A. A. Gargari, A. Ortiz, M. Pagin, W. de Sombre, M. Zorzi, A. Asadi ”Risk-Averse Learning for Reliable mmWave Self-Backhauling”, in IEEE/ACM Transactions on Networking, 2024.

Risk-Averse Learning for Reliable mmWave Self-Backhauling

Amir Ashtari Gargari, Andrea Ortiz, *Member, IEEE*, Matteo Pagin, Wanja de Sombre, Michele Zorzi, *Fellow, IEEE*, and Arash Asadi, *Senior Member, IEEE*

***Abstract*— Wireless backhauling at millimeter-wave frequencies (mmWave) in static scenarios is a well-established practice in cellular networks. However, highly directional and adaptive beamforming in today's mmWave systems have opened new possibilities for self-backhauling. Tapping into this potential, 3GPP has standardized Integrated Access and Backhaul (IAB) allowing the same base station to serve both access and backhaul traffic. Although much more cost-effective and flexible, resource allocation and path selection in IAB mmWave networks is a formidable task. To date, prior works have addressed this challenge through a plethora of classic optimization and learning methods, generally optimizing Key Performance Indicators (KPIs) such as throughput, latency, and fairness, and little attention has been paid to the reliability of the KPI. We propose Safehaul, a risk-averse learning-based solution for IAB mmWave networks. In addition to optimizing the average performance, Safehaul ensures reliability by minimizing the losses in the tail of the performance distribution. We develop a novel simulator and show via extensive simulations that Safehaul not only reduces the latency by up to 43.2% compared to the benchmarks, but also exhibits significantly more reliable performance, e.g., 71.4% less variance in latency.**

Manuscript received 5 October 2023; revised 21 May 2024; accepted 11 August 2024; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor S. Ioannidis. Date of publication 16 September 2024; date of current version 19 December 2024. This work was supported in part by the Deutsche Forschungsgemeinschaft [German Research Foundation (DFG)] within the mm-Cell project and the Collaborative Research Center 1053 Multi-Mechanisms Adaptation for the Future Internet (MAKI), in part by the Landes-Offensive zur Entwicklung Wissenschaftlichökonomischer Exzellenz [State Offensive for the Development of Scientific and Economic Excellence, Hesse, Germany (LOEWE)] initiative (Hesse, Germany) within the emergenCITY center, in part by the Bundesministerium für Bildung und Forschung [Federal Ministry of Education and Research, Germany (BMBF)] through the Open6GHub project, in part by European Commission under Grant 861222 (H2020 ITN MINTS project), in part by the Ministerio de Ciencia e Innovación/Agencia Estatal de Investigación [Ministry of Science and Innovation/State Research Agency, Spain (MCIN/AEI)]/10.13039/501100011033 and "European Regional Development Fund (ERDF)—A way of making Europe" under Grant PID2021-126431OB-I00, and in part by Generalitat de Catalunya under Grant 2021 SGR 00770. Part of this paper has been presented at INFOCOM 2023 [3]. *(Corresponding author: Arash Asadi.)*

Amir Ashtari Gargari is with the Department of Information Engineering, University of Padova, 35131 Padua, Italy, and also with the Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA), 08860 Barcelona, Spain.

Andrea Ortiz is with the Communications Engineering Laboratory, TU Darmstadt, 64283 Darmstadt, Germany, and also with the Institute of Telecommunications, TU Wien, 1040 Vienna, Austria.

Matteo Pagin and Michele Zorzi are with the Department of Information Engineering, University of Padova, 35131 Padua, Italy.

Wanja de Sombre is with the Communications Engineering Laboratory, TU Darmstadt, 64283 Darmstadt, Germany.

Arash Asadi was with the Department of Computer Science, TU Darmstadt, 64283 Darmstadt, Germany. He is now with the Embedded Systems Group, TU Delft, 2628 CD Delft, The Netherlands (e-mail: a.asadi@tudelft.nl).

Part of this paper has been presented at INFOCOM 2023 [3]. This work was conducted while Arash Asadi was affiliated with TU Darmstadt, Germany.

Digital Object Identifier 10.1109/TNET.2024.3452953

***Index Terms*— Millimeter-wave communication, integrated access and backhaul (IAB), self-backhauling, wireless backhaul.**

I. INTRODUCTION

THE emergence of mmWave cellular systems created a unique opportunity for Mobile Network Operators (MNOs) to leverage a scalable and cost-effective approach to deal with network densification. The fact that mmWave base stations can support fiber-like data rates facilitates the use of the same base station for both access and backhaul traffic, a solution which in 3GPP parlance is referred to as IAB. Consequently, 3GPP has included IAB in the standard [1], [2] covering the details on architecture, higher layer protocols, and the radio. Although Release 17 of 5G-NR defines the interfaces, architectures, and certain system parameters, the actual configuration and resource allocation is left to MNOs.

Traditional self-backhauled networks featured fixed-wireless links decoupled from access networks with static configurations. In contrast, IAB should account for the dynamic nature of the backhaul links (particularly in mmWave deployments) and their integration with the access network. Further, IAB allows the traffic to traverse several hops (i.e., base stations) to reach its destination, thus increasing the problem's complexity. *In addition to the scheduling problem, an IAB network should:* (i) *solve the problem of path selection and link activation at the backhaul while considering inter-cell interference, and* (ii) *decide on serving access or backhaul traffic depending on the access load and the ingress backhaul traffic from neighboring base stations.*

Prior work. Methodologically, the majority of the existing works [4], [5], [6], [7], [8], [9], [10], [11], [12] focus on classic optimization techniques to solve the above-mentioned problem. However, given the large number of parameters involved, such formulations often result in non-convex problems that are too complex for real-time operations, but are nonetheless valuable indicators as performance upper bounds. Recently, some works focus on more practical solutions which can be deployed in real networks [13], [14], [15]. Specifically, these works leverage Reinforcement Learning (RL) to tackle both resource allocation and/or path selection in IAB mmWave networks and demonstrate that RL-based solutions achieve real-time performance.

Regardless of the methodology, prior works mostly aim at maximizing the network capacity [4], [5], [6], [7], [8], [9], [10], [11], minimizing latency [16], [17] and improving

throughput fairness [5], [18]. Although these approaches successfully improve the network performance, MNOs are often more *concerned about their reliability*. For this reason many commercial products rely on *simplified* but reliable algorithms for resource allocation, despite their sub-optimal performance. In this article, we propose Safehaul, a reinforcement learning-based solution for reliable scheduling and path selection in IAB mmWave systems under network dynamics. We use the concept of risk aversion, commonly used in economics [19], [20], to measure and enhance the reliability of Safehaul. The following summarizes our contributions:

- We model the scheduling and path selection problem in IAB mmWave networks as a multi-agent multi-armed bandit problem (Section III). We consider multiple fiber base stations, simultaneously supporting many self-backhauled mmWave base stations. In our model, the self-backhauled base stations independently decide the links to activate. The consensus among the base stations is reached via standard-defined procedures (Section IV-C).
- We present the first solution to provide reliable performance in IAB-enabled networks (Section IV). Specifically, we investigate the joint minimization of the average end-to-end latency and its expected tail loss. To this aim, we propose Safehaul, a learning approach that leverages the coherent risk measure Conditional Value at Risk (CVaR) [19]. CVaR measures the tail average of the end-to-end latency distribution that exceeds the maximum permitted latency, thus ensuring the network's reliability.
- We analytically bound the regret of Safehaul, i.e., we bound the loss of Safehaul compared to the case when the delays associated to all end-to-end paths between self-backhauled base stations and fiber base stations are known a priori. We show that, for the case when there are no conflicts between the decisions of the self-backhauled base stations, the average regret of Safehaul tends to zero as the time increases. This regret bound characterizes the learning speed and proves that Safehaul converges to the optimal scheduling and path selection solution that jointly minimizes the average end-to-end latency and its expected tail loss.
- We provide a new means of simulating multi-hop IAB networks by extending NVIDA's GPU-accelerated simulator Sionna [21] (Section V). Specifically, we add codebook-based analog beamforming capabilities for both uplink and downlink communications. In addition, we add internal Ray-tracing (RT) of Sionna in order to generate Channel Impulse Response (CIR). Further, we extend Sionna by implementing system-level components such as layer-2 schedulers and buffers and Backhaul Adaptation Protocol (BAP)-like routing across the IAB network. We believe our IAB extensions will be instrumental for the open-source evaluation of future research on self-backhauled mmWave networks.
- Exploiting the above simulator, we evaluate and benchmark Safehaul against two state-of-the-art algorithms [17], [22] based on deployment in two different locations (Manhattan and Padova). The results confirm that Safehaul is significantly more reliable than the considered benchmarks, as it exhibits much tighter variance in terms of both latency (up to 71.4% smaller) and packet drop rate (at least 39.1% lower). Further, Safehaul achieves up to 43.2% lower average latency and 11.7% higher average throughput than the reference schemes.

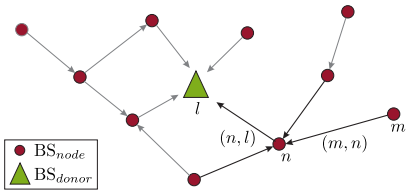


Fig. 1. Example of a graph $\mathcal{G}_i$.

II. System Model

We consider a cellular system with N base stations capable of self-backhauling and D base stations with a fiber connection to the core network. Following 3GPP terminology, we refer to self-backhauled base stations as IAB-nodes (BS-nodes) and to fiber base stations as IAB-donors (BS-donors).[1] Each BS-node connects to the core network via a (multi-hop) wireless link to a BS-donor. The sets of all BS-nodes and BS-donors are denoted by $\mathcal{N} = \{1, \ldots, N\}$ and $\mathcal{D} = \{N+1, \ldots, N+D\}$, respectively. The system works in a time-slotted fashion starting from time slot $i = 1$ until a finite time horizon I. All the time slots $i = 1, \ldots, I$ have the same duration. The BS-nodes are equipped with two RF chains. One RF chain is used exclusively for the communication with cellular users (access network), while the second RF chain is used for self-backhauling. In line with the 3GPP specification [23], we assume half-duplex self-backhauling, i.e., in each time slot i a BS-node can either transmit, receive or remain idle.

We model the connections between the base stations in slot i as a graph $\mathcal{G}_i = \{\mathcal{V}, \mathcal{E}_i\}$, see Fig. 1. The set $\mathcal{V} = \mathcal{N} \cup \mathcal{D}$ of vertices is formed by all the BS-nodes and BS-donors in the system. The set $\mathcal{E}_i$ of edges is composed of the available wireless links (n, l) between a BS-node $n \in \mathcal{N}$ and any BS (BS-donor or BS-node) $l \in \mathcal{V}$ in time slot i. Note that $\mathcal{G}_i$ is not static. In a given time slot i, some links may be unavailable due to failure, blockage, or interference. Thus, only feasible wireless links are considered in the set $\mathcal{E}_i$. The path $X_{n,d}$ from BS-node n to any BS-donor d is a sequence of intermediate links (n, l). $X_{n,d}$ changes over time according to the traffic loads of the intermediate BS-nodes and to the channel conditions. We model the activation of link (n, l) with the binary variable $x_{n,l,i}$. When $x_{n,l,i} = 1$, the link is activated and BS-node n transmits to BS $l \in \mathcal{V}$ in time slot i, whereas $x_{n,l,i} = 0$ indicates that the link is deactivated. $x_{n,n,i} = 1$ indicates that BS-node n does not transmit nor receives backhaul data in time slot i.

Each BS-node n has a finite data buffer with capacity $B_n^{\max}$ to store the backhaul data to be transmitted to any of the

[1]Please note that throughout the paper we will use interchangeably BS-nodes and IAB-nodes (and similarly for BS-donors and IAB-donors).

