Switching in the Rain: Predictive Wireless x-haul Network Reconfiguration

IGOR KADOTA*, Columbia University, USA
DROR JACOBY*, Tel Aviv University, Israel
HAGIT MESSER, Tel Aviv University, Israel
GIL ZUSSMAN, Columbia University, USA
JONATAN OSTROMETZKY, Tel Aviv University, Israel

4G, 5G, and smart city networks often rely on microwave and millimeter-wave x-haul links. A major challenge associated with these high frequency links is their susceptibility to weather conditions. In particular, precipitation may cause severe signal attenuation, which significantly degrades the network performance. In this paper, we develop a Predictive Network Reconfiguration (PNR) framework that uses historical data to predict the future condition of each link and then prepares the network ahead of time for imminent disturbances. The PNR framework has two components: (i) an Attenuation Prediction (AP) mechanism; and (ii) a Multi-Step Network Reconfiguration (MSNR) algorithm. The AP mechanism employs an encoder-decoder Long Short-Term Memory (LSTM) model to predict the sequence of future attenuation levels of each link. The MSNR algorithm leverages these predictions to dynamically optimize routing and admission control decisions aiming to maximize network utilization, while preserving max-min fairness among the nodes using the network (e.g., base-stations) and preventing transient congestion that may be caused by switching routes. We train, validate, and evaluate the PNR framework using a dataset containing over 2 million measurements collected from a real-world city-scale backhaul network. The results show that the framework: (i) predicts attenuation with high accuracy, with an RMSE of less than 0.4 dB for a prediction horizon of 50 seconds; and (ii) can improve the instantaneous network utilization by more than 200% when compared to reactive network reconfiguration algorithms that cannot leverage information about future disturbances.

CCS Concepts: • **Networks → Control path algorithms**; *Network performance analysis*; • **Computing methodologies → Machine learning**.

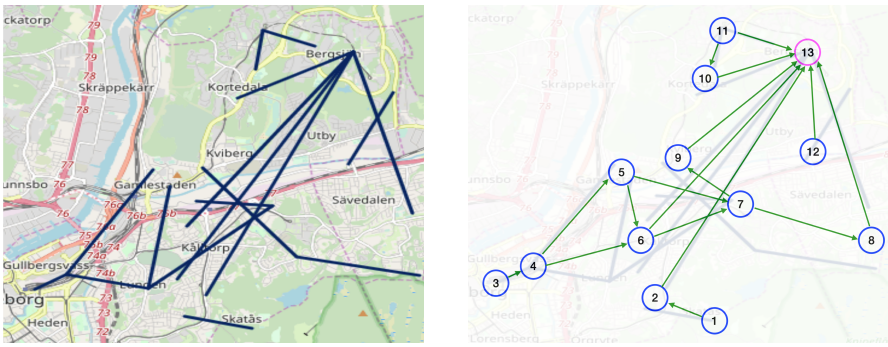
Additional Key Words and Phrases: wireless networks; millimeter-wave; fronthaul; backhaul; 5G; routing; machine learning; rain attenuation

ACM Reference Format:
Igor Kadota, Dror Jacoby, Hagit Messer, Gil Zussman, and Jonatan Ostrometzky. 2022. Switching in the Rain: Predictive Wireless x-haul Network Reconfiguration. *Proc. ACM Meas. Anal. Comput. Syst.* 6, 3, Article 55 (December 2022), 26 pages. https://doi.org/10.1145/3570616

*Both authors contributed equally to this research.

Authors' addresses: Igor Kadota, Columbia University, New York, USA, igor.kadota@columbia.edu; Dror Jacoby, Tel Aviv University, Tel Aviv, Israel, drorjacoby@mail.tau.ac.il; Hagit Messer, Tel Aviv University, Tel Aviv, Israel, messer@eng.tau.ac.il; Gil Zussman, Columbia University, New York, USA, gil.zussman@columbia.edu; Jonatan Ostrometzky, Tel Aviv University, Tel Aviv, Israel, jonatano@tauex.tau.ac.il.

2476-1249/2022/12-ART55 $15.00
https://doi.org/10.1145/3570616



(a) (b)

Fig. 1. (a) A wireless backhaul network in Gothenborg, Sweden (the map area is of approximately 10x10 km^2). The data utilized in this paper was collected from this network by *Ericsson AB* and is available, in part, at [3]. (b) An abstraction of the network topology (described in Sec. 2).

