

Scalable Network Tomography for Dynamic Spectrum Access

Aadesh Madnaik, N. Cameron Matson, Karthikeyan Sundaresan
 School of Electrical and Computer Engineering, Georgia Institute of Technology, USA
 Email: {amadnaik3, ncmatson, karthik}@gatech.edu

Abstract—Mobile networks have increased spectral efficiency through advanced multiplexing strategies that are coordinated by base stations (BS) in licensed spectrum. However, external interference on clients leads to significant performance degradation during dynamic (unlicensed) spectrum access (DSA). We introduce the notion of network tomography for DSA, whereby clients are transformed into spectrum sensors, whose joint access statistics are measured and used to account for interfering sources. Albeit promising, performing such tomography naively incurs an impractical overhead that scales exponentially with the multiplexing order of the strategies deployed – which will only continue to grow with 5G/6G technologies.

To this end, we propose a novel, scalable network tomography framework called *NeTo-X* that estimates joint client access statistics with just linear overhead, and forms a blue-print of the interference, thus enabling efficient DSA for future networks. *NeTo-X*’s design incorporates intelligent algorithms that leverage multi-channel diversity and the spatial locality of interference impact on clients to accurately estimate the desired interference statistics from just pair-wise measurements of its clients. The merits of its framework are showcased in the context of resource management and jammer localization applications, where its performance significantly outperforms baseline approaches and closely approximates optimal performance at a scalable overhead.

Index Terms—unlicensed spectrum, hidden terminal interference, LTE-LAA scheduler design

I. INTRODUCTION

Need for DSA: 5G/6G mobile networks are moving to higher frequencies to access larger bandwidths in response to exponential traffic growth. Nevertheless, because of the limited operational range of these frequencies, lower spectral bands (sub-6 GHz) have remained a highly sought after resource. Indeed, the FCC has been exploring the possibility of re-farming some of these federal-owned bands for mobile access (e.g. CBRS band, 3.5-3.7 GHz [11]) through various flexible models of spectrum sharing. As spectrum sharing moves towards lightly-licensed and unlicensed models, dynamic spectrum access (DSA) continues to be an important problem in our search for better use of our critical spectral resources.

Multiplexing gains vs. open spectrum access: Mobile networks (LTE/5G) have evolved to accommodate several sophisticated technologies (e.g. carrier aggregation [29], multi-user MIMO [23]) that increase spectral efficiency by leveraging synchronous and licensed spectrum access. For mobile networks to be able to democratize access, the exorbitant cost of spectrum ownership needs to be tackled. While this can be addressed by moving towards unlicensed spectrum (e.g. LTE/5G in CBRS bands [7]), it sacrifices the gains these

This work was supported in part by NSF (CNS 2208761) and Georgia Tech Research Institute.

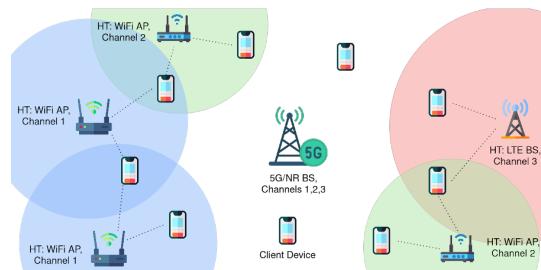


Fig. 1: Illustration of hidden terminal interference

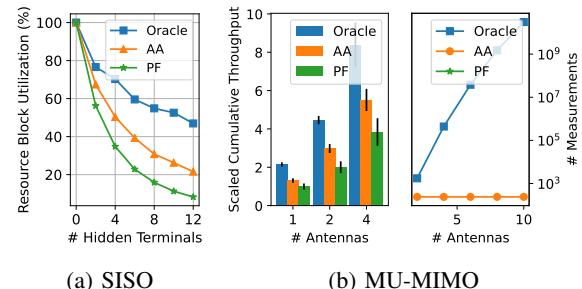


Fig. 2: HOD prevents drop in performance with (a) increased interference, (b) larger number of antennas

networks have worked hard to deliver. We conduct network simulations with a base station (BS), several tens of clients and a variable number of interfering sources, dubbed hidden terminals (HTs), to highlight this impact as illustrated in Fig. 1. The conventional proportional fair (PF) scheduler [21] is deployed alongside *mandated* unlicensed spectrum access protocol [4] at the BS. The results, from Fig. 2, reveal:

(a) *Interference can be modelled probabilistically*: Hidden terminal interference is commonly modelled as a Poisson Point Process [15], [17], [18]. As observed through our simulations, and verified by [31], this interference can be closely approximated by a Bernoulli Process with discretized time intervals.

(b) *Interference impact is significant*: As seen in Fig. 2(a), with an increase in the number of HTs, the ability of the BS to deliver or receive traffic from the affected clients diminishes, thus leading to a rapid drop (as high as 80%) in resource utilization. This reveals existing networks’ acute vulnerability to interference despite sophisticated access protocols.

(c) *Limited interference information is useful, but not sufficient*: In contrast to PF, which is interference-agnostic, we consider an alternate *access-aware* (AA) scheme. It incorporates some knowledge of the interference—namely the probability that individual clients can access the channel—in the scheduling decisions. The AA scheme performs 50-100% better than

PF, which is promising. However, as more antennas M are introduced in our 5G systems, the throughput depends on the specific set of users that are jointly scheduled on the same uplink resource. Hence, we also consider an Oracle that has access to all higher-order ‘joint’ access distributions (HODs) of all combinations of clients. Such an ideal scheduler benefits from interference-aware ‘group’ selection that is critical for MU-MIMO, and made possible by HODs. Fig. 2 reveals that while AA improves performance, there is still a wide gap to the theoretical performance that is possible with HODs.

Network Tomography for DSA: An important step in addressing such unknown interference is to first estimate it accurately. However, estimating interference is, in fact, a (receiver) *location*-dependent one. Thus, even a sophisticated spectrum scanning solution located at the BS cannot obtain a comprehensive view of the interference environment. This has led to the rise of network tomography for DSA [34], whereby the client devices’ existing communication interfaces double-up as virtual spectrum sensors. Through intelligent scheduling of the clients and measuring the corresponding outcomes, the BS can decipher the impact of interference on the clients, accounting for both spectrum and location dependence.

Challenge of Scalable Tomography: While HODs enables a network to operate in a smarter, more efficient way as seen in Fig. 2, the challenge lies in acquiring them in the first place. However, a naive approach would reveal only the first-order *marginal* probabilities. The HOD answer questions such as “What is the probability that clients i and j are interfered with but clients k and l are not?” We could continue to measure the HOD directly by scheduling each possible combination of clients, but in an M -antenna, C -channel MU-MIMO system with carrier aggregation [3], this results in an exponential measurement overhead $O(C \cdot 2^M)$, as seen in Fig. 2(b), which wastes spectral resources.

In this work, we aim to address a fundamental challenge: *Can we continue to realize mobile network’s multiplexing features, while maintaining performance comparable to licensed spectrum access, even in unlicensed spectrum with unknown interference with low overheads?*

Neto-X: To this end, we propose a novel, scalable network tomography framework called *Neto-X* that estimates HODs more accurately, at just a linear overhead, and generates a blue-print of the interferers, to enable efficient DSA for 5G/6G networks. *Neto-X* helps address the scalability challenge in accurately estimating HODs across multiple channels, antennas, and a large number of clients. The HOD serves as a key building block for multiple applications like (i) resource management—enabling mobile networks to retain the benefits of their multiplexing techniques, and (ii) security—allowing defense and private networks to localize potential interfering/jamming sources.