BS-donors. In each time slot i, BS-node n is characterized by its load and average queuing time. The load, denoted by $B_{n,i} \in \mathbb{N}$, indicates the number of data packets stored in the buffer at the beginning of time slot i. The average queuing time $t^{\mathrm{q}}_{n,i} \in \mathbb{R}^+$ is the average number of time slots the current packets in the data buffer have been stored. Additionally, we denote by $M_{n,l,i} \in \mathbb{N}$ the number of data packets transmitted from n and successfully received at l in time slot i (i.e., when $x_{n,l,i} = 1$), and with $t^{\mathrm{tx}}_{n,l,i} \in \mathbb{R}^+$ the transmission time needed to send these packets. Note that $M_{n,l,i} \leq B_{n,i}$ as only packets stored in the data buffer can be transmitted. At the receiver BS-node l, the load $B_{l,i+1}$ of its data buffer is updated at the beginning of the next time slot $i+1$ such that $B_{l,i} + M_{n,l,i} \leq B^{\max}_l$ holds. That is to say, packets exceeding the buffer capacity are dropped. Finally, when $x_{n,l,i} = 0$ both $M_{n,l,i}$ and $t^{\mathrm{tx}}_{n,l,i}$ are equal to zero.

We define the maximum tolerable latency $T_{\max}$ as the maximum time a packet can take from its source BS-node to any BS-donor. Any packet that is not delivered before $T_{\max}$ milliseconds will be dropped. The average maximum end-to-end latency $\bar{T}_{n,d}$ from BS-node n to BS-donor d is the average, over the complete time horizon I, of the maximum delay a packet originating from BS-node n takes to reach any BS-donor d in time slot i. This is calculated as $\bar{T}_{n,d} = \frac{1}{I}\sum_{i=1}^{I} T_{n,d,i}$, where $T_{n,d,i}$ is the maximum end-to-end latency among all the packets originating in BS-node n which reach BS-donor d in time slot i. $T_{n,d,i}$ is a sample of the random variable $T_{n,d}$ drawn from an unknown stationary probability distribution P that depends on the links $x_{n,l,i'}$, $n \in \mathcal{N}$, $l \in \mathcal{V}$, $i' = 1, \ldots, i$, activated up to time i, the user's mobility, the location of the BS-node n, the interference in the system, and the queue dynamics. Accordingly, we define the average maximum end-to-end latency in the system $\bar{T}$ as

$$\bar{T} = \frac{1}{ND}\sum_{n=1}^{N}\sum_{d=1}^{D} \bar{T}_{n,N+d}. \tag{1}$$

III. Problem Formulation

The joint minimization of the average maximum end-to-end latency and the expected value of its tail loss in IAB-enabled networks is formulated in this section. We first introduce CVaR, the risk metric accounting for minimizing the events in which the end-to-end latency is higher than $T_{\max}$. Next, we formulate the optimization problem in the complete network.

A. Preliminaries on CVaR

Traditionally, latency minimization in IAB-enabled networks has focused on optimizing the expected value of a latency function [16], [17]. However, such an approach fails to capture the time variability of the latency distribution, thus potentially leading to unreliable systems in which $T_{n,d,i} > T_{\max}$, for any $i = 1, \ldots, I$, $n \in \mathcal{N}$ and $d \in \mathcal{D}$. *For this purpose, we consider not only the average end-to-end latency $\bar{T}$ in the system, but also its expected tail loss based on the CVaR* [19], [24].

Having in mind that $T_{n,d}$ is a random variable, we assume it has a bounded mean on a probability space $(\Omega, \mathcal{F}, P)$, with Ω and $\mathcal{F}$ being the sample and event space, respectively. Using a risk level $\alpha \in (0, 1]$, the $\mathrm{CVaR}_\alpha(T_{n,d})$ of $T_{n,d}$ at risk level α quantifies the losses that might be encountered in the α-tail. More specifically, it is the expected value of $T_{n,d}$ in its α-tail distribution [24]. Formally, $\mathrm{CVaR}_\alpha(T_{n,d})$ is defined as [19]

$$\mathrm{CVaR}_\alpha(T_{n,d}) = \min_{q \in \mathbb{R}} \left\{ q + \frac{1}{\alpha}\mathbb{E}\left[\max\{T_{n,d} - q, 0\}\right] \right\}, \tag{2}$$

where the expectation in (2) is taken over the probability distribution P. Note that lower $\mathrm{CVaR}_\alpha(T_{n,d})$ results in higher system reliability because the expected end-to-end latency in the α-worst cases is low. Moreover, note that α is a risk aversion parameter. For $\alpha = 1$, $\mathrm{CVaR}_\alpha(T_{n,d}) = \mathbb{E}[T_{n,d}]$ which represents the traditional risk-neutral case. Conversely, $\lim_{\alpha \to 0} \mathrm{CVaR}_\alpha(T_{n,d}) = \sup\{T_{n,d}\}$. CVaR has been shown to be a coherent risk measure, i.e., it fulfills monotonicity, subadditivity, translation invariance, and positive homogeneity properties [25].

B. Optimization Problem

We jointly minimize the average maximum end-to-end latency and its expected tail loss for each BS-node. For this purpose, we decide which of the (n, l) links to activate in each time slot i during the finite time horizon I. In the following, we formulate the optimization problem from the network perspective and consider the sum over all BS-nodes in the system. The latency minimization problem should consider three different aspects: (i) link activation is constrained by the half-duplex nature of self-backhauling, (ii) only data stored in the data buffers can be transmitted, and (iii) packet drop due to buffer overflow should be avoided. Formally, the problem is written as:

$$\underset{\{x_{n,l,i}\}}{\text{minimize}} \sum_{n \in \mathcal{N}} \left(\sum_{d \in \mathcal{D}} \left(\frac{1}{I}\sum_{i=1}^{I} T_{n,d,i} \right) + \eta \mathrm{CVaR}_\alpha(T_{n,f}) \right) \tag{3a}$$

$$\text{subject to} \sum_{l \in \mathcal{V}, l \neq n} x_{n,l,i} + \sum_{l \in \mathcal{N}} x_{l,n,i} = 1, \quad n \in \mathcal{N},\ i = 1, \ldots, I \tag{3b}$$

$$B_{n,i} \geq M_{n,l,i}, \quad n \in \mathcal{N}, l \in \mathcal{V},\ i = 1, \ldots, I \tag{3c}$$

$$B_{l,j} + M_{n,l,j} \leq B^{\max}_l, \quad n \in \mathcal{N}, l \in \mathcal{V},\ i = 1, \ldots, I \tag{3d}$$

$$x_{n,l,i} \in \{0, 1\}, \quad n \in \mathcal{N}, l \in \mathcal{V},\ i = 1, \ldots, I. \tag{3e}$$

In (3a), $\eta \in [0, 1]$ is a weighing parameter to control the trade-off between minimizing the average maximum end-to-end latency $\bar{T}_{n,d}$ and the expected loss of its tail. The constraint in (3b) considers half-duplex transmissions by ensuring that, in each time slot i, every IAB-node communicates with up to one of its neighbors by either receiving or transmitting backhaul data. (3c) considers data causality, i.e., only data already stored in the data buffers can be transmitted, and (3d) prevents buffer exhaustion. As the considered scenario is not static, solving (3) would require complete non-causal knowledge of the system dynamics during the complete time horizon I. However, in practical scenarios, knowledge about

the underlying random processes is not available in advance. For example, the BS-node's loads $B_{n,i}$ depend not only on the transmitted and received backhaul data, but also on the randomly arriving data from its users. Similarly, the amounts of transmitted data $M_{n,l,i}$ depend on the varying channel conditions of both BS n and l. As a result, the exact values of $T_{n,l,i}$, $B_{n,i}$ and $M_{n,l,i}$ are not known beforehand. For this reason, we present in Sec. IV Safehaul, a multi-agent learning approach to minimize in each BS-node the average maximum end-to-end latency and the expected value of the tail of its loss.

IV. Our Proposed Solution: Safehaul

In this section, we describe Safehaul, a multi-agent learning approach for the joint minimization of the average maximum end-to-end latency and its expected tail loss in IAB mmWave networks. In Safehaul, each BS-node independently decides which links (n,l) to activate in every time slot i by leveraging a multi-armed bandit formulation. The consensus among the BS-nodes is reached by exploiting the centralized resource coordination and topology management role of IAB-donors [1, Sec. 4.7.1].

A. Multi-Armed Bandit Formulation

Multi-armed bandit is a tool well suited to problems in which an agent makes sequential decisions in an unknown environment [26]. In our scenario, each BS-node n decides, in each time slot i, which of the links (n,l) to activate without requiring prior knowledge about the system dynamics. The multi-armed bandit problem at BS-node n can be characterized by a set $\mathcal{A}_n$ of actions and a set $\mathcal{R}_n$ of possible rewards. The rewards $r_{n,i} \in \mathcal{R}_n$ are obtained in each time slot i as a response to the selected action $a_{n,i} \in \mathcal{A}_n$ and the observed latency. Since every BS-node n selects only one action during each time slot, we enforce the half-duplex constraint in (3b) by defining the set of possible actions as the set of feasible links for BS-node n. In particular, we define $\mathcal{A}_n$ for $n \in \mathcal{N}$ as $\mathcal{A}_n = \{(n,l),(m,n) | m \in \mathcal{N}, l \in \mathcal{V}\}$, where link (n,n) indicates that BS-node n remains idle. As blockages, overloads, or failures might render certain links (n,l) temporarily unavailable, we define the set $\mathcal{A}_{n,i} \subseteq \mathcal{A}_n$ of available actions in time slot i as $\mathcal{A}_{n,i} = \{(n,l),(l,n) | (n,l),(l,n) \in \mathcal{E}_i\}$. Selecting action $a_i = (n,l)$ in time slot i implies $x_{n,l,i} = 1$.

The rewards $r_{n,i}$ are a function of the end-to-end latencies $T_{n,d,i}$ and depend on whether at BS-node n a link (n,l) or (l,n) is activated. BS-node n is connected to the BS-donor via multi-hop wireless links. Consequently, $T_{n,d,i}$ cannot be immediately observed when a link (n,l), with $l \notin \mathcal{D}$ is activated. In fact, the destination BS-donor d might not even be known to BS-node n in time slot i. To overcome this limitation, we define the rewards $r_{n,i}$ as a function of the next-hop's estimated end-to-end latency $\hat{T}_{l,d,i}$ as

$$r_{n,i} = \begin{cases} t^{\mathrm{q}}_{l,i} + t^{\mathrm{tx}}_{n,l,i} + \hat{T}_{l,d,i}, & \text{for link } (n,l) \\ t^{\mathrm{q}}_{n,i} + \hat{T}_{n,d,i}, & \text{for link } (l,n), \end{cases} \tag{4}$$

where $\hat{T}_{l,d,i}$ is calculated as $\hat{T}_{l,d,i} = \min_{(l,m)\in\mathcal{E}_i} \hat{T}_{l,m,i}$ and $t^{\mathrm{tx}}_{n,l,i}$ is calculated based on $M_{n,l,i}$ to ensure the causality constraint in (3c) is fulfilled. Note that the constraint in (3d) cannot be enforced, since multi-armed bandit algorithms learn from the activation of both optimal and suboptimal links.