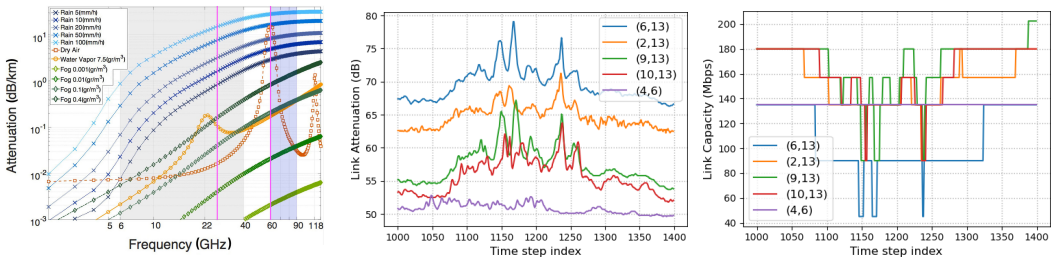
1 INTRODUCTION

4G, 5G, and smart city networks often use high bandwidth microwave and millimeter-wave (mmWave) links in their fronthaul, midhaul, and backhaul (x-haul) networks [14]. These wireless x-haul networks can connect a large number of Base-Stations (BSs), covering entire cities, as depicted in Fig. 1(a). However, a major challenge is the high susceptibility of microwave and mmWave links to weather conditions.

The signal attenuation due to different atmospheric and weather phenomena is described by the International Telecommunication Union in [21–24] and depicted in Fig. 2(a). It can be seen from Fig. 2(a) that, apart from the oxygen resonance frequency at 60 GHz, the dominant factor affecting link attenuation is precipitation. Notice that the stronger the downpour, the stronger the attenuation. Fig. 2(b) shows attenuation measurements from five links of the network in Fig. 1 collected during a rainy period on 2015-06-02. The impact of the attenuation on the capacity of the wireless links is displayed in Fig. 2(c). It is evident from Fig. 2 that weather-induced attenuation and the resulting capacity degradation vary over time, geographical location, rain intensity, and can be severe. *The need for a high capacity wireless x-haul that is robust to variations in the links' conditions calls for the development of a predictive network reconfiguration framework that can dynamically allocate resources based on current and future estimated conditions.*

Until recently, only local Physical/Link layer mechanisms were employed to alleviate the impact of the time-varying conditions of the links on the network performance. For example, the Automatic Transmit Power Control is a commonly used mechanism that adjusts the transmitter power based on measurements of the link attenuation. However, with the emergence of Software-Defined Networking (SDN) [10, 28, 38], it is now possible to develop global Network layer mechanisms (such as NEC's backhaul solution [40]) that monitor the entire network and react to weather-induced disturbances. However, a main drawback of *reactive reconfiguration* mechanisms is their delay in mitigating performance issues, which may severely affect time-sensitive applications. To overcome this challenge, *predictive reconfiguration* mechanisms should be developed.

Related Work: Prior work on predictive network reconfiguration algorithms (see [8] for a survey) focused mainly on alleviating the effects of node mobility [11, 13, 34, 37, 39, 60], traffic demand variability [1, 4, 7, 17, 19, 43, 48, 51, 55], and link quality degradation due to multi-path reflection, line-of-sight occlusion, and interference [9, 36, 52–54, 58, 61, 62]. Weather effects pose fundamentally



(a) (b) (c)

Fig. 2. (a) Signal attenuation (in dB/km) for various atmospheric phenomena as a function of frequency [21]. The commonly used bands of 6–40 GHz (K-band) and 60–90 GHz (E-band) are highlighted. (b) Measured attenuation over time for five links from the network in Fig. 1. Time-steps are separated by 10 seconds, adding up to 1.1 hours of measurements collected on 2015-06-02. Between time-steps 1, 100 and 1, 300 there was an increased attenuation due to rain which affected links $(2, 13)$, $(6, 13)$, $(9, 13)$, and $(10, 13)$, *in that order*, while not affecting link $(4, 6)$. Comparing with the network topology in Fig. 1(b), we can see that the rain was moving from Southeast to Northwest, and did not affect link $(4, 6)$ located on the Southwest. The spatio-temporal correlation is evident. (c) Link capacities over time for the same five links.

different challenges. In particular, weather-induced attenuation can be severe, affect large contiguous geographic areas, and last for extended periods of time. The literature on the prediction of microwave and mmWave signal attenuation due to weather conditions uses meteorological data (e.g., weather-radar echo measurements) to predict the current/future attenuation levels [25, 45] or uses past attenuation measurements to predict future attenuation levels [26, 27, 44, 50, 59]. Most relevant to this paper is [26] which employs an encoder-decoder LSTM model to predict future link attenuation levels. The main drawbacks of [26] are that: (i) it does not capture the significant *spatial correlation* of the rain-induced attenuation (i.e., predictions are at per-link level rather than for a network of links); and (ii) the prediction mechanism is not employed to inform any network algorithm.