Design: *Neto-X*’s design incorporates three key elements:

(a) *Interference-Aware Clustering:* *Neto-X* leverages the spatial consistency and channel diversity of the interference experienced by clients to cluster clients and choose representatives. A single representative client from a cluster is sufficient

for measurements, which helps to significantly reduce the measurement overhead.

(b) *Generating HODs from Pairwise Measurements:* Leveraging recent results in latent variable decomposition, *Neto-X* proposes an algorithm to estimate the HODs of arbitrary groups of clients, from just pairwise measurements of the clusters’ representatives.

(c) *Blue-Printing Interference:* *Neto-X* proposes a novel algorithm that accurately estimates the number of interfering sources and their corresponding dependencies on clients based on the latent variable decomposition of the HODs.

Applications: *Neto-X* showcases two key applications.

(a) *Resource Management:* The HODs are incorporated into mobile network’s scheduling framework to enable interference-aware multi-user access. *Neto-X* is implemented in NS3 [1] and supplemented with numerical evaluations to understand *Neto-X*’s ability to deliver gains in multiplexing (higher order MU-MIMO and carrier aggregation) schemes despite the presence of external interference. Our evaluations highlight that *Neto-X* helps deliver $> 2x$ throughput gain over existing schedulers in high interference regimes for SISO, while maintaining a gain of 2.5x even for low-moderate interference regimes for MU-MIMO systems.

(b) *Interferer/Jammer Localization:* Leveraging the blue-printed interference, *Neto-X* is able to identify a small set of key clients, whose locations in turn allow us to maximally narrow down the location of the interfering sources to a relatively small area of under 200 m^2 . Further, *Neto-X* localizes hidden terminals to a 75-th percentile error of under 10m - a feature not possible by existing DSA schemes.

Our contributions in this work are as follows.

- We propose a scalable network tomography framework *Neto-X* that enables efficient multi-user, multi-channel access for 5G/6G systems in unlicensed spectrum at low overhead.
- *Neto-X* designs a novel, interference-aware channel allocation and resource management mechanism that helps deliver the benefits of carrier aggregation and higher-order MU-MIMO even in the presence of interference.
- *Neto-X* proposes a novel mechanism that further leverages the hidden dependencies between interfering sources and clients, to localize the interfering sources from a carefully identified, small set of clients and their positions.

II. BACKGROUND AND RELATED WORK

5G Access Overview: Cellular networks (Long Term Evolution, LTE in 4G and upcoming New Radio, NR in 5G) are synchronous, scheduled access systems designed for operation in the licensed spectrum. The base station (BS) is responsible for scheduling both the downlink (DL) and uplink (UL) clients in its sub-frames, which consists of two-dimensional resource elements spanning both time (symbols) and frequency (sub-carriers), called resource blocks (RBs). LTE/5G-NR employs OFDMA (orthogonal frequency division multiple access), whereby multiple clients are scheduled in each sub-frame on *different* RBs - in the case of multi-user and massive MIMO that are central to 5G systems, multiple clients are scheduled on the *same* RB, but spatially separated by multiple antennas.

The schedule for both DL and UL transmissions is conveyed to the clients through the control part of the DL sub-frames.

5G in Un-licensed Spectrum: Unlike traditional cellular systems that operate in an always-on mode in licensed spectrum, operating in unlicensed spectrum *requires* 5G to adopt asynchronous access principles of clear-channel assessment (CCA) through energy sensing, called Listen-Before-Talk [4] and back-off for co-existence with other technologies (e.g. WiFi or other cellular networks). The prevalent solution today is license-assisted access (LTE-U [22] and LTE-LAA/eLAA [4], [5], [26]), along with the numerous works that have studied the LTE-WiFi coexistence problem [12], [16], [24], [25], [30], [32], [36], where unlicensed carriers are aggregated with existing licensed carriers. By relying on licensed carriers as anchors, the impact of asynchronous interference is confined to unlicensed carriers, thus limiting the overall impact. Similarly, the use of spectrum access servers in CBRS [33] enables coarse timescale coordination for exclusive use of the spectrum. In contrast, operating 5G entirely in unlicensed spectrum (e.g. MulteFire [28] in 3.5 GHz CBRS bands in GAA mode) that is shared across multiple entities, exposes the vulnerability of its capacity-enhancing multi-user schemes to asynchronous interference (e.g. WiFi APs interfering with clients but not sensed by the BS), highlighting its lack of readiness for practical deployments [34]. While [34] has showed promise for how 4G LTE systems can cope with such asynchronous interference, it left open several fundamental algorithmic and system level challenges that must be solved for a practical 5G solution, especially scaling to larger multiplexing gains (from larger bandwidths and antenna arrays in 5G) during real-time operations. This, in turn, forms one of the objectives of this work.

Network Tomography for DSA: Spectrum sensing has understandably become a key component of DSA research [2], [6], [12], [16], [26], but estimating interference is not just a spectrum-dependent process. Meanwhile, network tomography has been popular in wired networks [8]–[10], [35] where it refers to leveraging measurements between end-hosts of a network so as to reveal important properties of the network itself. [34] introduced a similar notion for DSA, whereby they transformed the clients of our cellular network into spectrum sensors, whose measurements in accessing the spectrum are leveraged to reveal interesting properties on the interference they face. This is made possible by OFDMA, where a BS can orchestrate measurements by scheduling clients either in isolation or simultaneously with others and estimating the resulting access distributions.

III. SYSTEM DESIGN

A. Overview

As stated in Section I, having complete information about the joint-access distribution (HODs) is beneficial for efficient resource management as well as localizing interfering sources. To this end, the goal of *NeTo-X* is to accurately estimate HODs with as few measurements as possible and without interrupting the normal operation of the network. *NeTo-X* does this by treating the clients as a distributed set of sensors

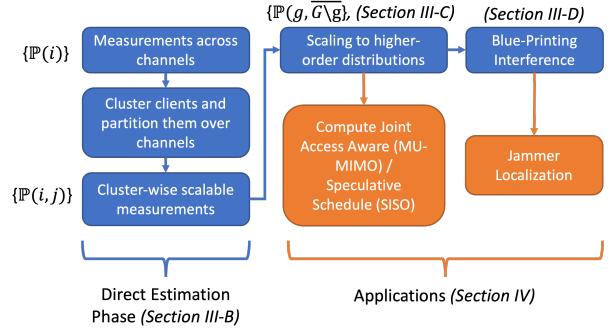


Fig. 3: System Design and Information Flow

to learn information about the interference environment and intelligently scale those measurements to estimate the HODs.

The system consists of several steps summarized in Fig. 3. *NeTo-X*'s operation is divided into two phases: a *direct measurement* phase and a *scaling* phase. The measurement phase consists of three steps: measuring first-order marginals per client, channel-wise clustering of clients, and measuring second-order marginals per cluster. First, the first-order marginals are measured during clients' ongoing data transmissions by observing the interference they experience on each channel (Section III-B1). Next, to reduce the amount of measurements required in estimating second-order marginals, *NeTo-X* uses a clustering algorithm to group clients with similar interference patterns such that interference dependencies can be captured at a cluster- (instead of client-) level without losing information (Section III-B2). Finally, second-order marginals are measured by scheduling pairs of clients from unique clusters (Section III-B3).