B. Latency and CVaR Estimation

As given in (4), BS-node n learns which links (n,l) to activate by building estimates of the expected latency $\hat{T}_{n,l}$ associated to each of them. Let $K_{n,l,i} = \sum_{j=1}^{i} x_{n,l,i}$ be the number of times link (n,l) has been activated up to time slot i. The estimated $\hat{T}_{n,l}$ is updated using the sample mean as

$$\hat{T}_{n,l,i+1} = \frac{K_{n,l,i}\hat{T}_{n,l,i} + r_{n,i}}{K_{n,l,i}+1}, \tag{5}$$

where the subindex i is introduced to emphasize that the estimate is built over time.

The CVaR definition given in (2) requires $T_{n,d}$ which, as discussed before, is not known a priori. Hence, we leverage the CVaR estimator derived in [27] to calculate the estimated CVaR of a link (n,l). Let $\tilde{r}^1_n, \ldots, \tilde{r}^{K_{n,l,i}}_n$ be all the rewards received up to time i. The estimated $\widehat{\mathrm{CVaR}}_i(n,l)$ in time slot i is calculated as [27]

$$\widehat{\mathrm{CVaR}}_i(n,l) := \inf_{t\in\mathbb{R}} \left(t + \frac{1}{\alpha \cdot K_{n,l,i}} \sum_{k=1}^{K_{n,l,i}} [\tilde{r}^k_n - t]^+ \right). \tag{6}$$

Using the estimates in (5) and (6), BS-node n computes the value $Q_n(a_{n,i} = (n,l))$ associated to the selected action $a_n \in \mathcal{A}_n$, and defined as

$$Q_n(a_{n,i}) = \hat{T}_{n.l,i} + \eta \widehat{\mathrm{CVaR}}_i(n,l). \tag{7}$$

Note that (7) is aligned with the objective function in (3a). Actions with an associated low value $Q_n(a_{n,i})$ lead to lower end-to-end latency and a low expected value on its tail.

C. Consensus

All the BS-nodes independently decide which links to activate based on their estimates of the end-to-end latency. As a consequence, conflicting actions may be encountered. A conflict occurs when two or more BS-nodes n and m aim at activating a link to a common BS l, $l \in \mathcal{V}$, i.e., $x_{n,l,i} = x_{m,l,i} = 1$. We reach consensus by first retrieving the buffer and congestion status of the various IAB-nodes, leveraging the related BAP layer functionality [1, Sec. 4.7.3]. With this information at hand, conflicts are resolved by prioritizing the transmission of the BS-node with the larger queuing times $t^{\mathrm{q}}_{n,i}$ and loads $B_{n,i}$. Then, we let the IAB-donor mark as *unavailable* the time resources of the remaining base stations with conflicting scheduling decisions [1, Sec. 10.9]. Note that as the learning is performed at each BS-node, only the link activation decision and the weighted sum of $t^{\mathrm{q}}_{n,i}$ and $B_{n,i}$ are transmitted. Thus, low communication overhead is achieved.

D. Implementation of Safehaul

Here, we describe how the above-mentioned solution can be implemented in a real system. Specifically, we elaborate on the required inputs and the interactions among the different entities as well as the pseudo-code of Safehaul, see Alg. 1.

Algorithm 1 Safehaul Algorithm at Each BS-node

Input: α, η, $\mathcal{A}_n$

1: Initialize $\hat{T}_{n,l}$, $\widehat{\mathrm{CVaR}}(n,l)$, and Q_n for all $(n,l) \in \mathcal{E}_1$
2: Set counters $K_{n,l} = 0$ and initial action $a_{n,1} = (n,n)$
3: **for** every time slot $i = 1, \ldots, I$ **do**
4: perform action $a_{n,i}$, observe reward $r_{n,i}$ and increase counter $K_{n,l}$ by one ▷ Eq. (4)
5: update latency estimate $\hat{T}_{n,l}$ ▷ Eq: (5)
6: update CVaR estimate $\widehat{\mathrm{CVaR}}(n,l)$ ▷ Eq: (6)
7: update $Q_n(a_{n,i})$ ▷ Eq: (7)
8: select next action $a_{n,i+1}$ using ϵ-greedy ▷ Eq. (8)
9: share $a_{n,i+1}$, $t^{\mathrm{q}}_{n,i}$ and $B_{n,i}$ with the other BS-nodes
10: if required, update $a_{n,i+1}$ to reach consensus ▷ Sec. IV-C
11: **end for**

Safehaul is executed at each BS-node n. For its implementation, the MNO provides α, η and $\mathcal{A}_n$ as an input. α is the risk level parameter that influences the level of reliability achieved in the system. Similarly, η controls the impact of the minimization of the latency in the α-worst cases on the overall performance. Both parameters, α and η, are set by the MNO depending on its own reliability requirements. The set $\mathcal{A}_n$ depends on the considered network topology, which is perfectly known by the MNO. $\mathcal{A}_n$ includes all links (n,l) and (l,n) to and from the first-hop neighbors of BS-node n.

The execution of Safehaul begins with the initialization of the latency and CVaR estimates, and the values Q of the actions in $\mathcal{A}_n$. Additionally, the counters $K_{n,l}$, that support the calculations of $\hat{T}_{n,l}$ and $\widehat{\mathrm{CVaR}}(n,l)$, are initialized for all links in $\mathcal{A}_n$ (lines 1-2). These parameters are updated and learnt throughout the execution of Safehaul. At time slot $t = 0$, no transmission has occurred and $B_{n,0} = 0$. Hence, BS-node n remains idle for the first time slot $i = 1$, i.e., $a_{n,1} = (n,n)$ (line 2). Next, and in each of the subsequent time slots $i \in \{1, \ldots, I\}$, the selected action is performed and the corresponding reward is obtained (line 4). If BS-node n transmits in time slot i, i.e., $a_{n,i} = (n,l)$, the reward $r_{n,i}$ is sent by the receiving BS l through the control channel. If $a_{n,i} = (l,n)$, the reward $r_{n,i}$ depends, as given in (4), only on the current estimates at BS-node n and the status of its buffer $B_{n,i}$. With the observed reward $r_{n,i}$, the counter for action $a_{n,i}$ is increased and the latency and CVaR estimates are updated (lines 4-6). Using the new estimates (lines 5 and 6), the value $Q(a_{n,i})$ of the performed action $a_{n,i}$ is updated (line 7). The next action $a_{n,i+1}$ is then selected according to ϵ-greedy (line 8), which is a well-known method to balance the exploitation of links with estimated low latency, and the exploration of unknown but potentially better ones. In ϵ-greedy, a random action $a_{n,i+1}$ from the set $\mathcal{A}_{n,i+1}$ is selected with probability $\epsilon \in [0,1]$. With probability $(1-\epsilon)$, instead, the action that yields the estimated lowest value is chosen, i.e.,

$$a_{n,i+1} = \begin{cases} \text{randomly selected action from } \mathcal{A}_{n,i+1}, & \text{if } x \leq \epsilon \\ \underset{b_n \in \mathcal{A}_{n,i+1}}{\operatorname{argmax}} \; Q_n(b_n), & \text{if } x > \epsilon, \end{cases} \tag{8}$$

where x is a sample taken from a uniform distribution in the interval $[0,1]$. Once the action $a_{n,i+1}$ is selected, it is shared with other BS-nodes in the network along with $t^{\mathrm{q}}_{n,i}$ and $B_{n,i}$ (line 9). As described in Section IV-C, this goes through the control channel. If conflicts arise, consensus is reached by prioritizing the transmission of the BS-node with the largest loads and queuing times (line 10).

E. Regret Analysis

The regret ζ is the expected loss caused by the fact that the optimal action is not always selected [28]. Let $\bar{T}^*$ and $\bar{T}_{a_n}$ be the expected delay associated to the optimal action $a^* \in \mathcal{A}_n$ and the non-optimal action $a_n \in \mathcal{A}_n$, respectively. Similarly, let CVaR^* and CVaR_{a_n} be the CVaR of the optimal action $a^* \in \mathcal{A}_n$ and the non-optimal action $a_n \in \mathcal{A}_n$, respectively. Formally, the regret ζ_i after i time slots is defined as

$$\begin{aligned} \zeta_i &= \sum_{a_n \in \mathcal{A}_n} \left(\left(\bar{T}_{a_n} + \eta \mathrm{CVaR}_{a_n} \right) - \left(\bar{T}^* + \eta \mathrm{CVaR}^* \right) \right) \mathbb{E}[K_{a_n,i}] \\ &= \sum_{a_n \in \mathcal{A}_n} \Delta_{a_n} \mathbb{E}[K_{a_n,i}], \end{aligned} \tag{9}$$

where $K_{a_n,i}$ is the number of times action a_n has been selected up to time slot i.

Proposition 1: For a network $\mathcal{G}$ in which the independent decisions of the BS-nodes do not lead to conflicts, let $A_n = |\mathcal{A}_n|$ be the number of available actions for BS-node n. Additionally, let $c > 0$, $0 < d \leq 1$, and $\epsilon_i := \min(1, \frac{cA_n}{d^2 i})$. Then, there exists a positive constant $C > 1$, such that the probability that Safehaul chooses a non-optimal action $a_n \neq a^*$ after $i \geq cA_n/d$ time slots is upper bounded as

$$\begin{aligned} \mathbb{P}[a_{n,i} = a_n] \leq \frac{c}{d^2 i} + \frac{4e}{d^2} B_i^{\frac{c}{2}} + \frac{2Cd^2}{c \ln\left(\frac{(i-1)d^2 e^{0.5}}{cA_n} \right)} \\ + 4C \left(\frac{c}{d^2} \ln\left(\frac{(i-1)d^2 e^{0.5}}{cA_n} \right) \right) B_i^{\frac{c}{5d^2}}, \end{aligned}$$

with $B_i = \frac{cA_n}{(i-1)d^2 e^{0.5}}$.

Proof: See the Appendix. □

Theorem 1: For a network $\mathcal{G}$ in which the independent decisions of the BS-nodes do not lead to conflicts, the regret ζ_i of Safehaul after i time slots is upper bounded by

$$\begin{aligned} \zeta_i \leq \sum_{a_n \in \mathcal{A}_n} \Delta_{a_n} \Bigg(1 + \sum_{i'=2}^{i} \Bigg[\frac{c}{d^2 i'} + \frac{4e}{d^2} B_{i'}^{\frac{c}{2}} + \frac{2Cd^2}{c \ln\left(\frac{(i'-1)d^2 e^{0.5}}{cA_n} \right)} \\ + 4C \left(\frac{c}{d^2} \ln\left(\frac{(i'-1)d^2 e^{0.5}}{cA_n} \right) \right) B_{i'}^{\frac{c}{5d^2}} \Bigg] \Bigg), \end{aligned}$$

where $c > 0$ and $0 < d \leq 1$.

Proof: From the definition in (9), the regret can be upper bounded as

$$\zeta_i \leq \sum_{a_n \in \mathcal{A}_n} \Delta_{a_n} \left(1 + \sum_{i'=2}^{i} \mathbb{P}[a_{n,i'} = a_n] \right), \tag{10}$$

by considering that $\mathbb{E}[K_{a_n,i}] \leq 1 + \sum_{i'=2}^{i} \mathbb{P}[a_{n,i'} = a_n]$. The bound is obtained by including the result of Proposition 1

in (10) as

$$\zeta_i \leq \sum_{a_n \in \mathcal{A}_n} \Delta_{a_n} \left(1 + \sum_{i'=2}^{i} \left[\frac{c}{d^2 i'} + \frac{4e}{d^2} B_{i'}^{\frac{c}{2}} + \frac{2Cd^2}{c \ln\left(\frac{(i'-1)d^2 e^{0.5}}{cA_n} \right)} + 4C \left(\frac{c}{d^2} \ln\left(\frac{(i'-1)d^2 e^{0.5}}{cA_n} \right) \right) B_{i'}^{\frac{c}{5d^2}} \right] \right), \quad (11)$$

As every term in square brackets decreases monotonically in i', the regret ζ_i grows sub-linearly. □

V. Simulation Setup

Given the lack of access to actual 5G (and beyond) network deployments, prior works mostly rely on *home-grown* simulators for performance evaluation. Although this is a valid approach, these simulators often cannot fully capture the real network dynamics, introducing strong assumptions in the physical and/or the upper layers of the protocol stack. Until very recently, the most complete simulator for IAB networks was a system-level simulator [29] developed as an extension of the ns-3 *mmWave* module [30]. However, despite accurate modeling of the IAB protocol stack, it is currently behind the latest IAB specifications.[2] Moreover, the ns-3 IAB extension is unsuitable for large simulations with hundreds of nodes due to reliance on an older version of the *mmWave* module. Therefore, in our work we opt for Sionna [21], which is an open-source GPU-accelerated toolkit based on TensorFlow.