The literature on predictive weather-aware reconfiguration algorithms is rather sparse [25, 29, 45, 59]. In particular, [25, 29, 45] develop modifications to standard *distributed* routing protocols, such as Open Shortest Path First (OSPF), which may converge slowly, thereby making them unsuitable for networks that support time-sensitive applications. SDN is leveraged in [59] to perform centralized predictive network-wide reconfiguration. The framework of [59] predicts the future attenuation of each link using a mathematical model *specific to rain fading* and then computes current and future routing decisions aiming to maximize throughput. However, developing a framework that can capture the spatial correlation of the weather-effects, predict future link attenuation levels during dry and rain periods, prevent transient congestion that may be caused by re-routing, and/or guarantee fairness in the allocation of network resources is still an open problem.

Our contributions: We develop and evaluate (based on a real dataset) a *Predictive Network Reconfiguration (PNR) framework*, illustrated in Fig 3, that prepares the network ahead of time for imminent disturbances, significantly enhancing the network's robustness to variations in the links' conditions. The PNR leverages existing local Physical/Link layer mechanisms and adds an Attenuation Prediction (AP) mechanism and a Multi-Step Network Reconfiguration (MSNR) algorithm.

The AP mechanism employs an encoder-decoder LSTM model to predict the sequence of future attenuation levels based on past measurements, capturing both time and spatial correlation that are typical of weather-effects *without incorporating weather-related models*. This allows it to

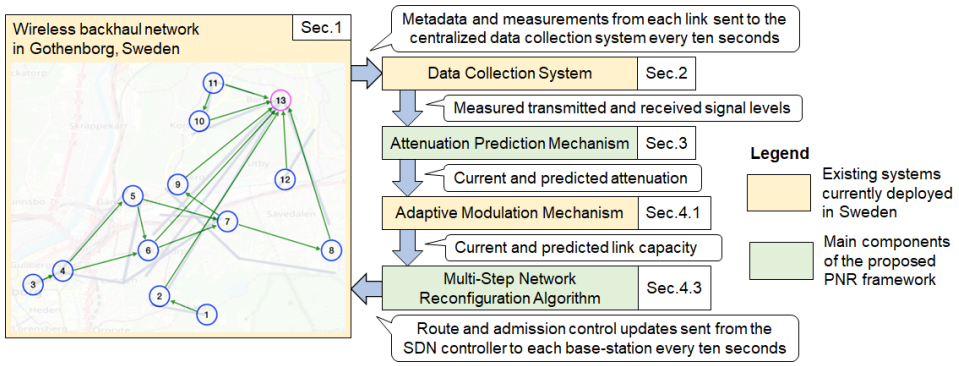


Fig. 3. Overview of the main components and information flow in the proposed PNR framework.

be used both in dry and rain periods, and *without relying on meteorological data from external sources* such as weather radars. To train, validate, and evaluate the AP mechanism, we use *a dataset obtained from the real-world city-scale backhaul network in Gothenborg, Sweden* (see Fig. 1(a)), *collected by Ericsson AB* and available[1], in part, at [3]. The dataset utilized in this paper contains 2,295,000 measurements of link attenuation levels. In Fig. 2(b), we can see that weather-induced attenuation depends not only on the time-varying rain intensity, but also on links' characteristics such as frequency, polarization, and length, which makes the prediction task challenging. *The AP mechanism leverages the spatio-temporal correlation of the weather-effects to achieve high attenuation prediction accuracy.*

The MSNR algorithm leverages the predictions from the AP mechanism to prepare the network for future disturbances. Specifically, it generalizes the Maximum Concurrent Flow [2, 49] problem and uses Model Predictive Control [35] to compute the sequence of current and future routing and admission control decisions that: (i) maximize network utilization, while (ii) guaranteeing max-min fairness among the BSs sharing the network, and (iii) preventing transient congestion that may be caused by switching routes. These routing and admission control decisions are employed by the centralized SDN controller to reconfigure the network over time. For example, based on a prediction that a set of links will become unavailable in 30 seconds, the MSNR algorithm can determine when it is optimal for the SDN controller to redirect flows in order to avoid potential interruptions to service and can decide whether or not it is necessary to decrease traffic admission from low priority services. An important challenge associated with the MSNR algorithm is computational complexity. We present a principled implementation of the MSNR algorithm which has a computational complexity that grows polynomially with the prediction horizon, as opposed to a naive implementation that can have exponential complexity.