In the second phase (Section III-C), we use latent variable decomposition to scale these estimates up to the HOD. Once the HODs have been estimated, we can use these for client scheduling as discussed in Section I. An additional step (Section III-D), which we call interference blue-printing, uses the HOD to uniquely identify the sources of interference. This information, along with knowledge of a few client locations, can be used to localize the interferers themselves.

Notation: Vectors, matrices and tensors are represented as \mathbf{a} , \mathbf{A} , \mathbf{X} , respectively. The i -th element of a vector is given by $\mathbf{a}(i)$. Similarly, f -th column and i -th row of a matrix \mathbf{A} are notated $\mathbf{A}(:, f)$ and $\mathbf{A}(i, :)$, respectively.

B. Direct Estimation Phase

1) *Channel Sampling:* The first phase of *NeTo-X* consists of estimating the channel access probability for each client individually across all of the channels. The probability that a client $i \in [1, \dots, N]$ is able to access channel $c \in [1, \dots, C]$ is described by a Bernouli random variable with parameter a_i^c . We estimate these individual channel access probabilities empirically by counting the number of times a scheduled client is able to access the channel.

$$\mathbb{P}(\text{client } i \text{ can access channel } c) = \frac{\# \text{ times } i \text{ accesses } c}{\# \text{ times } i \text{ scheduled on } c} = a_i^c$$

By stacking each of the channel access probabilities for a given client, we create a channel access vector $\mathbf{a}_i = [a_i^1, a_i^2, \dots, a_i^C]$.

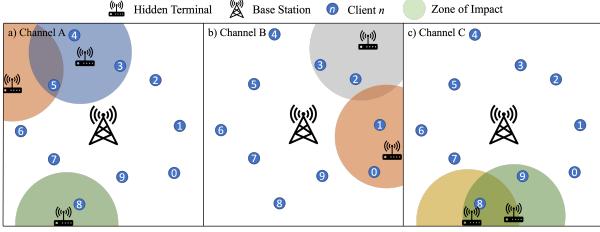


Fig. 4: Impact of hidden terminals across three channels.

This process requires scheduling each of the N clients across each of the C channels to take measurements, i.e. total overhead of $O(CN)$. Measurements are collected through uplink grants, and the clients send uplink packets as part of the normal operation of the network.

2) *Client Clustering*: The direct phase continues by measuring pair-wise access probabilities. Doing so naively would require us to schedule every pair of clients (jointly) on each of the channels, resulting in an overhead of $O(C \cdot \binom{N}{2})$ measurements. To reduce this number, we leverage the following observation: if two clients are physically located close enough such that they are affected by the same set of HTs, they will likely have similar first-degree marginal channel access probabilities over all of the channels. To illustrate this, consider Fig. 4. In this topology there are ten clients, seven HTs, and three channels. For example, in channel A , client 5 is impacted independently by two HTs. However, in channel C , the two HTs are close enough such that their transmissions can be sensed, through some form of carrier sensing like CSMA, by one another. So, when the yellow HT is transmitting, the green HT cannot use the channel (which frees client 9 to transmit). In this scenario, client 8 is impacted by a *dependent* set of HTs. Now, consider clients 0 and 1. After the first phase of measurements, we would observe that they have similar channel access probabilities on all channels—no interference on channels A and C , and same probability of interference on channel B , i.e. $\mathbf{a}_0 \approx \mathbf{a}_1$.

We group the clients into clusters via K-means using the squared L_2 norm of the difference between their channel access vectors, \mathbf{a} , to measure the distance. Note that clustering based on identifiers like RSSI has previously been used to spatially locate clients [27]. After clustering it is sufficient to perform pair-wise measurements with a single representative client from each cluster. Intuitively, this is because the clusters group clients that experience similar interference; therefore, the joint distributions of the representative clients across the clusters will be the same as those between the remaining, non-representative clients from those same clusters. The clustering phase reduces the size of measurement set from N clients to K clusters which does not depend directly on N . We note that K is a design parameter of *NeTo-X*. An approximation is sufficient, since K should simply capture number of the contention (interference) regions in the network. We investigate the impact of incorrect clustering as well as the choice of K on the accuracy of the eventual HOD and the measurement overhead reduction in Section V-A5.

3) *Pairwise Probability Estimation*: The final step of the direct estimation phase is to measure the pair-wise channel access which can be computed empirically as before with the single client access. The BS simultaneously schedules a representative client from distinct clusters i and j and counts the number of opportunities one, both, or neither are able to access the channel. This incurs $O(C \binom{K}{2})$ measurements, after which we have an estimate of all (cluster) pair-wise channel access probabilities, $\mathbb{P}(i, j)$.

C. Scaling to High Order Distributions

Clustering helps to keep the measurement set manageable for estimating the pair-wise densities, but the number of measurements required to estimate HODs directly—even between clusters—quickly becomes unmanageable. In order to scale the pair-wise marginals up to higher order marginals of arbitrary size (including full joint distribution) we leverage latent variable decomposition techniques from [19], [20].

1) *Joint Probability Representation*: Consider a set of discrete, finite-alphabet random variables Z_1, Z_2, \dots, Z_N . We use $\mathbb{P}(z_1, z_2, \dots, z_N)$ as a shorthand notation to represent $\mathbb{P}(Z_1 = z_1, Z_2 = z_2, \dots, Z_N = z_N)$. Consider the scenario in which we do not have access to the joint probability distribution for $\{Z_n\}_{n=1}^N$. Rather, we only have access to one- and two-degree marginals, $\mathbb{P}(z_i, z_j)$ over all pairs $\{i, j\}$. This leads to an important question:

Can we recover the joint probability distribution given we have access to solely low-dimensional marginals without any structural assumptions on the random variables?

We can use the fact that any joint probability mass function (PMF) admits a Naive Bayes' model representation. This implies that any joint PMF can be generated from a latent variable model with just one hidden variable (H) given a sufficiently rich alphabet (F , [19] Theorem 1). Thus, the joint PMF of $\{Z_n\}_{n=1}^N$ can always be decomposed as,

$$\mathbb{P}(z_1, \dots, z_N) = \sum_{f=1}^F \mathbb{P}(H = f) \prod_{n=1}^N \mathbb{P}(Z_n = z_n | H = f). \quad (1)$$

We exploit this property to express degree-2 marginals as

$$\mathbb{P}_{ab}(z_a, z_b) = \sum_{f=1}^F \lambda(f) \mathbb{P}(Z_a = z_a | f) \mathbb{P}(Z_b = z_b | f). \quad (2)$$

The correctness of Eq. 1 is related to canonical polyadic decomposition (CPD). Given an N -way tensor $\mathbf{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$ with CP rank F , it admits a decomposition of the form:

$$\mathbf{X} = \sum_{f=1}^F \lambda(f) \mathbf{A}_1(:, f) \otimes \mathbf{A}_2(:, f) \otimes \dots \otimes \mathbf{A}_N(:, f) \quad (3)$$

where \otimes is the tensor outer product operator, $\mathbf{A}_n \in \mathbb{R}^{I_n \times F}$ is called the mode- n latent factor, and $\|\lambda\|_0 = F$ is used to normalize the columns of \mathbf{A}_n . Mapping Eq. 3 onto 1, as explored in [19], [20], indicates that the random variables Z_1, \dots, Z_N are conditionally independent given a latent variable H .