However, unlike the aforementioned ns-3 module, Sionna is a physical layer-focused simulator that does not explicitly model 5G networks, thus lacking the characterization of the 5G-NR upper-layer protocol stack. Hence, we extend Sionna by including the system-level functionalities such as MAC-level scheduling and RLC-level buffering. Furthermore, since Sionna exhibits slight differences compared to the 5G-NR physical layer, we extend Sionna's physical layer model [21] with the 5G-NR procedures. In the following, we describe the details of our extensions, which are publicly available.[3]

A. Extensions to Sionna's Physical Layer Module

In this section, we describe the physical layer modification that were necessary to evaluate IAB scenarios using Sionna.

1) Codebook-Based Beamforming: Sionna's native beamforming only supports Zero-Forcing (ZF) pre-coding in downlink. Therefore, as a first step, we extend Sionna by implementing an NR-like codebook-based analog beamforming both at the transmitter and at the receiver. Specifically, we assume that the beamforming vectors at the transmitter w_{tx} and at the receiver w_{rx} are a pair of codewords selected from a predefined codebook. The codebook is computed by defining a set of beam directions $\{\omega_{p,q}\}$ which scans a given angular sector with a fixed beamwidth. The steering vector $a_{p,q}$ corresponding to direction $\omega_{p,q}$ can be computed as:

$$a_{p,q} = \Big[1, \ldots, e^{j\frac{2\pi}{\lambda} d(i_H \sin\alpha_p \sin\beta_q + i_V \cos\beta_q)}, \ldots, e^{j\frac{2\pi}{\lambda} d((N_H-1)\sin\alpha_p \sin\beta_q + (N_V-1)\cos\beta_q)}\Big]^T, \quad (12)$$

where N_H and N_V are the number of horizontal and vertical antenna elements, respectively. The horizontal and vertical indices of a radiating element are denoted by $i_H \in \{0, \ldots, N_H - 1\}$ and $i_V \in \{0, \ldots, N_V - 1\}$, respectively. α_p and β_q represent the azimuth and elevation angles of $\omega_{p,q}$. Next, we define the codebook as the set $\{\left(\sqrt{N_H N_V}\right)^{-1} w_{p,q}\}$.

In line with the 5G-NR beam management procedure [31], we assume the lack of complete channel knowledge, i.e., the communication endpoints do not know the corresponding channel matrix. Accordingly, an exhaustive search is conducted to identify the best pair of codewords resulting in the highest Signal to Interference plus Noise Ratio (SINR). We leverage a hierarchical search, in which the communication pairs first perform a wide-beam search in which the transmitter and the receiver approximate the direction of communication, see Fig. 2. Next, the beamforming direction is fine-tuned through a beam refinement procedure going through a codebook with narrow beams. Consequently, we employ two types of codebooks, one with wide beams for sector sweep and another with narrow beams for beam refinement.

2) SINR Computations: Since Sionna does not natively calculate the SINR, we add this functionality to the simulator to better model the impact of interference in our simulations. We compute the SINR experienced by Transport Blocks (TBs) by combining the power of the intended signal with that of

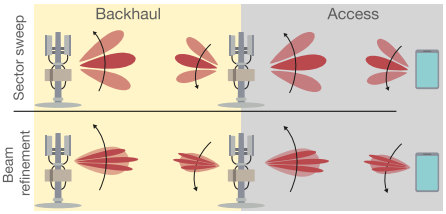


Fig. 2. Schematic of the hierarchical beam management procedure. First, the general direction is estimated using wide beams (top). Then, the search is refined using the narrow beams codebook.

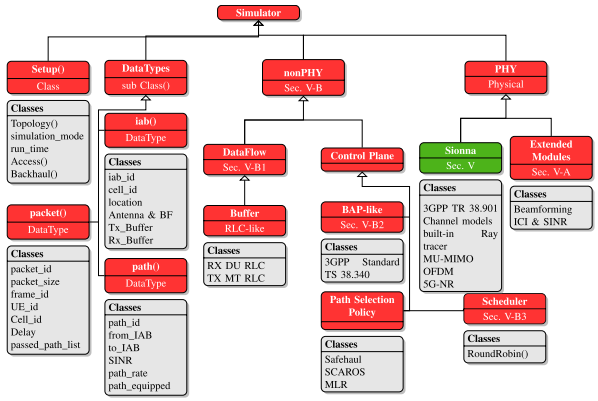


Fig. 3. Overall design of our Sionna's extension. The red blocks represent our additions to the baseline simulator, i.e., Sionna [21].

[2]For instance due to the assumption of L-3 (instead of L-2) relaying at the IAB-nodes which was based on a draft version of TR 38.874 [23].

[3]https://github.com/TUDA-wise/safehaul_infocom2023

the interferers and of the thermal noise. Specifically, we first compute the power $P_n(i,f)$ of the intended signal at receiver n over frequency f and in time slot i. Then, we obtain the overall interference power by leveraging the superposition principle and summing the received power from all other interfering base stations $P_m(i,f)$ where $m \neq n$. For the purposes of this computation, we assume that each interferer employs the beamforming vector yielding the highest Signal to Noise Ratio (SNR) towards its intended destination. Similarly, the transmitter and the receiver use the beamforming configuration estimated via the hierarchical search procedure. Finally, the SINR is $\gamma_n(i,f) = \frac{P_n(i,f)}{\sum_{m \neq n} P_m(i,f) + \sigma^2(i,f)}$ where $\sigma^2(i,f)$ is the thermal noise power at the receiver.

B. System-Level Extensions to Sionna

As mentioned, Sionna is mainly a physical layer simulator. However, to get closer to IAB networks as specified in Rel. 17, we have extended Sionna by implementing a selection of system-level features. To such end, we introduced a discrete-event network simulator for modeling IAB networks. This system-level extension operates on top of Sionna and provides basic functionalities such as a Medium Access Control (MAC)-level scheduler, layer-2 buffers, and data flow and path selection mechanisms. Our simulator, depicted in Fig. 2, generates a variety of system-level KPIs such as latency, throughput, and packet drop rate.

1) Data Flow and Buffer: 3GPP has opted for a layer-2 relaying architecture for BS-nodes where hop-by-hop Radio Link Control (RLC) channels are established. This enables retransmissions to take place on the affected hops only, thus preventing the need for traversing again the whole route from the BS-donor whenever a physical layer TB cannot be successfully decoded. This design results in a more efficient recovery from transmission failures and reduces buffering at the communication endpoints [32]. To mimic this architecture, we have implemented RLC-like buffers at each base station. Specifically, each BS-node features layer-2 buffers for both received and transmitted packets. For instance, the data flow for an uplink packet is the following. The User Equipment (UE) generates packets and sends a transmission request to the base station. Consequently, the scheduler allocates OFDM symbols for this transmission, which is eventually received and stored at the RX buffer of its Distributed Unit (DU). Next, the packet is placed into the TX buffer to be forwarded to the suitable next hop BS-node. This procedure is repeated until the packet crosses all the wireless-backhaul hops and reaches the BS-donor. Note that the packet can be dropped due to latency constraints or to interference.

2) BAP: To manage routing within the wireless-backhauled network, the 3GPP introduced BAP, i.e., an adaptation layer above RLC which is responsible for packet forwarding between the BS-donor and the access BS-nodes [33]. Our simulator mimics this by associating each BS-node to a unique BAP ID. Moreover, we append a BAP routing ID to each packet at its entry point in the Radio Access Network (RAN) (i.e., the BS-donor and the UEs for DL and UL data, respectively). Then, this identifier is used to discern the (possibly multiple) routes toward the packet's intended destination [33]. The choice of the specific route is managed by Safehaul.

3) Scheduler: We implemented a MAC-level scheduler which operates in a Time Division Multiple Access (TDMA) mode. The scheduler periodically allocates the time resources to backhaul or access transmissions in a Round-Robin fashion.[4] Specifically, each cell first estimates the number of OFDM symbols needed by each data flow by examining the corresponding buffer. Then, the subframe's OFDM symbols are equally allocated to the users. If a user requires fewer symbols to transmit its complete buffer, the excess symbols (the difference between the available slot length and the needed slot length) are distributed to the other active users.

TABLE I
SIMULATION PARAMETERS

Parameter	Value
Carrier frequency and bandwidth	28 GHz and 400 MHz
IAB RF chains	2 (1 access + 1 backhaul)
Pathloss model	UMi-Street Canyon [34]
Number of BS-nodes N	{223 NY, 100 Padova}
Source rate	{40, 80} Mbps
IAB Backhaul and access antenna array	8H×8V and 4H×4V
UE antenna array	4H×4V
IAB and UE height	15 m and 1.5 m
IAB antenna gain	33 dB
Noise Figure	10 dB
Risk level α	0.1
Reliability weight factor η	1

VI. PERFORMANCE EVALUATION

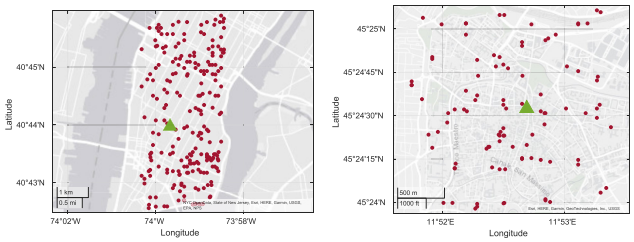
In our simulations, we consider realistic cellular base station deployments in Manhattan, New York City[5] and in the historical city center of Padova. Specifically, for the former we collect the locations of $N = 223$ 5G-NR base stations in an area of 15 Km^2 as depicted in Fig. 4b. On the other hand, in the Padova topology we combine locations of $N = 100$ 4G-LTE Base Station (BS) of different MNOs (WINDTRE, TIM, and Vodafone) in an area of 10 Km^2 as depicted in Fig. 4, due to the lack of 5G-NR base station deployment at the time of writing of this paper. The detailed simulation parameters are provided in Table I. We used the channel model outlined by 3GPP in TR 38.901 [34], which provides a statistical channel model for 0.5-100 GHz, and analyzed the "Urban Micro (UMi)-StreetCanyon" scenario.

Benchmarks. To provide better insights on the performance of Safehaul, we replicate two approaches from the state of the art: (i) Scalable and Robust Self-backhauling Solution (SCAROS), a learning-based approach that minimizes the average latency in the network [17], and (ii) Maximum Local Rate (MLR), a greedy approach aiming to maximize throughput by selecting the links with the highest data rate.

Our evaluations consider six scenario to study the algorithms' convergence to a steady state, the number of BS-nodes, the number of BS-donors, and the impact of risk aversion.

[4]The choice of the specific scheduling algorithm is outside of the scope of the 3GPP NR specifications, and is thus left to the MNOs. Accordingly, a Round-Robin scheduling policy represents a typical baseline assumption.