We evaluate the PNR framework using the data collected from the backhaul network. Specifically, we show that the AP mechanism can achieve high attenuation prediction accuracy with a Root Mean Square Error (RMSE) of less than 0.4 dB for a prediction horizon of 50 seconds. We evaluate two benchmark time series prediction methods that do not capture the spatial correlation of the weather-effects and show that both of them can perform 30% worse than the AP mechanism in terms of RMSE. In addition, we show that the MSNR algorithm can improve the instantaneous network

[1]In general, datasets with high frequency link attenuation measurements are proprietary information of the network operators. However, many data-processing code repositories with example datasets similar to the one used in the paper are available online as described in [32]. For access, one can join the collaborative COST Action OpenSense (https://opensenseaction.eu/join-us/), which supports data sharing.

utilization by more than 200% when compared to reactive network reconfiguration algorithms that do not prepare the network for future disturbances.

To the best of our knowledge, this is the first attempt to propose and evaluate, based on a real dataset, an integrated framework for x-haul network reconfiguration that leverages the spatio-temporal correlation of the weather-effects to jointly optimize routing and admission control decisions. A patent including some of the results is pending [42]. The remainder of this paper is organized as follows. In Sec. 2, we describe the network model and the dataset. In Sec. 3, we develop the AP mechanism. In Sec. 4, we develop the MSNR algorithm. In Sec. 5, we evaluate the performance of the PNR framework. In Sec. 6, we conclude the paper and discusses future work.

2 PROBLEM FORMULATION AND DATASET

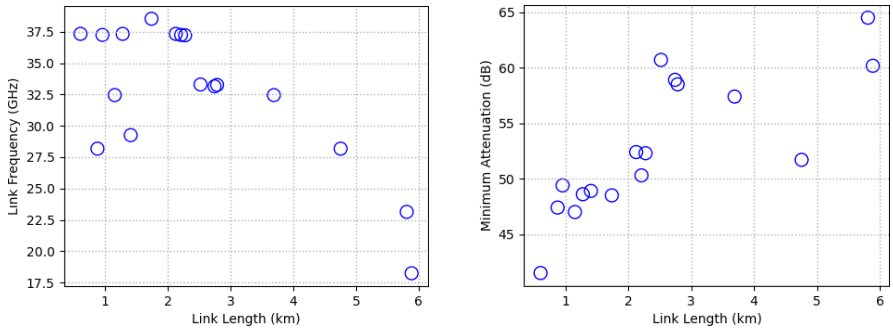
We now present the network model used to develop the PNR framework. We first describe the model in general and then establish the connection between the model and the real-world backhaul network. Let $G = (V, E)$ be the directed graph that represents an x-haul communication network with BSs, also called nodes, $n \in V = \{1, 2, \ldots, N\}$, connected by wireless links $(k, l) \in E$ where $k, l \in V$ and (k, l) represents the link $k \rightarrow l$. Time is divided into time-steps with index $t \in \{1, 2, \ldots, T\}$, where T is the time-horizon and the time interval between t and $t + 1$ is $\Delta = 10$ seconds (without loss of generality). Let $d_n > 0$ be the demand associated with commodity $n \in V$. The demand d_n represents the uplink traffic that BS n aggregates from its associated users. Let $z_{n,t} \in [0, 1]$ be the fraction of the demand d_n admitted during time-step t. It follows that the *admitted demand* from BS n during time t is given by $z_{n,t}d_n$. For simplicity, we assume[2] that demands d_n remain fixed over time and that node N is the common destination for all commodities $n \in V \setminus N$. Naturally, for the common destination N, we have $d_N = 0$ and $z_{N,t} = 0, \forall t$. Let $f_{n,t}^{(k,l)} \in [0, 1]$ be the fraction of the admitted demand $z_{n,t}d_n$ that flows through link (k, l) during time t. The constraints on the incoming flows at the source nodes are given by $f_{n,t}^{(k,n)} = 0, \forall (k, n) \in E, \forall t$, the constraints on the outgoing flows at the destination node N are given by $f_{n,t}^{(N,l)} = 0, \forall n \in V, \forall (N, l) \in E, \forall t$, and non-existing links are such that $f_{n,t}^{(k,l)} = 0, \forall n \in V, \forall (k, l) \notin E, \forall t$. The *total flow* in link $(k, l) \in E$ during time t is given by $\sum_{n=1}^{N-1} z_{n,t}d_n f_{n,t}^{(k,l)}$.