2) *From Pairwise to HOD*: In the case of channel access probability, each of the random variables is binary ($z_n \in \{0, 1\}, \forall n$), so it suffices to write the degree-2 marginal for a pair of clients i and j as $\mathbb{P}_{ij}(\cdot, \cdot)$. Using Eq. 2 we represent the joint access probability of clients i and j using the conditionals

on the latent variable H . As shorthand notation, we use p_i^f for $\mathbb{P}(Z_i = 1 | H = f)$. There are $N \times F$ independent parameters for a given set of clients and latent variable H . We stack them to create the matrix \mathbf{P} . The probability that clients i and j will be able to access the channel jointly can now be written as

$$\mathbb{P}_{ij}(1, 1) = \sum_{f=1}^F \lambda(f) p_i^f p_j^f. \quad (4)$$

Similarly, the probability that client i can access the channel, while j cannot is

$$\mathbb{P}_{ij}(1, 0) = \sum_{f=1}^F \lambda(f) p_i^f (1 - p_j^f). \quad (5)$$

We observe that the probability measure \mathbb{P}_{ij} is parameterized by p_i^f, p_j^f , and the vector λ . Next, we minimize the KL-divergence between the measured pair-wise marginals $\hat{\mathbb{P}}(i, j)$ and the model $\mathbb{P}(i, j)$ for all pairs $\{i, j\}$. The loss function is

$$\mathcal{L} = \sum_{\text{all pairs } \{i, j\}} \mathbb{D}_{KL}(\hat{\mathbb{P}}_{ij}, \mathbb{P}_{ij}). \quad (6)$$

Under the naive Bayes' model assumption, the problem is a convex optimization problem with one hyper-parameter, the alphabet size F of the hidden variable H . Thus, we can use gradient descent to find λ^* and \mathbf{P}^* .

The power of this technique is that it allows us to write any marginal distribution of arbitrary degree using Eq. 1. Consider we have a group of clients G and we want to know the probability that a subset of that group $g \subset G$ will be able to access the channel while the remainder of G will not. This can be scalably found by:

$$\mathbb{P}(g, \overline{G \setminus g}) = \sum_{f=1}^F \lambda(f) \prod_{i \in g} p_i^f \prod_{j \in G \setminus g} (1 - p_j^f) \quad (7)$$

3) *High Order Distribution Accuracy Analysis:* This technique attempts to strike a balance between measurement overhead and the accuracy of the estimated HOD. It would simply not be possible to collect enough measurements to estimate the full distribution, nor would we be able to store the 2^N values for any moderate size value of N . On the other hand, if we start with a higher low-order marginal (e.g. three-way marginals instead of two) we could in principle increase the accuracy of the estimated HOD [20]. In practice, we find that using the pair-wise marginals as inputs is sufficient to achieve accurate HOD. Crucially, the modest gains provided by a higher order direct estimation do not outweigh the measurement overhead cost associated with them.

Fig. 5 shows the mean squared error between the true and estimated HOD, for $N = 20$ over 50 randomized seed values, with various latent variable sizes F . The estimated HODs are calculated using the measured pair-wise marginals. On increasing the alphabet size of the latent/hidden variable, we see that the higher-order marginals are estimated with lower error. However, the error does not reduce substantially when we increase the alphabet size from N to $2N$. Thus, in the results that follow in Section V, we use a latent variable of size $F = N$.

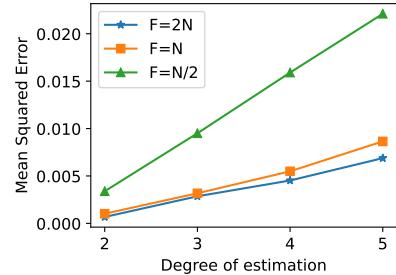


Fig. 5: Mean Squared Error between true and estimated HOD from pair-wise marginal inputs vs. latent variable size.

D. Blue-Printing Interference

The estimate of the HOD is sufficient for scheduling as we will see in Section IV-A, but it also provides the basis for generating a more accurate joint description of clients, and consequently interferers, in the physical space. Instead of treating the clients as users that are subject to interference, we change our perspective and use them as anchors/sensors with a finite search/impact radius. If we can successfully identify and assign interferers to groups of clients based on their impact, we can create an interference graph. Leveraging the interference graph along with location information of a limited set of clients, we can narrow down and locate HTs to a small, useful area. This has implications for jammer localization, as we will see in Sec. IV-B. We break down the process of generating the map of HTs into two steps, and illustrate using Fig. 6.

Step 1: HT-focused client clustering: Define $\mathbf{p}_i = \lambda^* \circ \mathbb{P}^*(i, :)$ where \circ is the Hadamard product. The vector $\mathbf{p}_i \in [0, 1]^F$ is a representation of the joint access probability distribution of client i and the realizations of the latent variable. Joint access vector of two clients with high correlation over the realizations of H , implies that the clients are affected by the same sets of HTs. On each channel, we form clusters of clients based on these vectors \mathbf{p}_i using DBSCAN [13]. In the associated example, this results in the clusters $\{0\}$, $\{1, 2, 8\}$, $\{3, 4\}$, $\{6, 7\}$, $\{9\}$, each colored differently, on the selected channel (Fig. 6(b)). We then identify a representative for each of the clusters at random. The number of clusters obtained is of the order of HTs active on the channel.

Step 2: Estimating and mapping HTs: Next, we characterize how groups of clients are jointly impacted. We first generate a dependency graph $G = (V, E)$ where each representative client $i \in V$ and $E = \{(i, j) | \mathbb{P}(i, j) \neq \mathbb{P}(i)\mathbb{P}(j)\}$. That is, an edge exists between two representative clients if the product of their first-order marginals is not equal to the pair-wise marginal, indicating that the channel access probabilities are not independent. A dependency graph is created based on these dependence relationships. In this example, the edges of the graph are $\{3, 5\}$, $\{5, 7\}$ and $\{0, 9\}$ (Fig. 6(c)).

We then take the following approach to approximate the number of HTs active on a channel. We note that

$$\mathbb{P}(\overline{g} | V \setminus g) = f(\text{HT impacts at least one of } g \text{ and not } V \setminus g) \quad (8)$$

This equation encodes the relationship between a HT and the likelihood it uses the same channel as the set of clients

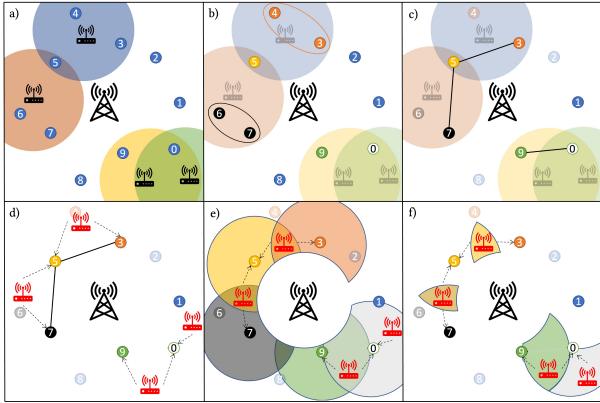


Fig. 6: Blue-printing interference and localizing jammers: a) True interference map, b) HOD clustering, c) Graph creation, d) ID HTs, e) Estimated zones, f) Refine zones.

in g . The function $f(\cdot)$ evaluates to 0 if and only if there is no such HT. This is the key to identifying the smallest descriptive set of HTs. Using Eq. 8, we obtain all such sets of clients affected *jointly* with non-zero probability. We do this by iterating over cliques g generated by graph G . We first consider the smallest size cliques and increment the size in each iteration—individuals, pairs, triples, etc.—and add a HT affecting the clique when the following is satisfied:

$$\mathbb{P}(\bar{g}|V \setminus g) > \prod_i \mathbb{P}(\bar{g}_i|V \setminus g) \quad \forall \text{ tuples } \{g_i\} \quad (9)$$

where $\cup_i g_i = g$ and g_i 's are sets found to be affected by HTs satisfying Eq. 8 in a previous iteration. Intuitively, this process identifies whether the clique g is affected by a *new* HT rather than an independent combination of the *existing* ones. This procedure is summarized in Algorithm 1.