[5]The locations correspond to the network of T-Mobile, which has the largest deployment among the MNOs.



(a) Manhattan (223 BS-nodes). (b) Padova (100 BS-nodes).

Fig. 4. Locations of BS-nodes (red dots) and of the BS-donor (green triangle) in the Manhattan (left) and Padova (right) topologies.

When demonstrating the results, we show the average throughput, latency, and packet drop rate per UE. We also show the statistical variance of the obtained results using candlesticks which include the corresponding max, min, mean, and 10 and 90 percentiles.

A. Scenario 1: Average Network Performance

Analyzing the performance of the algorithms as a function of time is crucial to determine the convergence speed of the learning-based techniques, i.e., Safehaul and SCAROS. Hence, in Fig. 5 we show the average network performance over time for three metrics: latency, throughput, and packet drop rate.

In Fig. 5a, we can observe that Safehaul rapidly converges to an average latency of approximately 8.6 ms which is 12.2% and 43.4% lower than the latency of SCAROS and MLR, respectively. The high performance of Safehaul stems from the joint minimization of the average latency and the expected value of its tail loss, which results in avoiding risky situations where latency goes beyond $T_{\max}$. This is not the case for SCAROS where we observe a high peak in the latency before convergence, i.e., between zero and 1000 ms. *It is exactly the avoidance of such transients in Safehaul that leads to higher reliability in the system.* The reliability offered by Safehaul allows MNOs to deploy self-backhauling in an online fashion and without disrupting the network operation. The performance of MLR is constant throughout the simulation, as it is not designed as an adaptive algorithm.

Figure 5b shows that the risk-aversion capabilities of Safehaul have no negative impact on the average throughput of the network. The performance of Safehaul is comparable to that of SCAROS, approximately 79.3 Mbps, and 11.7% larger than the performance of MLR. The performance shown in Figure 5c is consistent with the behavior observed in Figure 5a. As Safehaul additionally minimizes the α-worst latency, it achieves the lowest packet drop rate compared to the reference schemes, namely, 30.1% (84.0%) lower than SCAROS (MLR).

B. Scenario 2: Impact of the Network Size

In Fig. 6 we evaluate the reliability of the three considered approaches for different network sizes. Specifically, we vary the number of BS-nodes from 25 to 200. At the same time, we increase the load in the network by increasing the number of UEs. From the figures, we can clearly see that Safehaul consistently achieves a lower variation compared to the reference schemes. This verifies that Safehaul achieves the intended optimization goal, i.e., the joint minimization of the average end-to-end delay and its expected tail loss.

Fig. 6a shows that Safehaul is able to maintain an almost constant latency as the number of BS-nodes increases. Specifically, the variation of latency with Safehaul is 56.1% and 71.4% less than with SCAROS and MLR, respectively. Furthermore, Safehaul achieves 11.1% and 43.2% lower latency compared to SCAROS and MLR, where the high variance exhibited by the latter is due to a lack of adaptation capabilities. As shown in Fig. 6b, the average throughput of the learning-based approaches Safehaul and SCAROS remains constant for the different values of the network size. However, the lowest variation in the throughput is achieved by Safehaul, i.e., only 0.90 compared to 1.9 and 2.8 in the benchmark schemes. Such behavior corroborates Safehaul's reliability capabilities. The packet drop rate for different numbers of BS-nodes is shown in Fig. 6c. Safehaul not only consistently outperforms the reference schemes, but also has the minimum variation in the results (at least 47.3% lower compared to the benchmarks). Considering the largest network size and load, i.e., 200 BS-nodes and 400 UEs, Safehaul achieves 49.3% and 81.2% lower packet drop rate compared to SCAROS and MLR, respectively.

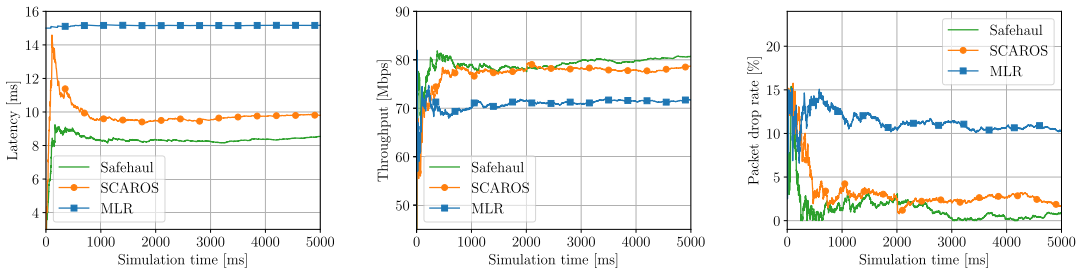
C. Scenario 3: Impact of the Number of BS-donors

Although the benchmark schemes do not support multiple BS-donors, Safehaul is designed to accommodate such scenarios. In Fig. 7, we investigate the impact of the number of BS-nodes on Safehaul. To this end, we keep the number of UEs and their data rate constant.

We observe in Fig. 7a that the highest latency is experienced when only one BS-donor is present in the network. This stems from the tributary effect of self-backhauling where the traffic flows towards a central entity which itself can become a bottleneck. As the number of BS-donors increases, the traffic is more evenly distributed, resulting in lower latency. Specifically, the average latency decreases from 8.2 ms for $D = 1$ to 1.7 ms when $D = 5$. Since the load is constant in this scenario, the average throughput also remains constant for all different numbers of BS-donors, see Fig. 7b. Notably, Safehaul's learning speed is maintained for the different values of D. This is an important feature because having more BS-donors exponentially increases the number of paths a BS-node has to the core network. From a learning perspective, such increment implies a larger action set and a lower learning speed. Safehaul avoids this problem by learning the average latency based on the estimates of its neighbors and not on the complete paths to the BS-donors. Finally, Fig. 7c shows that a larger number of BS-donors significantly reduces the packet drops, which also stems from a better distribution of traffic flows in the network, as observed in Fig. 7a.

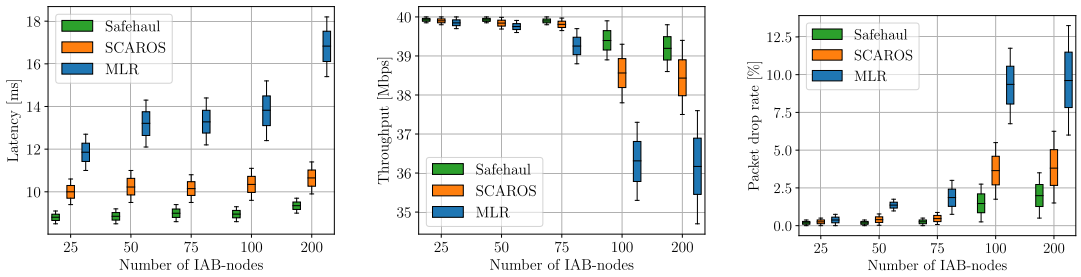
D. Scenario 4: Impact of the Risk Parameter α

The definition of losses in the tail of the latency distribution is controlled by the risk level parameter α. Its impact on the



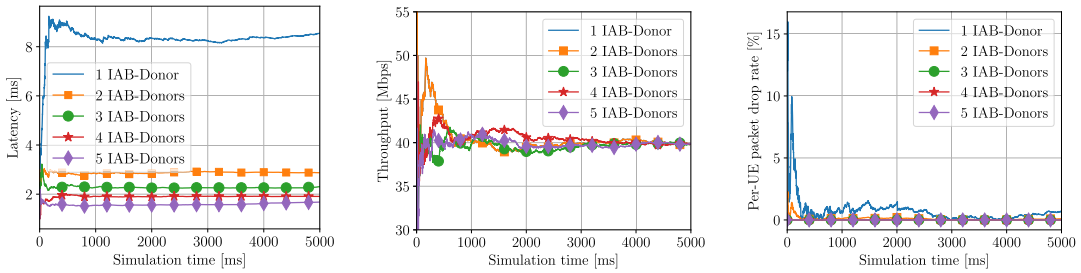
(a) Average per-UE end-to-end latency (b) Average per-UE throughput (c) Average per-UE packet drop rate

Fig. 5. Average network performance for 50 UEs and 80 Mbps per-UE source rate (Scenario 1).



(a) Per-UE end-to-end Latency (b) Per-UE throughput (c) Per-UE packet drop rate

Fig. 6. Network performance for $\{25, 50, 75, 100, 200\}$ BS-node, 2 UEs per BS-nodes on average, and 40 Mbps per-UE source rate (Scenario 2).



(a) Average per-UE end-to-end Latency (b) Average per-UE throughput (c) Average per-UE packet drop rate

Fig. 7. Network performance for 50 UEs and 40 Mbps per-UE source rate, versus the number of BS-donors (Scenario 3).

average latency is shown in Fig. 8, where an increasing behavior is observed for $\alpha \leq 0.7$. The lowest latency is achieved for $\alpha = 0.1$, which corresponds to the most risk-averse, and therefore the most reliable, case out of all the considered ones. The non-monotonic behavior of the average latency versus α can be explained by the so-called exploration-exploitation trade-off: the higher α, the higher the level of risk, which in turn leads Safehaul to learn more about the environment and choose a more reliable action. Eventually, as α grows beyond approximately 0.7, the performance of Safehaul tends to that of the risk-neutral case. As a consequence, the algorithm undertakes excessive exploration, which causes a degradation of the average latency performance.

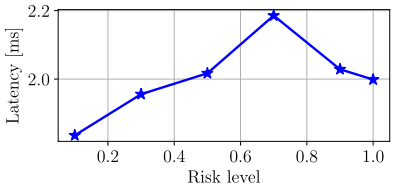
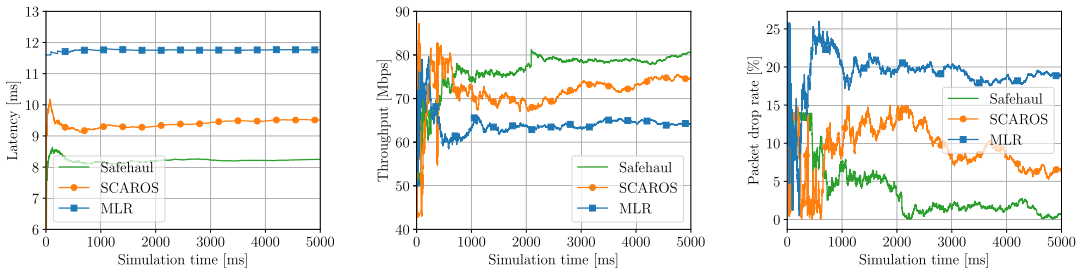


Fig. 8. Average latency for 50 UEs and 20 Mbps per-UE source rate, versus the risk level α (Scenario 4).