Feasibility and Fairness. We assume that $G = (V, E)$ and d_n are given and remain fixed over time. We assume that routing and admission control decisions by the centralized SDN controller at time-step t, namely $f_{n,t}^{(k,l)}$ and $z_{n,t}$, respectively, remain fixed in the interval between t and $t + 1$. Routing and admission control decisions at each time t are *feasible* when they satisfy flow conservation and capacity constraints. The *flow conservation* associated with commodity $n \in V$ and node $l \in V$ at time t is given by

$$\sum_{k=1}^{N} f_{n,t}^{(k,l)} - \sum_{m=1}^{N} f_{n,t}^{(l,m)} = \mathbb{1}_{\{\mathrm{l=N}\}} - \mathbb{1}_{\{\mathrm{l=n}\}} , \tag{1}$$

where $\mathbb{1}_{\{\mathrm{X}\}}$ is the indicator function that is 1 when X is true and 0 otherwise, $l = n$ indicates that node l is the source of commodity n, and $l = N$ indicates that node l is the destination of commodity n. $c_t^{(k,l)} \geq 0$ is the capacity of link (k, l) at time t and $\hat{c}_{t+1}^{(k,l)} \geq 0$ is the *predicted* capacity of link (k, l) at time $t + 1$. Since the exact moment between t and $t + 1$ in which the capacity changes from $c_t^{(k,l)}$ to $\hat{c}_{t+1}^{(k,l)}$ is unknown, we assume the worst-case and represent the capacity in this interval by $\min\{c_t^{(k,l)}, \hat{c}_{t+1}^{(k,l)}\}$. Hence, the *capacity constraint* associated with link $(k, l) \in E$ at time t is given

[2]The PNR framework can be easily adapted to networks with multiple destinations and to incorporate mechanisms that predict time-varying traffic demands d_n, such as the mechanisms developed in [17, 51].



(a) (b)

Fig. 4. Characteristics of the 17 links that compose the backhaul network. (a) Center frequency and geographical Euclidean distance. (b) Minimum attenuation and length. The minimum attenuation is selected among the measurements collected on 2015-06-02 over a period of 14.3 hours. The minimum attenuation represents the baseline attenuation during a dry period.

by

$$\sum_{n=1}^{N-1} z_{n,t} d_n f_{n,t}^{(k,l)} \leq \min\{c_t^{(k,l)}, \hat{c}_{t+1}^{(k,l)}\} \ . \tag{2}$$

Definition 1 (Feasibility). *The set of routing and admission control decisions at time t, namely $\{f_{n,t}^{(k,l)}, z_{n,t}\}$, $\forall n \in V$, $\forall (k,l) \in E$, is* feasible *when it satisfies the flow conservation in (1) and the capacity constraints in (2).*

Definition 2 (Max-Min Fairness). *The feasible set $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ at time-step t has admission rates $z_{n,t}$ that are max-min fair if, in order to* maintain feasibility, *an increase of any $z_{n,t}$ necessarily results in the decrease of $z_{m,t}$ of another source m for which $z_{m,t} \leq z_{n,t}$.*

The goal of the PNR framework is to dynamically optimize routing and admission control decisions over time, taking into account future predicted network conditions, aiming to maximize the cumulative sum of admission rates $\sum_{t=1}^{T} \sum_{n=1}^{N-1} z_{n,t}$, while guaranteeing that, *in each time-step t*, the selected feasible set $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ is max-min fair and can be implemented without inducing transient congestion. Omitting the fairness constraint and obtaining a solution to the corresponding Maximum Flow problem could lead to an increased cumulative sum of admission rates $\sum_{t=1}^{T} \sum_{n=1}^{N-1} z_{n,t}$. However, a critical drawback of this "unfair" solution is that it may starve flows from one or more BSs. To avoid that, we formulated the dynamic optimization problem with max-min fairness at its center. This challenging optimization problem and its complexity are addressed in Sec. 4.