In the accompanying example, we start with a clique size of 1. The only probability greater than 0 is the probability that client 0 is blocked conditioned on all others transmitting. Hence, we introduce a HT impacting client 0 (Fig. 6(d) HT furthest to the right). In the subsequent iteration (clique size 2), we consider the cliques $\{3,5\}$, $\{5,7\}$ and $\{0,9\}$. For each, the probability of being blocked while other clients can transmit is greater than zero. Hence, we introduce three additional HTs to explain the additional information (Fig. 6(d) remaining HTs).

Algorithm 1: Estimating Hidden Terminals

Data: Graph $G = (V, E)$, λ^*, \mathbf{P}^*
Result: Set of tuples (g) affected by unique hidden terminals

```

Q = formCliques(V,E);
g = emptyList(); /* Initializing g */
tuples = formTuples(g); /* set of all comb. of g */
for q in Q do
  for tuple in sortBySize(tuples) do
    if q == Union(tuple) then
      if P(̄g|V \ q) > ∏i P(̄tuplei|V \ q) then
        g.append(q); tuples = formTuples(g);

```

The result is to produce a topology like the one in Fig. 6(d) which maps four HTs to their respective impacted clients. In Section IV-B, we show how the estimated interference graph can be leveraged along with a limited amount of knowledge of the clients location to assist in localizing the HTs (shown in Fig. 6e and f).

IV. LEVERAGING *Neto-X* IN PRACTICE

We now describe two use cases in which knowledge of interference can significantly benefit the network: resource management and localization of adversarial jammers.

A. Application in Resource Management

1) *Current LTE/NR Schedulers*: We briefly capture the essence of existing cellular schedulers where the BS is responsible for allocating time-frequency resource blocks (RBs) to clients on both DL and UL. Scheduling policies that balance throughput and fairness, are typically derived using a utility-optimization approach. We consider the widely-adopted proportional fair (PF) scheduling policy [21].

For SISO systems with B RBs and N single-antenna clients, the schedule is a matrix $\mathbf{X} \in \{0, 1\}^{N \times B}$ that assigns the RBs to users in each time slot. If the set of all possible schedules is S , the optimal schedule aims to maximize the aggregate marginal utility:

$$\mathbf{X}^*(t) = \arg \max_{\mathbf{X} \in S} \left\{ \sum_{b=1}^B \sum_{i=1}^N \frac{x_{i,b} r_{i,b}(t)}{R_i(t-1)} \right\}, \sum_{i=1}^N x_{i,b} = 1, \forall b \quad (10)$$

where $r_{i,b}(t)$ and $R_i(t-1)$ are the instantaneous rate (on resource block b) and average throughput of client i , respectively, and the constraint on RB assignment $x_{i,b}$, ensures that only one user is scheduled on each RB.

For MU-MIMO systems, where the BS has several antennas (M), a group of users is selected, $G \subset \{1, \dots, N\}$, $|G| < M$, and resource blocks are allocated to the group. Here, the optimal schedule is:

$$\mathbf{Y}^*(t) = \arg \max_{\mathbf{Y} \in S} \left\{ \sum_{b=1}^B \sum_{i \in G} \frac{y_{i,b} r_{i,b}(t, G)}{R_i(t-1)} \right\}, \quad (11)$$

where $G = \{i | y_{i,b} > 0\}$ and $\sum_{i=1}^N y_{i,b} \leq M, \forall b$. It is important to note that the MU-MIMO rate of a client $r_{i,b}(t, G)$ varies based on the other clients in its MU-MIMO group. At the end of every schedule, the average throughput of all clients is updated through a weighted moving average, to accommodate dynamic traffic conditions.

2) *Interference-aware Scheduling*: The existing schedulers, originally designed for licensed spectrum, do not incorporate interference information. However, we progressively add interference information to the process. We consider the case of uplink, where the impact of HT interference is more pronounced. When each device (including both clients and HTs) employ LBT [4] before transmission in unlicensed spectrum, it is possible several of the BS's scheduled UL grants will not be utilized by the clients because of HT interference.

Recall that we can measure the probability of a client i utilizing its allocated grant, $\mathbb{P}(i)$. This indirect interference information can be incorporated to benefit the PF scheduler (Eq. 10, 11), so as to maximize the *expected* marginal utility .

We term this optimal scheduler *Access-Aware* (AA). For MU-MIMO the scheduler is:

$$\mathbf{Y}_{AA}^*(t) \underset{MU-MIMO}{=} \arg \max_{\mathbf{Y} \in S} \left\{ \sum_{b=1}^B \sum_{i=1}^N \frac{\mathbb{P}(i)y_{i,b}r_{i,b}(t, G)}{R_i(t-1)} \right\} \quad (12)$$

where the same constraints on $y_{i,b}$ from Eq. 11 apply. The equation for SISO can be recovered by setting $M = 1$.

3) *Joint Access-Aware Scheduling using HOD for MU-MIMO*: The aggregate rate of an MU-MIMO group not only depends on the set of users in the group, but also the ability of all users to transmit (subject to interference)—fewer participating streams results in higher SINR for the active streams in the group. Therefore, a more accurate model of rate is $r_i(t, g, G)$, which is a function of both the group G , and the *successful* transmissions $g \subseteq G$. We can compute $\mathbb{P}(g, G \setminus g)$, i.e. the probability that all clients in g can transmit and the remaining clients, $G \setminus g$ cannot because they are blocked by interference using the parameters λ, \mathbf{P} found in Section III-C1. The resulting scheduler is termed *Joint-Access-Aware* (JAA):

$$\mathbf{Y}_{JAA}^*(t) = \arg \max_{\mathbf{Y} \in S} \left\{ \sum_{b=1}^B \sum_{i=1}^N \sum_{g \subseteq G} \frac{\mathbb{P}(g, G \setminus g)y_{i,b}r_{i,b}(t, g, G)}{R_i(t-1)} \right\} \quad (13)$$

Choosing a user-group is non-trivial to solve computationally. We adopt a greedy approach to approximating the solution in a tractable time complexity (refer Eq. 16). The difference between AA (Eq. 12) and JAA (Eq. 13) is that AA assumes that all client transmissions in the group G are successful, which is sub-optimal.

4) *Speculative Scheduling using HOD for SISO*: With the JAA reducing to AA in a SISO scenario, one might question the value for HOD in SISO systems. Note that while AA incorporates access information to cope better with interference, it is not able to efficiently address it and maximize usage of its resource blocks. Through a policy termed *Speculative Scheduling* (SP) (introduced in [34]), we aim to leverage the additional information on these interferers to adaptively *over-schedule* clients on a RB. Overscheduling is analogous to a popular phenomenon of airlines overbooking flights in expectation that a certain number of people will not make it, leaving the exact number of seats occupied.