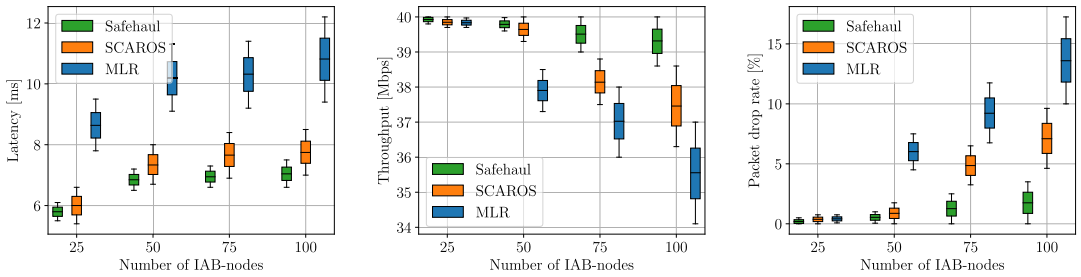
E. Scenario 5: Performance in Different Topologies

To verify the generality of the proposed algorithms, it is essential to examine how they perform in different topologies, and consider both typical network performance metrics (i.e., along the lines of Scenario 1) and their stability with respect to the number of BS-nodes and BS-donors (Scenarios 2 and 3). To this end, we ran additional simulations in the deployment depicted in Fig. 4b, which mimics the BS-nodes locations of the historic center of Padova. We report the average network performance over time, in terms of end-to-end packet drop rate, throughput, and latency in Fig. 9. Overall, the outcomes of this simulation campaign are in line with those obtained in Scenario 1. Specifically, as seen in Fig. 9a, Safehaul quickly converges to an average latency of approximately 8 ms, which is 14% and 31% lower than



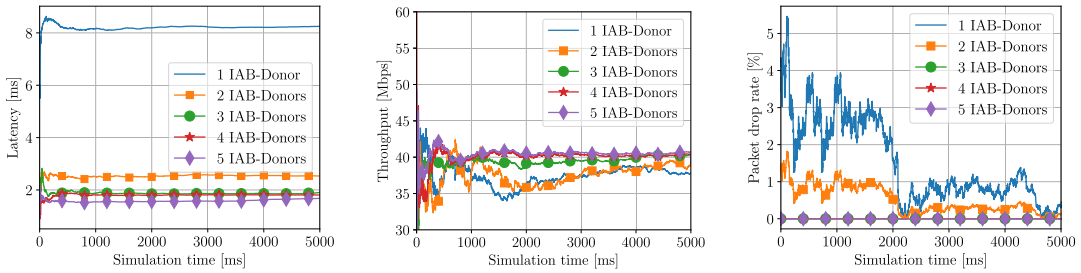
(a) Average per-UE end-to-end latency (b) Average per-UE throughput (c) Average per-UE packet drop rate

Fig. 9. Average network performance for 50 UEs and 80 Mbps per-UE source rate (Scenario 1) in Padova.



(a) Per-UE end-to-end Latency (b) Per-UE throughput (c) Per-UE packet drop rate

Fig. 10. Network performance for $\{25, 50, 75, 100\}$ BS-nodes,2 UEs per BS-node on average, and 40 Mbps per-UE source rate (Scenario 2) in Padova.



(a) Average per-UE end-to-end Latency (b) Average per-UE throughput (c) Average per-UE packet drop rate

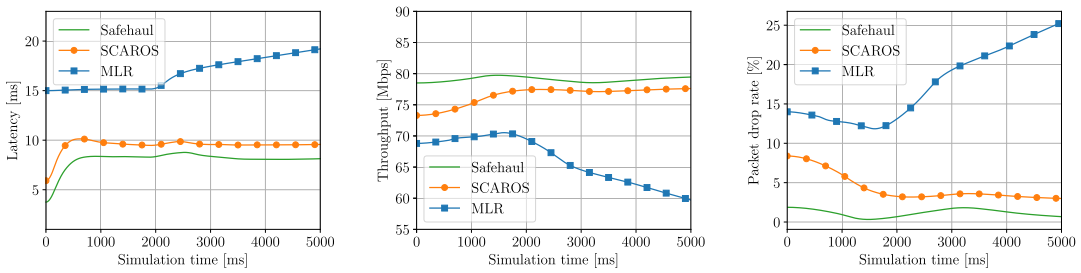
Fig. 11. Network performance for 50 UEs and 40 Mbps per-UE source rate, versus the number of BS-donors in Padova (Scenario 3).

SCAROS and MLR's latency. Fig. 9b shows the average per-UE throughput, for which Safehaul achieves about 4% and 17% better performance than SCAROS and MLR, respectively. Similarly, the performance depicted in Fig. 9c is in line with that reported in Figs. 9a and 9b, with Safehaul achieving approximately a 24% and 38% smaller packet drop rate than SCAROS and MLR, respectively.

In Fig. 10, we compare the consistency of the performance of the three algorithms with respect to the network size. In particular, we change the number of BS-nodes from 25 to 100, keeping fixed the number of UEs per BS-node and thus effectively increasing the network load on the BS-donor. Results show that Safehaul, when compared to other schemes, exhibits minimal performance degradation when introducing additional BS-nodes and UEs. As can be seen in Fig. 10a, the latency achieved by Safehaul increases by at most 16% in the case of 100 BS-nodes, while SCAROS and MLR lead to a latency which is consistently higher and increases up to 27% and 25% when deploying additional nodes, respectively. Similar trends can be observed in Figs. 10b and 10c, which report throughput and packet loss versus the network size, respectively. Indeed, Safehaul is the best performer across the whole range of BS-nodes which have been considered. Furthermore, Safehaul loses 20% more packets with the denser network deployment (i.e., 100 BS-nodes), while reference schemes exhibit an increase in packet loss of up to 33%.

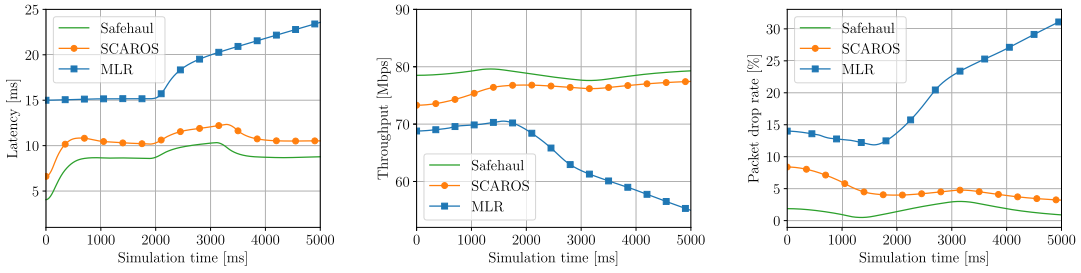
We complete this analysis by examining how the number of donors affects the performance achieved by Safehaul in the Padova-like topology. As can be seen in Fig. 11, increasing the number of fiber-backhauled base stations progressively reduces the latency. Similarly, and in line with the results obtained in Scenario 3 and reported in Fig. 11c, the packet drop rate varies from approximately 0.08% when considering a single BS-donor, to approximately 0.003% in the presence of five BS-donors. The performance improvements introduced by additional fiber links saturate after 3 donors, thanks to the efficient routing and scheduling performed by Safehaul.

In summary, the results obtained in the additional topology mimicking the historical center of Padova are well aligned with those obtained in the Manhattan topology. Although



(a) Average per-UE end-to-end latency (b) Average per-UE throughput (c) Average per-UE packet drop rate

Fig. 12. Average network performance for 50 UEs and 80 Mbps per-UE source rate where 1 random BS-node is shut down.



(a) Average per-UE end-to-end latency (b) Average per-UE throughput (c) Average per-UE packet drop rate

Fig. 13. Average network performance for 50 UEs and 80 Mbps per-UE source rate where 3 random BS-nodes are shut down.

the specific values of the network metrics achieved by the considered schemes in the two topologies are different (for instance, SCAROS achieves a 66% lower packet drop rate in Scenario 1 compared to Scenario 5), the trends among the various schemes are the same. Specifically, we observed that Safehaul consistently achieves the best performance in comparison to SCAROS and MLR across different metrics, which supports the claim that the proposed scheduler is capable of learning how to optimize arbitrary deployment topologies.

F. Scenario 6: Network Resilience

In networking, resilience refers to the ability of a network to recover in a quick and effective fashion from disruptions, thus providing reliable and high-quality communication services to its users. Specifically, the ability to recover from link failures is particularly important in IAB networks, where backhaul links are susceptible to the typical disruptions which plague the RAN due to its mobile and wireless nature. For instance, the links among BS-nodes can be degraded by adverse environmental conditions such as heavy rain and monsoons, physical obstacles and network congestion. These disruptions can cause temporary or permanent communication failures, which in turn result in degraded performance and/or loss of connectivity for the end users. To prevent and/or recover from these undesired events, a backhaul scheduler must detect, mitigate, and recover from various types of disruptions and failures, and must maintain the required level of service availability and performance despite the time-varying channel conditions.

We benchmark the resilience of the proposed algorithm by mimicking radio link failures, which we simulate by stopping BS-nodes at a fixed time instant (2000 s), and inspecting the resulting performance degradation. Since the failed node(s) is (are) chosen at random, we run multiple simulations to estimate the average network performance, as shown in Figs. 12 and 13 for the case of one and three link failures, respectively.

Results show that MLR is unable to react to the link failure(s) due to its static and myopic policy. Specifically, the disruption causes an increase of 33% (60%) in latency, and a decrease of up to 15% (23%) in throughput when considering one (three) link failure(s). On the other hand, both Safehaul and SCAROS are capable of adapting the scheduling to the new topology. Indeed, both schemes show a transient region where the performance is slightly degraded since the algorithms are learning new routes and resource partitions to account for the lost link. Nevertheless, Safehaul and SCAROS eventually converge to a solution which provides approximately the same network performance as before the failures, in both cases of one and three lost links.

VII. Related Work

Self-backhauling wireless networks have been studied in different contexts. Ranging from the so-called Heterogeneous Networks (HetNets) and IAB 5G New Radio (NR) systems, to Cloud Radio Access Networks (C-RANs), each has considered a different set of premises and optimization goals. In this section, we review the related work in the context of basic assumptions and their optimization goals.

Ideal backhaul links. Numerous works assume either an *infinite or fixed capacity backhaul link*. This is often motivated by the presence of a wired fiber link between the Small Base Stations (SBSs) and the Macro Base Station (MBS) [4], [6], [7], [8]. Indeed, most of these works consider a scenario where a centralized Baseband Unit (BBU) is connected to several Remote Radio Heads (RRHs), i.e., radios which lack signal processing capabilities [4], [6], [7]. In particular, the

authors of [7] consider an even more complex C-RAN scenario where RRHs feature caching and signal processing capabilities. However, in an IAB context it is fundamental to consider *limited-rate, time-varying backhaul channels* and to study the impact of such limitations on the performance of the RAN.

Constrained topologies. It is often assumed that self-backhauled networks have a *specific topology*. This assumption usually simplifies the problem and makes it tractable and/or solvable in polynomial time. For instance, the authors of [9], [10], and [13] assume a single-hop network where each SBS is directly connected to the MBS. In [11], a k-ring deployment is considered, i.e., a topology where a single IAB-donor provides backhaul connectivity to k rings of IAB-nodes. Even though this topology can be used to model networks with arbitrary depth, it maintains a symmetric load for each node, an assumption which generally does not hold in real networks. In fact, the 3GPP does not impose any limits on the number of IAB-nodes which can be connected to a given IAB-donor, nor does it set an upper bound on the number of wireless hops from the latter to other wireless-backhauled base stations [23]. Accordingly, in our problem formulation we consider IAB networks with an *arbitrary number of nodes and an arbitrary maximum number of wireless hops* between MBSs and SBSs.

Simplistic traffic models. Some works either assume a *full buffer traffic model and/or impose flow conservation constraints*. In particular, the authors of [8] and [35] consider systems where the capacity of each link can always be fully exploited thanks to the presence of *infinite data to transmit at each node*. However, in actual IAB deployments the presence of packets at the MBSs and SBSs is conditioned on the *status of their RLC buffers and, in turn, on the previous scheduling decisions*. Moreover, *packets can actually be buffered at the intermediate nodes*, thus preventing the need for transmitting a given packet in consecutive time instants along the whole route from the IAB-donor to the UEs (or vice versa).