Real-World Network and Dataset. Consider the backhaul network in Fig. 1(a) composed of 17 line-of-sight wireless links whose lengths vary from 0.6 to 5.9 km, operating between 18 and 40 GHz, using vertical polarization, and with antenna diameters ranging from 20 cm to 1.2 m and antenna gains ranging from 31 dBi to 47 dBi. Fig. 4 shows the links' center frequencies, lengths, and minimum attenuation levels. Notice that, in general, higher-frequency links are installed at shorter distances to reduce attenuation.

The directed graph $G = (V, E)$ with $N = 13$ nodes in Fig. 1(b) is generated by assuming that *link endpoints* in Fig. 1(a) which are in close proximity (up to 300 m apart) are connected by fiber

which is not capacity-limited. Under this assumption[3], a node in $G = (V, E)$ represents one or more neighboring link endpoints in Fig. 1(a).

The data collection system, described in [3, 6], runs at Ericsson[4] and utilizes the Simple Network Management Protocol to fetch the latest measurements of the transmitted and received signal levels (in dB), represented by $P_{Tx,t}^{(k,l)}$ and $P_{Rx,t}^{(k,l)}$, respectively, at every 10 sec from every link in the backhaul network. According to [3, 6], the extra load associated with the transmission of measurements via the x-haul network is insignificant (around 10 bytes per second per node) and, in general, the measurements are ready in the data collector within 1 to 3 seconds. From May to December 2015, the data collection success rate, excluding maintenance periods, was greater than 99.6% with data loss being caused by rare events such as link blockages. Notice that the PNR framework could help to further improve the data collection success rate by re-routing flows before link blockages.

Throughout the paper, we assume that in each time-step t, the following events occur: (i) the data collection system shares the latest measurements of $P_{Tx,t}^{(k,l)}$ and $P_{Rx,t}^{(k,l)}$ with the centralized PNR framework; (ii) the AP mechanism computes the attenuation levels $x_t^{(k,l)} = P_{Tx,t}^{(k,l)} - P_{Rx,t}^{(k,l)}$ of every link and predicts future attenuation levels; (iii) the MSNR algorithm computes new routing $f_{n,t}^{(k,l)}$ and admission control $z_{n,t}$ decisions; and (iv) the SDN controller implements the new network configuration by propagating $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ to the corresponding BSs. Similarly to [3, 6], we assume that the extra load associated with the transmission of routes and admission control updates $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ utilize negligible resources of the x-haul network, as they are transmitted at most once every $\Delta = 10$ seconds.

To train, validate, and evaluate the PNR framework we use a dataset containing measurements collected during 2015 from the network in Fig. 1(a). We use the method proposed in [47] to detect rainy periods[5]. Then, we select 20 days which contain both rainy and dry periods and use 2, 295, 000 measurements (i.e., 135, 000 per link) from these 20 days to train/test the encoder-decoder LSTM model. The *test* data utilized to evaluate the PNR framework in Sec. 5 is composed of three sequences of measurements, **each containing a period of rain**: *Test Seq. I* with 87, 890 measurements collected over a period of 14.3 hours on 2015-06-02, *Test Seq. II* with 11, 900 measurements collected over a period of 1.9 hours on 2015-05-19, and *Test Seq. III* with 94, 690 measurements collected over a period of 15.5 hours on 2015-06-17.

3 ATTENUATION PREDICTION MECHANISM

In this section, we present the AP mechanism which predicts the sequence of future attenuation levels based on historical data, as illustrated in Fig. 3. The high accuracy of the AP mechanism (validated in Sec. 5.1) stems from: (i) the suitability of the encoder-decoder LSTM model to sequence-to-sequence learning problems, such as machine translation, natural language generation, and speech recognition [15, 16, 57]; and (ii) the capability of LSTMs to capture both *time* and *spatial* correlations that are typical of weather-effects. The LSTM model has two main parts: the *encoder*, which maps the input sequence into a state vector and the *decoder*, which maps the state vector into a sequence of predictions.

The AP mechanism employs the sliding-window method and the encoder-decoder LSTM model illustrated in Figs. 5(a) and 5(b), respectively, to predict the next H attenuation levels based on the previous W measurements. In particular, let $x_t^{(k,l)} = P_{Tx,t}^{(k,l)} - P_{Rx,t}^{(k,l)}$ be the *attenuation measurement*

[3]Other assumptions could have been made without affecting the results' generality.