AA limits its vision to single-user scheduling, which naturally restricts the upper bound on RB utilization, determined by interference. We observe that different clients in the same cell can be interfered by different sets of HTs, leading to *interference diversity*. This creates opportunities to schedule multiple clients on the same RB with the expectation that only one of them will be able to transmit, preventing collisions. We primarily consider SISO since the opportunities for overscheduling diminish with higher multiplexing (MU-MIMO).

For instance, consider a client $i = \arg \max_k \left\{ \frac{\mathbb{P}(k)r_{k,b}}{R_k} \right\}$ that was selected by AA. The speculative scheduler tries to select an additional client j such that scheduling it on the same resource block will likely increase the utility because they are impacted by different HTs. Specifically:

$$j = \arg \max_{j \neq i} \left\{ \mathbb{P}(i, \bar{j}) \frac{r_{i,b}}{R_i} + \mathbb{P}(\bar{i}, j) \frac{r_{j,b}}{R_j} \right\}. \quad (14)$$

Note that we want *exactly* one of them to be able to transmit to allow for SISO decoding (lest a collision). For a given set of over-scheduled clients on an RB, its expected utility depends on the joint access probability distribution of the set/group

$$E(G) = \sum_{i \in G} \mathbb{P}(i, \overline{G \setminus i}) \frac{r_{i,b}(t)}{R_i(t-1)}. \quad (15)$$

The optimal scheduler for a SISO system is $\mathbf{G}^*(t) = \arg \max_G E(G)$. It is difficult to efficiently compute such groupings, and we resort to a greedy strategy described by Eq. 16. We start from a baseline G of two users and updating $G \leftarrow G \cup l^*$ after each iteration.

$$l^* = \arg \max_{l \notin G} (E(G \cup l) - E(G)) \quad (16)$$

B. Application in Jammer Localization

Our framework for *blue-printing* the network interference from Section III-D, helped identify the HTs, the sets of clients they impact, and the magnitude of impact. An immediate use-case of such HT-client mapping is to identify the *location* of the HTs. This is of special interest when the HTs are adversarial and the interference is an active, yet subtle (not always ON) attempt to impact the network.

We accomplish this by changing the anchoring of the interference from clients to physical space. Specifically, we request the coordinates of representatives $\{(x_i, y_i)\}_{i \in R}$, one each from the clusters identified through Alg. 1. Now, the region of impact around a particular client i is an area called $A_i := \{(x, y) | \text{distance}((x, y), (x_i, y_i)) < D\}$, where D is an estimate of the maximum range between a client and a HT, given the client's energy detection threshold (and maximum transmit power in the band). A HT located in the area A_i is assumed to impact the access of client i , located at (x_i, y_i) , while a HT outside it does not. Now, given a HT obtained using Alg. 1 and the group of clients it impacts (g), we can narrow down its location to the following area,

$$A_{HT} = \{\cap_{i \in g} A_i\} \cap \{\cap_{i \in R \setminus g} A_i^c\} \cap A_{BS}^c \quad (17)$$

where, A_i^c is the complementary area to A_i and A_{BS} is the area around the base station with range D . In other words, the HT is restricted to the common area determined by its affected clients and outside the area of both its unaffected clients as well as the BS (by definition of a HT). This process is captured in Fig. 6(e) and (f). We take the centroid of the resulting area to be the HT's location. Thus, with the locations of a few clients, we are able to construct an approximate physical map of the interferers. While this indirect localization is not exact, it is nonetheless impressive (median accuracy under 10m, Sec. V-B) and sufficient in our application.

V. EVALUATION

We evaluate *Neto-X* comprehensively in both resource management and jammer localization applications using a combination of NS3 [1] and numerical simulations.

We are able to instrument NS3 for DSA on a single channel SISO system using the LAA-LBT implementation [14]. In these simulations, we consider 100 topologies (similar to Fig. 4). The HTs are distributed uniformly, but to ensure they are “hidden” from the BS, they are placed at least 70m away,

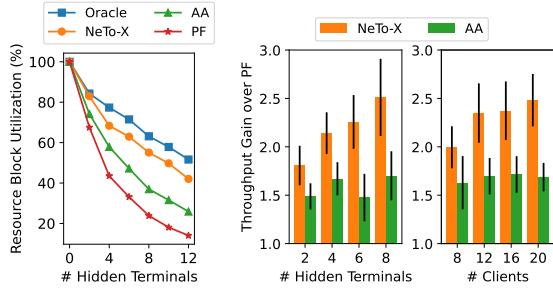


Fig. 7: a) RB utilization on a single channel SISO system, b) Throughput gains over PF with varying HTs and c) clients.

where $50m$ is approximately the radius of the HT’s zone of impact. There are 2 to 8 HTs (Wi-Fi APs) randomly distributed amongst 8 to 20 clients. The BS is centrally located and the Wi-Fi nodes are out of its range, but impact the clients. NS3 currently supports LBT only at the BS (not at clients). Hence, we incorporate the impact of collisions using the physical layer traces. While this works for SISO, it is not conducive for multi-channel and MU-MIMO scenarios with many clients, wherein we supplement with numerical simulations. Our topologies for numerical evaluations mimic our NS3 network set-up. In all our simulations, clients are uniformly distributed in a circular area bounded by the BS’s transmission range.

To estimate the HOD, the BS follows the procedure in Section III. The first- and second- order marginals are estimated over 1000 frames. Unless specified, the number of clusters is N/C , for N clients and C channels. Each HT has a fixed probability of transmission drawn uniformly from the interval $[0.2, 0.8]$. If any node, either a client or another HT, is within the HT’s zone of impact, the probability that they will not be able to transmit at a given frame is dependent on the HT transmit probability. Simulations consists of scheduling 1,500 LTE/5G frames. In each frame the BS attempts to schedule the N clients on $B = 10$ available RBs according to the schedulers presented in Section IV-A.

A. Resource Management

First, we look at how the estimated HOD can be used to optimize client scheduling. We compare *NeTo-X* to PF and AA schedulers as described in Section IV-A as well as an optimal *oracle* scheduler. Our metrics include resource block utilization (RBU), which captures how efficient the scheduler is. We also report a measure of the cumulative throughput a schedule can achieve which is simply the sum of the long-term rates, and is normalized to that achieved by PF. Note that fairness is automatically incorporated by the schedulers in their policy. Additionally, we investigate the trade-off between performance and the measurement overhead.

1) *Impact of Interference*: We aim to understand the scheduler performance with increasing levels of interference. Fig. 7(a) and (b) capture the effect of adding hidden terminals on the same channel for 20 clients using NS3. Fig. 7(a) shows the percentage of resource blocks successfully utilized in a single-channel SISO system for 20 clients. The resource block utilization of all schedulers, including the *oracle* decrease as

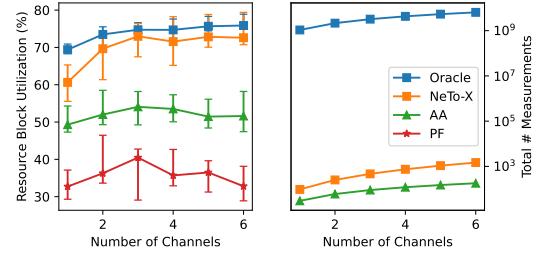


Fig. 8: Resource block utilization per channel and measurement overhead of SISO schedulers.

the level of interference increases, highlighting the inherent challenges of DSA. The PF scheduler performs worse since it is oblivious to the interference. Beyond eight HTs, its efficiency falls below 25%. The AA scheduler, which simply weights the PF schedule with the first-order marginals, performs only slightly better. However, by estimating the HOD with *NeTo-X*, and performing speculative scheduling, we are almost able to achieve the the performance of the *oracle*, differing by only a few percent. In Fig. 7(b), with an increase in the number of hidden terminals, *NeTo-X* minimizes losses through speculative scheduling to yield an appreciable gain of 50% over AA and 2x over PF.