Optimization goals. The works in the literature focus on different optimization goals. Therefore, they prioritize different network metrics. For instance, the authors of [36] aim to optimize the beam alignment between MBSs and SBSs. Instead, the work of [5] aims to compute the optimal user-to-base-station association. However, they neglect backhaul associations and focus on the access only. In [5], [9], [35], and [37] the objective function is a function of the users data-rate. In particular, the authors of [35] optimize the max-min user throughput, arguing that such a metric better captures the performance of the bottleneck links. In [16], the average rate of each link is maximized under bounded delay constraints. In our work, we focus on reliability by minimizing not only the average end-to-end delay, but also the expected value of the worst-case performance. The work closest to this article is SCAROS [17], a learning-based latency-aware scheme for resource allocation and path selection in self-backhauled networks. Assuming a single IAB-donor, the authors study arbitrary multi-tier IAB networks considering the impact of interference and network dynamics. In contrast, we aim at enhancing the reliability of the IAB-network by jointly minimizing the average end-to-end delay and its expected tail loss.

VIII. Conclusion

In this work, we proposed the first reliability-focused scheduling and path selection algorithm for IAB mmWave networks. We illustrated that our RL-based solution can cope with the network dynamics including channel, interference, and load. Furthermore, we demonstrated that Safehaul not only exhibits highly reliable performance in the presence of the above-mentioned network dynamics, but also outperforms the benchmark schemes in terms of throughput, latency and packet-drop rate. The reliability of Safehaul stems from the joint minimization of the average latency, and the expected value of its tail losses, by leveraging CVaR as a risk metric.

Reliability is a highly under-explored topic that definitely deserves more investigation. Some interesting research directions are the maximization of reliability under the assumption of statistical system knowledge, or the evaluation of the network's reliability when the functionality of the BAP layer is compromised. Furthermore, our system-level extension to Sionna can be further developed to support an arbitrary number of RF chains and in-band backhauling, allowing more extensive investigation of IAB protocols and architectures.

Appendix

For the proof of Proposition 1, Theorem 3 in [27] is needed. For completeness, we first present the theorem in [27] for the special case in which the considered random variables are independent. Next, we present the proof of Proposition 1.

Theorem 2: Let $T_{a_n,i}$ be independent random variables where $\max_{1\leq j\leq i} T_{a_n,j} = T_{\max}$, with $i \in \{1,2,\dots\}$. Then, for any $0 < \delta \leq 1/2$, $\xi > 0$ and $\gamma > 0$, there exists a positive constant C which only depends on ξ and γ, such that the probability of the event $|\widehat{\mathrm{CVaR}}_{a_n,i} - \mathrm{CVaR}_{a_n,i}| \geq 2\xi\alpha^{-1}T_{\max}i^{-\delta}(\ln\ln i)^{1/2}\ln i$ is smaller than or equal to $Ce^{-(1+\gamma)\ln i}$.

Proof: See Theorem 3 in [27]. □

Proof of Proposition 1

Proof: In this proof, we use the result of the regret bound for the risk-neutral case without CVaR, shown in [28, Theorem 3], as a basis. Additionally, we use the bound for the terms related to the CVaR formulated in [27, Theorem 3]. Using both these results, we first bound the probability that Safehaul chooses a suboptimal arm in the exploitation phase. Then, we combine the latter with the probability of choosing a suboptimal arm in the exploration phase to derive the bound given in Proposition 1.

From the system model and Proposition 1, we have that $c > 0$, $0 < d \leq 1$, and $\epsilon_n := \min(1, \frac{cA_n}{d^2 i})$. Moreover, $a_{n,i}$ is the action chosen by ϵ-greedy in time slot i and $K_{a_n,i}$ is the number of times, up to time slot i, in which Safehaul chose action a_n *at random*. Similarly, we use K_i^* for the counter of the optimal action. $T_{a_n,i}$ are independent random variables distributed according to the rewards linked to action a_n. We use T_i^* for the optimal action, and $\hat{T}_{a_n,i}$ is the estimated mean of the probability distribution of the rewards linked to action a_n using $K_{a_n,i}$ samples. As before, we use $\hat{T}_i^*$ for the optimal action. $\widehat{\mathrm{CVaR}}_{a_n,i}$ is the estimated CVaR of

action a_n up to time slot i and $\widehat{\text{CVaR}}_i^*$ is the estimated CVaR of the optimal action up to time slot i. Then, the probability that action a_n is chosen in time slot i is upper bounded as

$$\mathbb{P}[a_{n,i}=a_n] \leq \mathbb{P}\left[\delta_{a_n,i-1} \leq \delta^*_{i-1}\right]\left(1-\frac{\epsilon_i}{A_n}\right)+\frac{\epsilon_i}{A_n}, \quad (13)$$

with $\delta_{a_n,i-1}=\hat{T}_{a_n,i-1}+\eta\widehat{\text{CVaR}}_{a_n,i-1}$ and $\delta^*_{i-1}=\hat{T}^*_{i-1}+\eta\widehat{\text{CVaR}}^*_{i-1}$. The first term in (13) is the probability of exploitation and the second term to the probability of exploration. Using the mean $\bar{T}_{a_n}$ and CVaR_{a_n} of action a_n, and the likewise defined $\bar{T}^*$ and CVaR^* for the optimal action, we set $\Delta^{\text{mean}}_{a_n}:=\bar{T}_{a_n}-\bar{T}^*$ and $\Delta^{\text{cvar}}_{a_n}:=\text{CVaR}_{a_n}-\text{CVaR}^*$. Using these definitions in (13) we conclude

$$\begin{aligned}
&\mathbb{P}\left[\delta_{a_n,i-1}\leq\delta^*_{i-1}\right]\leq\\
&\mathbb{P}\left[\delta_{a_n,i-1}\leq\eta\text{CVaR}_{a_n}-\frac{\Delta^{\text{mean}}_{a_n}}{2}+\bar{T}_{a_n}-\eta\frac{\Delta^{\text{cvar}}_{a_n}}{2}\right]\\
&+\mathbb{P}\left[\bar{T}^*+\frac{\Delta^{\text{mean}}_{a_n}}{2}+\eta\text{CVaR}^*+\eta\frac{\Delta^{\text{cvar}}_{a_n}}{2}\leq\delta^*_{i-1}\right]\\
&\mathbb{P}\left[\hat{T}_{a_n,i-1}\leq\bar{T}_{a_n}-\frac{\Delta^{\text{mean}}_{a_n}}{2}\right]+\mathbb{P}\left[\bar{T}^*+\frac{\Delta^{\text{mean}}_{a_n}}{2}\leq\hat{T}^*_{i-1}\right]\\
&+\mathbb{P}\left[\widehat{\text{CVaR}}_{a_n,i-1}\leq\text{CVaR}_{a_n}-\frac{\Delta^{\text{cvar}}_{a_n}}{2}\right]\\
&+\mathbb{P}\left[\text{CVaR}^*+\frac{\Delta^{\text{cvar}}_{a_n}}{2}\leq\widehat{\text{CVaR}}^*_{i-1}\right]. \quad (14)
\end{aligned}$$

Similar to [28], we use the Chernoff-Hoeffding bound for the first two terms in (14). For the last two summands, there remains to find a bound for the difference between the CVaR and its estimate $\widehat{\text{CVaR}}$. From Theorem 2, we set $\xi:=\Delta^{\text{cvar}}_{a_n}\alpha/4T_{max}$, $\delta=0.5$ and by using the limit $\gamma\to 0$, we obtain

$$\mathbb{P}\left[|\widehat{\text{CVaR}}_{a_n,i}-\text{CVaR}_{a_n,i}|\geq\frac{\Delta^{\text{cvar}}_{a_n}}{2}i^{-0.5}(\ln\ln i)^{0.5}\ln i\right]\leq\frac{C}{i}. \quad (15)$$

As $\max_i i^{-0.5}(\ln\ln i)^{0.5}\ln i < 1$, the condition $(\Delta^{\text{cvar}}_{a_n}/2)i^{-0.5}(\ln\ln i)^{0.5}\ln i \leq \frac{\Delta^{\text{cvar}}_{a_n}}{2}$ holds for all i. Therefore, considering the last two summands in (14), we conclude that there exists a positive constant C that satisfies

$$\mathbb{P}\left[|\widehat{\text{CVaR}}_{a_n,i}-\text{CVaR}_{a_n,i}|\geq\frac{\Delta^{\text{cvar}}_{a_n}}{2}\right]\leq\frac{C}{i}. \quad (16)$$

The number of times action a_n has been selected up to time slot i is smaller than or equal to i, i.e., $K_{a_n,i}\leq i$. Using (16) we write the last two summands in (14) as

$$\mathbb{P}\left[\widehat{\text{CVaR}}_{a_n,i-1}\leq\text{CVaR}_{a_n}-\frac{\Delta^{\text{cvar}}_{a_n}}{2}\right]\leq\frac{C}{K_{a_n,i-1}}, \quad (17)$$

and

$$\mathbb{P}\left[\text{CVaR}^*+\frac{\Delta^{\text{cvar}}_{a_n}}{2}\leq\widehat{\text{CVaR}}^*_{i-1}\right]\leq\frac{C}{K^*_{i-1}}. \quad (18)$$

As in [28], we use Bernstein's inequality to get an estimate for $K_{a_n,i-1}$. Defining $x_0:=1/2A_n\sum_{j=1}^{i-1}\epsilon_j$ for $i-1\geq\frac{cA_n}{d^2}$ we get $P(K_{a_n,i-1}\leq x_0)\leq e^{-\frac{x_0}{5}}$. Additionally, from [28]:

$$x_0\geq\frac{c}{d^2}\ln\left(\frac{(i-1)d^2e^{0.5}}{cA_n}\right)=:C'(i). \quad (19)$$

The same holds for the optimal action and K^*_{i-1}. Using these estimations for x_0, we can conclude that for $i-1\geq cA_n/d^2$

$$\begin{aligned}
&\mathbb{P}\left[\widehat{\text{CVaR}}_{a_n,i-1}\leq\text{CVaR}_{a_n}-\frac{\Delta^{\text{cvar}}_{a_n}}{2}\right] \quad (20)\\
&\leq\sum_{j=1}^{i-1}\mathbb{P}[K_{a_n,i-1}=j]\frac{C}{j}\\
&=\sum_{j=1}^{\lfloor x_0\rfloor}\mathbb{P}[K_{a_n,i-1}=j]\frac{C}{j}+\sum_{j=\lfloor x_0\rfloor+1}^{i-1}\mathbb{P}[K_{a_n,i-1}=j]\frac{C}{j}\\
&\leq Cx_0e^{-\frac{x_0}{5}}+\frac{C}{x_0}\leq Cx_0e^{-\frac{x_0}{5}}+\frac{C}{C'(i)}. \quad (21)
\end{aligned}$$

The same holds again for the optimal action

$$\mathbb{P}\left[\text{CVaR}^*+\frac{\Delta^{\text{cvar}}_{a_n}}{2}\leq\widehat{\text{CVaR}}^*_{i-1}\right]\leq Cx_0e^{-\frac{x_0}{5}}+\frac{C}{C'(i)}. \quad (22)$$

Together with the bounds from Theorem 3 in [28] it follows that for $C\geq 1$:

$$\begin{aligned}
&\mathbb{P}\left[a_{n,i}=a_n\right]\\
&\leq\frac{\epsilon_i}{A_n}+4Cx_0e^{-\frac{x_0}{5}}+\frac{4}{\left(\Delta^{\text{mean}}_{a_n}\right)^2}e^{\frac{-\left(\Delta^{\text{mean}}_{a_n}\right)^2\lfloor x_0\rfloor}{2}}+2\frac{C}{C'(n)}\\
&\leq\frac{c}{d^2i}+2\frac{Cd^2}{c\ln\left(\frac{(i-1)d^2e^{0.5}}{cA_n}\right)}+\frac{4e}{d^2}\left(\frac{cA_n}{(i-1)d^2e^{0.5}}\right)^{\frac{c}{2}}\\
&+\ 4C\frac{c}{d^2}\ln\left(\frac{(i-1)d^2e^{0.5}}{cA_n}\right)\left(\frac{cA_n}{(i-1)d^2e^{0.5}}\right)^{\frac{c}{5d^2}}.
\end{aligned}$$

□

REFERENCES

[1] *NR; Overall Description; Stage-2*, document TS 38.300, Version 17.1.0, 3GPP, Jun. 2022.