[4]The office is located in Gothenborg, in the West part of the map in Fig. 1(a)

[5]This method is based on the observation that the variability of the attenuation increases during rain. In [47], this method was shown to accurately classify rainy/dry periods without the need for side information (such as meteorologic data).

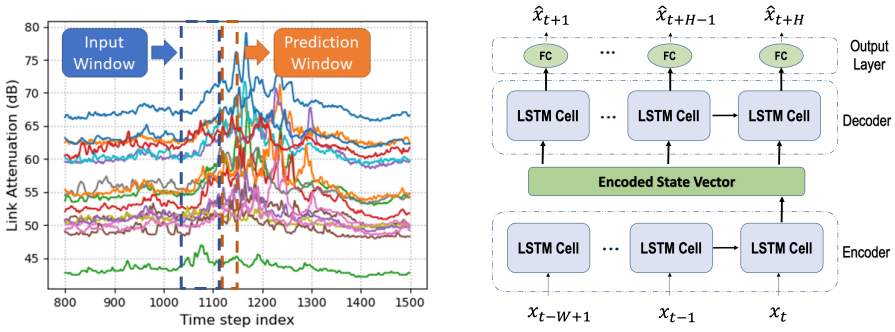


Fig. 5. (a) Measured attenuation for all the 17 links in the network in Fig. 1 with time-steps t separated by 10 seconds (adding up to 1.9 hours) together with an illustration of the sliding-window method with *input window* size of W time-steps and *prediction window* size of H time-steps. The different baseline attenuation levels are due to the difference in the links' frequency and length. (b) Encoder-decoder LSTM model that employs the last W attenuation measurements $\{\boldsymbol{x_{t-W+1}}, \ldots, \boldsymbol{x_{t-1}}, \boldsymbol{x_t}\}$ from every link in the network (i.e., the input window) to predict the future H attenuation levels $\{\hat{\boldsymbol{x}}_{\boldsymbol{t+1}}, \ldots, \hat{\boldsymbol{x}}_{\boldsymbol{t+H-1}}, \hat{\boldsymbol{x}}_{\boldsymbol{t+H}}\}$ in each link (i.e., the prediction window).

for link $(k,l) \in E$ at time t, and let $\boldsymbol{x_t} = (x_t^{(k,l)})$ and $\hat{\boldsymbol{x}}_{\boldsymbol{t+h}} = (\hat{x}_{t+h}^{(k,l)})$ be the vector of *attenuation measurements* and the vector of *h-steps-ahead attenuation predictions* for all links at time t, respectively. In each time-step t, the encoder-decoder LSTM model employs the sequence of measurements in the *input window* $\{\boldsymbol{x_{t-W+1}}, \boldsymbol{x_{t-W+2}}, \ldots, \boldsymbol{x_t}\}$ to predict the sequence of attenuation levels in the *prediction window* $\{\hat{\boldsymbol{x}}_{\boldsymbol{t+1}}, \hat{\boldsymbol{x}}_{\boldsymbol{t+2}}, \ldots, \hat{\boldsymbol{x}}_{\boldsymbol{t+H}}\}$. Notice that the measurements in the *input window* allow the encoder-decoder LSTM model to capture the spatio-temporal correlation of weather-induced attenuation. We employ an input window size of $W = 12$ and a prediction window size of $H = 5$, which corresponds to 120 seconds and 50 seconds, respectively.

We train the encoder-decoder LSTM model to minimize the prediction error. In particular, consider a dataset with a sequence of attenuation measurements in the interval $t \in \{1, \ldots, T\}$. The encoder and decoder are jointly trained to minimize the objective function:

$$\mathcal{L}(\Theta) = \sum_{t=1}^{T-H} \sum_{h=1}^{H} \| \boldsymbol{x_{t+h}} - \hat{\boldsymbol{x}}_{\boldsymbol{t+h}} \|^2 \tag{3}$$

where $\|\cdot\|$ is the Euclidean norm, and Θ represents the parameters of the encoder-decoder LSTM model, i.e., weights and biases. We implement the encoder and the decoder LSTM with one hidden layer containing 128 units. To train, tune, and evaluate the AP mechanism, we use the dataset described in Sec. 2 with a *train-validation-test split of 80-10-10*. We train the AP mechanism using Backpropagation Through Time [56] and Adaptive Moment Estimation (Adam) [31] with a batch size of 150. The prediction accuracy of the AP mechanism is evaluated in Sec. 5.1.