2) *Impact of Client Density*: The LAA-LBT simulations in NS3 support a limited number of clients. In Fig. 7(c), we observe that the throughput gains over PF increase substantially with an increasing number of clients, wherein the diversity of interference impact is more. *NeTo-X* is able to leverage this diversity to speculatively schedule more groups of clients together. These gains are also observed for a much larger client density in our numerical evaluations.

3) *Multi-Channel Performance*: Next, we would like to investigate how the number of channels impact scheduling performance. Recall that modern LTE/5G networks use multiple channels through carrier aggregation to increase overall throughput, but this increases resource management complexity especially in the interference-rich environment of DSA.

The performance and measurement overhead of a SISO system with 30 clients vs. number of channels is shown in Fig. 8. To isolate the effect of the channels from the level of interference, the number of hidden terminals per channel is fixed at 4. The performance of PF and AA is independent of the number of channels as they don’t utilize cross-channel information. In contrast, *NeTo-X* clusters clients affected together, enhancing joint access probability estimation across channels for more efficient channel allocation. This allows *NeTo-X* to approach the performance of the *oracle* scheduler with fewer measurements (a factor of 10^{-6}) as the number of channels increases.

4) *Multi-Antenna Systems*: We now investigate the benefits of *NeTo-X* for scheduling in MU-MIMO systems. *NeTo-X* outperforms both PF and AA schemes for client scheduling in MU-MIMO systems as seen in Fig. 9. Indeed, the cumulative throughput of *NeTo-X* grows linearly with the number of antennas, while the PF scheduler starts exhibiting saturation with more antennas and is unable to deliver the promised

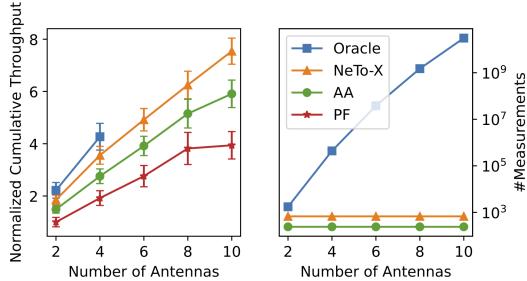


Fig. 9: Normalized total throughput and overhead of MU-MIMO vs. number of antennas.

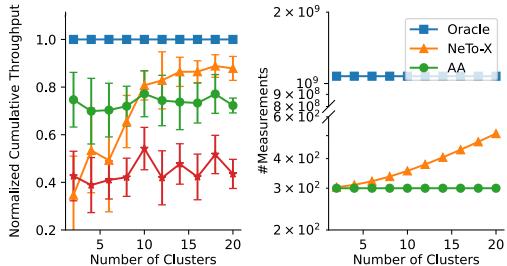


Fig. 10: Effect of the number of clusters used by *NeTo-X* on the performance and measurement overhead.

gains. For a limited number of antennas we are able to compare *NeTo-X* to the *oracle*; however, as the number of antennas increases, the computational and storage overhead, let alone the required number of measurements, increases to a point where even simulation becomes infeasible. On the other hand, *NeTo-X* has a significantly smaller measurement overhead which *does not* scale with the number of antennas and clients making its implementation computationally feasible.

5) *Clustering Analysis*: One of the design parameters is the number of clusters, K , formed after obtaining the first-degree marginals as described in Section III-A. By grouping clients into clusters, we are making the assumption that they are impacted by the same HTs. A smaller number of clusters requires fewer measurements to estimate the pairwise marginals, but it may mean that we have made incorrect assumptions about the nature of the interference. The trade off between scheduling performance and measurement overhead as a function of the number of clusters is captured in Fig. 10. Aggressive clustering ($K < 10$ for 150 clients) results in the least amount of measurements but also causes *NeTo-X* to underperform compared to AA and potentially even PF. Increasing the number of clusters provides a better estimation of the joint access distribution at the cost of an increase in the number of measurements which scales as $\binom{K}{2}$, though this overhead is still significantly less than that of the *oracle*.

B. Evaluation for Jammer Localization

To evaluate how well *NeTo-X* can be used to locate jammers, we simulate the presence of 2-8 HTs with 40 clients over 200 randomized topologies. For each topology, we try to localize jammers using the method described in Section IV. We perform the same analysis using the locations of (1) all UEs and (2) only the representatives described in Section III-D. The number of representatives ranges from 3 to 10 clients

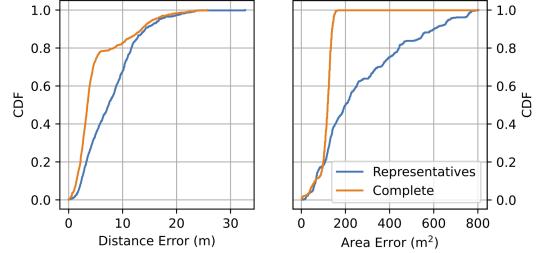


Fig. 11: Accuracy (left) and Precision (right) of *NeTo-X* in localizing jammers with 40 clients and 3 hidden terminals

Number of HTs	2	3	4	6	8
Accuracy (%)	99.3	90.5	81.7	73.5	70.6

TABLE I: Classification accuracy of *NeTo-X*

depending on the topology. After narrowing down the HTs to a candidate zone, we measure the metrics

- *Accuracy*: the distance between the ground truth location of the HT and the centroid of the candidate zone,
- *Precision*: the total area of the candidate zone.

Fig. 11 depicts the accuracy and precision. When using all clients' location as anchors, the median accuracy is about 5 m. Using only the representatives' locations, the median accuracy reduces to 10 m. This granularity is still sufficient to infer the location of the jammers with high likelihood.

However, using the locations of only the representatives makes pinpointing the location of jammers precisely more challenging. The client placement is sparse and narrowing down the candidate zone is difficult. While the median precision is 200 m², there is a long tail which extends beyond 400 m² in the 80th percentile. When we have complete information of the locations, *NeTo-X* is able to determine an appreciably smaller candidate zone.

Table I shows the accuracy with which *NeTo-X* identifies the number of HTs present on a channel over 200 sample topologies with 40 clients. *NeTo-X* accurately identifies the number of jammers on the network (> 90% for up to 3 HTs). However, for a larger number of HTs, the accuracy dips to 70% as some HTs are very closely located to each other and tend to present themselves as a single, consolidated HT. However, this might still be satisfactory to localize the area of aggregate HT presence.

VI. CONCLUSIONS

As future wireless networks explore flexible spectrum sharing models, understanding the impact of external interference has become central to delivering the increased spectral efficiencies offered by their multi-antenna, multi-channel solutions. To this end, we proposed a novel, scalable network tomography framework called *NeTo-X* that transforms clients into sensors, and leverages their joint channel access statistics to accurately characterize the interference sources and their impact at a highly scalable overhead. Its merits are showcased in the context of a resource management and jammer localization application, where its performance significantly outperforms baseline approaches, and closely approximates optimal performance at significantly (several orders of magnitude) lower overhead.