[2] *NR; Integrated Access and Backhaul (IAB) Radio Transmission and Reception*, document TS 38.174, V.17.1.0, 3GPP, Jun. 2022.

[3] A. A. Gargari et al., "Safehaul: Risk-averse learning for reliable mmWave self-backhauling in 6G networks," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, May 2023, pp. 1–10.

[4] C. Pan, H. Zhu, N. J. Gomes, and J. Wang, "Joint precoding and RRH selection for user-centric green MIMO C-RAN," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 2891–2906, May 2017.

[5] A. Alizadeh and M. Vu, "Load balancing user association in millimeter wave MIMO networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 2932–2945, Jun. 2019.

[6] X. Huang, G. Xue, R. Yu, and S. Leng, "Joint scheduling and beamforming coordination in cloud radio access networks with QoS guarantees," *IEEE Trans. Veh. Technol.*, vol. 65, no. 7, pp. 5449–5460, Jul. 2016.

[7] H. T. Nguyen, H. D. Tuan, T. Q. Duong, H. V. Poor, and W.-J. Hwang, "Nonsmooth optimization algorithms for multicast beamforming in content-centric fog radio access networks," *IEEE Trans. Signal Process.*, vol. 68, pp. 1455–1469, 2020.

[8] M. E. Rasekh, D. Guo, and U. Madhow, "Interference-aware routing and spectrum allocation for millimeter wave backhaul in urban picocells," in *Proc. 53rd Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2015, pp. 1–7.

[9] G. Kwon and H. Park, "Joint user association and beamforming design for millimeter wave UDN with wireless backhaul," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 12, pp. 2653–2668, Dec. 2019.

[10] A. Pizzo and L. Sanguinetti, "Optimal design of energy-efficient millimeter wave hybrid transceivers for wireless backhaul," in *Proc. 15th Int. Symp. Model. Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, May 2017, pp. 1–8.

[11] M. N. Kulkarni, A. Ghosh, and J. G. Andrews, "Max-min rates in self-backhauled millimeter wave cellular networks," 2018, *arXiv:1805.01040*.

[12] L. F. Abanto-Leon, A. Asadi, A. Garcia-Saavedra, G. H. Sim, and M. Hollick, "RadiOrchestra: Proactive management of millimeter-wave self-backhauled small cells via joint optimization of beamforming, user association, rate selection, and admission control," *IEEE Trans. Wireless Commun.*, vol. 22, no. 1, pp. 153–173, Jan. 2023.

[13] W. Lei, Y. Ye, and M. Xiao, "Deep reinforcement learning-based spectrum allocation in integrated access and backhaul networks," *IEEE Trans. Cognit. Commun. Netw.*, vol. 6, no. 3, pp. 970–979, Sep. 2020.

[14] B. Zhang, F. Devoti, I. Filippini, and D. De Donno, "Resource allocation in mmWave 5G IAB networks: A reinforcement learning approach based on column generation," *Comput. Netw.*, vol. 196, Sep. 2021, Art. no. 108248.

[15] Q. Cheng, Z. Wei, and J. Yuan, "Deep reinforcement learning-based spectrum allocation and power management for IAB networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2021, pp. 1–6.

[16] T. K. Vu, C.-F. Liu, M. Bennis, M. Debbah, and M. Latva-aho, "Path selection and rate allocation in self-backhauled mmWave networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2018, pp. 1–6.

[17] A. Ortiz, A. Asadi, G. H. Sim, D. Steinmetzer, and M. Hollick, "SCAROS: A scalable and robust self-backhauling solution for highly dynamic millimeter-wave networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 12, pp. 2685–2698, Dec. 2019.

[18] M. Pagin, T. Zugno, M. Polese, and M. Zorzi, "Resource management for 5G NR integrated access and backhaul: A semi-centralized approach," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 753–767, Feb. 2022.

[19] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *J. Risk*, vol. 2, no. 3, pp. 21–41, 2000.

[20] H. Levy, *Stochastic Dominance*. New York, NY, USA: Springer, 1998.

[21] J. Hoydis et al., "Sionna: An open-source library for next-generation physical layer research," 2022, *arXiv:2203.11854*.

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

[23] *NR; Study on Integrated Access and Backhaul*, document TS 38.874, V.16.0.0, 3GPP, Jan. 2019.

[24] R. T. Rockafellar and S. Uryasev, "Conditional value-at-risk for general loss distributions," *J. Banking Finance*, vol. 26, no. 7, pp. 1443–1471, Jul. 2002.

[25] G. C. Pflug, *Some Remarks on the Value-at-Risk and the Conditional Value-at-Risk*. Boston, MA, USA: Springer, 2000, pp. 272–281, doi: 10.1007/978-1-4757-3150-7_15.

[26] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[27] Z. Luo and S. Ou, "The almost sure convergence rate of the estimator of optimized certainty equivalent risk measure under α-mixing sequences," *Commun. Statist.-Theory Methods*, vol. 46, no. 16, pp. 8166–8177, Aug. 2017.

[28] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Mach. Learn.*, vol. 47, nos. 2–3, pp. 235–256, May 2002.

[29] M. Polese, M. Giordani, A. Roy, S. Goyal, D. Castor, and M. Zorzi, "End-to-end simulation of integrated access and backhaul at mmWaves," in *Proc. IEEE 23rd Int. Workshop Comput. Aided Model. Design Commun. Links Netw. (CAMAD)*, Sep. 2018, pp. 1–7.

[30] M. Mezzavilla et al., "End-to-end simulation of 5G mmWave networks," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 2237–2263, 3rd Quart., 2018.

[31] M. Giordani, M. Polese, A. Roy, D. Castor, and M. Zorzi, "A tutorial on beam management for 3GPP NR at mmWave frequencies," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 173–196, 1st Quart., 2019.

[32] C. Madapatha et al., "On integrated access and backhaul networks: Current status and potentials," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 1374–1389, 2020.

[33] *NR; Backhaul Adaptation Protocol (BAP) Specification*, document TS 38.340, V.17.0.0, 3GPP, May 2022.

[34] *Study on Channel Model for Frequencies From 0.5 To 100 GHz*, document TR 38.901, V.15.0.0, 3GPP, Jun. 2018.

[35] D. Yuan, H. Lin, J. Widmer, and M. Hollick, "Optimal joint routing and scheduling in millimeter-wave cellular networks," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2018, pp. 1205–1213.

[36] S. Hur, T. Kim, D. J. Love, J. V. Krogmeier, T. A. Thomas, and A. Ghosh, "Millimeter wave beamforming for wireless backhaul and access in small cell networks," *IEEE Trans. Commun.*, vol. 61, no. 10, pp. 4391–4403, Oct. 2013.

[37] Y. Zhu, Y. Niu, J. Li, D. O. Wu, Y. Li, and D. Jin, "QoS-aware scheduling for small cell millimeter wave mesh backhaul," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 1–6.

Amir Ashtari Gargari received the Ph.D. degree in information engineering from the University of Padova, Italy, in 2024. He is currently with the Centre Tecnològic de Telecomunicacions de Catalunya (CTTC) as an R2 Researcher (Post-Doctoral Researcher). During his Ph.D. study, he spent five months as a Visiting Researcher with the WISE Group, TU Darmstadt, Germany, and three months at Telefónica Research and Development, Spain. His research interests include artificial intelligence, modeling and simulation, and next-generation wireless communication. He was awarded an EU MINTS Horizon 2020 Marie Skłodowska-Curie Fellows grant to pursue his Ph.D. degree.

Andrea Ortiz (Member, IEEE) is currently a Vienna Research Group Leader holding a Tenure-Track Position at TU Wien. Previously, she was a Post-Doctoral Researcher at TU Darmstadt. Her research interests include the use of reinforcement learning for resource allocation in wireless communications. She was a recipient of several awards, including the Dr. Wilhelmy-VDE-Preis given by German Association for Electrical, Electronic and Information Technologies and the WWTF Vienna Research Group for Young Investigators Grant.

Matteo Pagin received the B.Sc. and M.Sc. degrees in telecommunication engineering from the University of Padova, Italy, in 2018 and 2020, respectively, where he is currently pursuing the Ph.D. degree. From October 2020 to September 2021, he was a Postgraduate Researcher with the Department of Information Engineering. He has collaborated with several institutions, such as Northeastern University, CTTC, NYU, TU Darmstadt, Orange, Viasat, and Huawei. His research interests include the design and evaluation of protocols for next-generation cellular networks (5G and beyond). He was awarded the Best Paper Award from the IEEE MedComNet 2020 and the IEEE WCNC Workshops 2023.

Wanja de Sombre received the B.Sc. and interdisciplinary M.Sc. degrees in mathematics from TU Darmstadt, Darmstadt, Germany, with a focus on mathematical logic and artificial intelligence, where he is currently pursuing the Ph.D. degree with the Communications Engineering Laboratory, under the supervision of Prof. Anja Klein and Dr. Andrea Ortiz. His research interests include reinforcement learning, especially safe and distributional reinforcement learning.

Michele Zorzi (Fellow, IEEE) received the Laurea and Ph.D. degrees in electrical engineering from the University of Padova, Padua, Italy, in 1990 and 1994, respectively. From 1992 to 1993, he was on leave with the University of California at San Diego (UCSD), USA. In 1993, he joined the Faculty of the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milan, Italy. After spending three years with the Center for Wireless Communications, UCSD. In 1998, he joined the School of Engineering, University of Ferrara, Ferrara, Italy, where he was a Professor in 2000. Since November 2003, he has been with the Department of Information Engineering, University of Padova. His current research interests include performance evaluation in mobile communications systems, wireless sensor networks, the Internet of Things, cognitive communications and networking, millimeter-wave and terahertz communications, vehicular networks, non-terrestrial networks, and underwater communications and networks. He has served as the Member-at-Large for the Board of Governors of the IEEE Communications Society from 2009 to 2011 and from 2021 to 2023. He received several awards from the IEEE Communications Society, including the Best Tutorial Paper Award in 2008 and 2019, the Education Award in 2016, the Stephen O. Rice Best Paper Award in 2018, and the Joseph LoCicero Award for Exemplary Service to Publications in 2020. He was the Editor-in-Chief of *IEEE Wireless Communications Magazine* from 2003 to 2005, IEEE TRANSACTIONS ON COMMUNICATIONS from 2008 to 2011, and IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING from 2014 to 2018. He was the Director of Education from 2014 to 2015 and the Director of Journals from 2020 to 2021.

Arash Asadi (Senior Member, IEEE) is currently an Assistant Professor with the Embedded Systems Group, TU Delft, where he leads the Wireless Communication and Sensing Laboratory (WISE). His research interests include wireless communication and sensing for 6G networks. He was a recipient of several awards, including the Athena Young Investigator Award from TU Darmstadt and the Outstanding Ph.D. and Master's Thesis awards from UC3M. Some of his papers on D2D communication have appeared in IEEE COMSOC's best reading topics on D2D communication and IEEE COMSOC Tech Focus.