = q^T(t)q(t) \geq \sum_{n,f} q_n^f(t) \]

We have the 1-step Lyapunov drift is

\[ \Delta_1(t) = E[\mathcal{L}(q(t+1)) - \mathcal{L}(q(t))] \]

We denote the queue vector \( q = q(t) \) and use the queue update function (7). This leaves

\[ \Delta_1(t) = E[|q + (C - C^T)\mathbf{1}_{NF,1} + a|^2 - |q|^2] \]

Expanding the square sum we get

\[ \Delta_1(t) = E\left[|q|^2 - |q|^2\right] + E\left[|C - C^T\mathbf{1}_{NF,1} + a|^2\right] \]

\[ \leq N^2 \lambda_{\max} + N \Omega \lambda_{\max} \]

\[ + 2E[q^T[(C - C^T)\mathbf{1}_{NF,1} + a]] \]

we add and subtract the term \( E[2V_1,\mathbf{1}_{NF,U}(a)] \), wrap the first terms in a constant \( C_1 \) and reorganize the last term

\[ \Delta_1(t) = C_1 + E[2V_1,\mathbf{1}_{NF,U}(a)] - E[2V_1,\mathbf{1}_{NF,U}(a)] - 2q^Ta] \]

\[ + 2E[q^T(C - C^T)\mathbf{1}_{NF,1}] \]

We introduce \( x^V \), the solution to the approximate problem (10), which by definition minimizes the third term

\[ \Delta_1(t) \leq C_1 + E[2V_1,\mathbf{1}_{NF,U}(a)] - E[2V_1,\mathbf{1}_{NF,U}(x^V)] - 2q^Tx^V \]

\[ + E[q^T(C - C^T)\mathbf{1}_{NF,1}] \]

we then arrange this as

\[ \Delta_1(t) \leq C_1 + E[2V_1,\mathbf{1}_{NF,U}(a)] - 2V_1,\mathbf{1}_{NF,U}(x^V)] + 2E[q^T(C - C^T)\mathbf{1}_{NF,1} - q^Tx^V] \]

From the assumption that \( x^V \in \Lambda \) we get that there exists some convex linear combination of feasible capacities \( C^V \in \mathcal{C}(\ell) \) such that the net traffic in the source is \( (C^V - (C^V)\mathbf{1}_{NF,1} - x^V = 0 \) for some small \( \epsilon \).

Finally, we note that \( \min q^T(C - C^T)\mathbf{1}_{NF,1} \) is a reordering of the terms in (9), therefore \( q^T(C - C^T)\mathbf{1}_{NF,1} < q^T(C - C^T)\mathbf{1}_{NF,1} \)

\[ \Delta_1(t) \leq C_1 + E[2V_1,\mathbf{1}_{NF,U}(a)] - 2V_1,\mathbf{1}_{NF,U}(x^V)] - E[q^T\mathbf{1}_{NF,1}] \]

The second term can be upper bounded by \( C_2 = 2V_N\Omega(\lambda_{\max}\Omega_{\max}) \). Which means that the Lyapunov drift can be written as a positive constant minus the sum queue length.

\[ \Delta_1(t) \leq C_1 + C_2 - \epsilon |q| \]

By the Foster-Lyapunov criteria, this means that when \( q \) is long enough, the drift becomes negative and system queue lengths decrease and the system is stabilized.

In addition, since this stabilization converges to the approximate optimum \( x^V \), we have that the distance to the true optimum is \( 1,\mathbf{1}_{NF,U}(x^*) - 1,\mathbf{1}_{NF,U}(x^V) \leq \frac{C_2}{\epsilon} \). This suggest that \( V \geq 10\lambda_{\max}^2 \) guarantees close approximations of \( x^* \).

**APPENDIX B**

**MESSAGE PASSING ALGORITHM**

We wish to find a solution to

\[ \max_s \sum_n c_n,m(p_n,m) \max_f q_n^f - q_m^f \]

satisfying \( p_n,m \in \{0,1\} \) and \( \sum_m p_n,m \leq 1 \forall n \), which can be rewritten as

\[ \max_s \sum_n \max_m \left( c_n,m(1) \max_f q_n^f - q_m^f \right) \]

This is similar to a partition of the graph in two sets (transmitters and receivers) maximizing the weight on the links that connect a node in the first set to a node in the second. We represent this as a graph with one node for each variable \( s_n \), and one node for the utility contributed by each potential transmitter, depending on its state and that of its neighbors \( f_n(s_n, s(\Omega(n))) = \max_m \in \{m \in \Omega(n)\} \sum_n c_n,m(1) \max_f (q_n^f - q_m^f) \)

This corresponds to the min-sum algorithm defined in [17].

1By contradiction, if there is no such linear combination, then the source queue grows to infinity and \( x^V \notin \Lambda \).