REFERENCES

[1] Network simulator (ns-3). <https://www.nsnam.org/>.

[2] LTE in unlicensed spectrum: Harmonious coexistence with Wi-Fi. *Qual-comm Research*, June 2014.

[3] 3GPP. Evolved universal terrestrial radio acces (e-utra); medium access control (mac) protocol specification (release 10). In *TS 36.321*.

[4] 3GPP Release 14, 36.213. Channel Access Procedures for LAA, 2016.

[5] 3GPP Technical Report 36.889. Feasibility Study on Licensed-Assisted Access to Unlicensed Spectrum, 2015.

[6] W. S. H. M. W. Ahmad, N. A. M. Radzi, F. S. Samidi, A. Ismail, F. Abdullah, M. Z. Jamaludin, and M. N. Zakaria. 5g technology: Towards dynamic spectrum sharing using cognitive radio networks. *IEEE Access*, 8:14460–14488, 2020.

[7] Monica Alveen. CBRS Alliance completes 5G specs, February 2020. Section: Fierce Wireless Homepage. Tech, 5G, Wireless.

[8] Ran Ben Basat, Sivaramakrishnan Ramanathan, Yuliang Li, Gianni Antichi, Minian Yu, and Michael Mitzenmacher. Pint: Probabilistic in-band network telemetry. In *Proceedings of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication, SIGCOMM '20*, page 662–680, New York, NY, USA, 2020.

[9] Ivan Brugere, Brian Gallagher, and Tanya Y. Berger-Wolf. Network Structure Inference, A Survey: Motivations, Methods, and Applications. *ACM Comput. Surv.*, 51(2):24:1–24:39, April 2018.

[10] Tian Bu, Nick Duffield, Francesco Lo Presti, and Don Towsley. Network tomography on general topologies. In *Proceedings of the 2002 ACM SIGMETRICS international conference on Measurement*, SIGMETRICS '02, pages 21–30, New York, NY, USA, June. ACM.

[11] CBRS Alliance. Citizen broadband radio service. www.cbrsalliance.org/, 2016.

[12] Qimei Chen, Guanding Yu, Hangguan Shan, Amine Maaref, Geoffrey Ye Li, and Aiping Huang. An opportunistic unlicensed spectrum utilization method for lte and wifi coexistence system. In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, 2015.

[13] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, Portland, Oregon, 1996. AAAI Press.

[14] Lorenza Giupponi, Thomas Henderson, Biljana Bojovic, and Marco Miozzo. Simulating lte and wi-fi coexistence in unlicensed spectrum with ns-3, 2016.

[15] Robert W. Heath, Marios Kountouris, and Tianyang Bai. Modeling heterogeneous network interference using poisson point processes. *IEEE Transactions on Signal Processing*, 61(16):4114–4126, 2013.

[16] Mohammed Hirzallah, Marwan Krunz, Balkan Kecicioglu, and Belal Hamzeh. 5g new radio unlicensed: Challenges and evaluation. *IEEE Transactions on Cognitive Communications and Networking*, 7(3), 2021.

[17] Lin Hu and Zhijian Dai. Performance and reliability analysis of prioritized safety messages broadcasting in dsrc with hidden terminals. *IEEE Access*, 8:177112–177124, 2020.

[18] Kai-Chuang Huang and Kwang-Cheng Chen. Interference analysis of nonpersistent csma with hidden terminals in multicell wireless data networks. In *Proceedings of 6th International Symposium on Personal, Indoor and Mobile Radio Communications*, volume 2, pages 907–911 vol.2, 1995.

[19] Shahana Ibrahim and Xiao Fu. Recovering joint probability of discrete random variables from pairwise marginals. *IEEE Transactions on Signal Processing*, 2021.

[20] Nikos Kargas, Nicholas D. Sidiropoulos, and Xiao Fu. Tensors, learning, and “kolmogorov extension” for finite-alphabet random vectors. *IEEE Transactions on Signal Processing*, 66(18):4854–4868, 2018.

[21] H.J. Kushner and P.A. Whiting. Convergence of proportional-fair sharing algorithms under general conditions. *IEEE Transactions on Wireless Communications*, 3(4):1250–1259, 2004.

[22] LTE-U Forum White Paper. Lte-u, 2014.

[23] Thomas L. Marzetta. Noncooperative cellular wireless with unlimited numbers of base station antennas. *IEEE Transactions on Wireless Communications*, 9(11):3590–3600, 2010.

[24] A. Mukherjee, Jung-Fu Cheng, S. Falahati, L. Falconetti, A. Furuskar, B. Godana, Du Ho Kang, H. Korapat, D. Larsson, and Yu Yang. System architecture and coexistence evaluation of licensed-assisted access lte with ieee 802.11. In *Communication Workshop (ICCW), 2015 IEEE International Conference on*, 2015.

[25] Gaurang Naik, Jung-Min Park, Jonathan Ashdown, and William Lehr. Next generation wi-fi and 5g nr-u in the 6 ghz bands: Opportunities and challenges. *IEEE Access*, 8:153027–153056, 2020.

[26] Qualcomm White Paper. License Assisted Access (LAA). <https://www.qualcomm.com/invention/technologies/lte/laa>.

[27] Braulio Pinto, Raimundo Barreto, Eduardo Souto, and Horacio Oliveira. Robust rssi-based indoor positioning system using k-means clustering and bayesian estimation. *IEEE Sensors Journal*, 21(21):24462–24470, 2021.

[28] Qualcomm White Paper. MulteFire Specification. www.multefire.org/specification, 2016.

[29] R. Ratasuk, D. Tolli, and A. Ghosh. Carrier aggregation in lte-advanced. In *IEEE VTC*, May 2010.

[30] Nadisanka Rupasinghe and İsmail Güvenç. Reinforcement learning for licensed-assisted access of lte in the unlicensed spectrum. In *2015 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1279–1284, 2015.

[31] P. S. Ruzankin. On the rate of poisson process approximation to a bernoulli process. *Journal of Applied Probability*, 41(1):271–276, 2004.

[32] Shweta Sagari, Samuel Baysting, Dola Saha, Ivan Seskar, Wade Trappe, and Dipankar Raychaudhuri. Coordinated dynamic spectrum management of lte-u and wi-fi networks. In *2015 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pages 209–220, 2015.

[33] Shamik Sarkar, Milind Buddhikot, Aniqua Baset, and Sneha Kumar Kasera. Deepradar: A deep-learning-based environmental sensing capability sensor design for cbrs. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*, MobiCom '21, page 56–68, New York, NY, USA, 2021. Association for Computing Machinery.

[34] Ramanujan K. Sheshadri, Karthikeyan Sundaresan, Eugene Chai, Amir Khojastepour, Sampath Rangarajan, and Dimitrios Koutsoukolas. Blu: Blue-printing interference for robust lte access in unlicensed spectrum. In *Proceedings of the 13th International Conference on Emerging Networking EXperiments and Technologies*, CoNEXT '17, page 15–27, New York, NY, USA, 2017. Association for Computing Machinery.

[35] Nofel Yaseen, John Sonchack, and Vincent Liu. Synchronized network snapshots. *SIGCOMM '18*, page 402–416, New York, NY, USA, 2018. Association for Computing Machinery.

[36] Sangki Yun and Lili Qiu. Supporting wifi and lte co-existence. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 810–818, 2015.