4 MULTI-STEP NETWORK RECONFIGURATION ALGORITHM

SDN enables the design of algorithms that dynamically reconfigure the entire network. Building on that capability, we develop the MSNR algorithm, which leverages information about links' future conditions to compute the sequence of current and future routing and admission control decisions that: (i) attempt to maximize network utilization, while (ii) guaranteeing max-min fairness (in every

time-step t) among the BSs sharing the network, and (iii) preventing transient congestion that may be caused by switching routes. Hereafter, we denote this sequence of routing and admission control decisions as the *optimal sequence of network configurations*.

The problem of finding the optimal sequence of network configurations is a generalization of the Maximum Concurrent Flow (MCF) problem. The solution to the traditional MCF problem in [2] considers a network with *fixed* link capacities and finds routing and admission control decisions that maximize admission rates at time t while ensuring that the admitted traffic is max-min fair. The MSNR algorithm generalizes this solution to the more challenging setting where: (i) link capacities are time varying; (ii) the optimization is performed (jointly) over multiple time-steps $t, t+1, \cdots, t+H$; and (iii) there is a performance cost for re-routing. The time-varying link capacities and the re-routing costs lead to a more realistic but significantly more complex network model, since they create a *trade-off between the future expected benefits of re-routing and the present cost of re-routing*.

The MSNR algorithm employs Model Predictive Control to address this generalized MCF optimization problem. In particular, in each time-step t, the MSNR algorithm uses its knowledge of future (predicted) network conditions to evaluate and compare the performance of different congestion-free max-min fair sequences of network configurations $\{f_{n,t+h}^{(k,l)}, z_{n,t+h}\}$, $\forall n \in V$, $\forall (k,l) \in E$, $\forall h \in \{0, 1, \ldots, H-1\}$ and then selects the sequence with highest cumulative sum of admission rates $\sum_{h=0}^{H-1} \sum_{n=1}^{N-1} z_{n,t+h}$. The SDN controller implements the first configuration in the selected sequence, i.e., the configuration $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ associated with the current time t. This iterative process allows the SDN controller to account for future predicted network conditions when optimizing the current network configuration. An important feature of the MSNR algorithm is that it guarantees that the selected network configuration is always *max-min fair*.

A major challenge associated with the MSNR algorithm is computational complexity. A naive implementation computes and compares the performance of all possible sequences of network configurations within the prediction window $\{t, \ldots, t+H\}$. The number of such sequences grows exponentially with H, as discussed in Sec. 4.3, which could render the MSNR algorithm impractical. To overcome this challenge, we develop a principled implementation of the MSNR algorithm which employs the structure of the optimization problem to recursively explore the space of all possible sequences of network configurations. This recursive method reduces the complexity from exponential $O(2^H)$ to polynomial $O(H^4)$.

Prior to introducing the MSNR algorithm, we describe: (i) the adaptive modulation mechanism in [5], which is a Physical layer mechanism employed by the backhaul network in Sweden to maximize link capacity over time; and (ii) the SWAN mechanism developed in [18], which is a Network layer mechanism that eliminates transient congestion that may be caused by re-routing. The MSNR algorithm builds upon both these existing solutions to optimize routing and admission control decisions over time, as illustrated in Fig. 3.

4.1 Adaptive Modulation Mechanism

Three parameters that can be dynamically adjusted to compensate for high attenuation levels in microwave and mmWave links are: the transmission power, the coding rate, and the modulation scheme. The dataset utilized in this paper was collected for a backhaul network that uses radios similar to the ones described in [3, 5, 6] which: (i) employ a constant transmit power $P_{Tx,t}^{(k,l)}$ and a constant coding rate over time; (ii) use Quadrature Amplitude Modulation (QAM) with *adaptive* constellation size M; and (iii) use a fixed channel bandwidth of 28 MHz that achieves a capacity of 45 Mbps when $M = 4$. Recall that when M is increased by a factor of k, the capacity $c_t^{(k,l)}$ increases by a factor of $\log_2 k$ and the Bit Error Rate (BER) decreases according to [12, Eq. (18)].

Table 1. Parameters of the adaptive modulation mechanism with hysteresis for a BER threshold of 10^{-9}.

M	Bitrate (Mbps)	Limit up (dBm)	Limit down (dBm)
4	45	-72	N/A
16	90	-66	-74
64	135	-62.5	-68
128	157	-61	-64
256	180	-57	-62
512	202.5	-53	-58
1024	225	N/A	-54

The adaptive modulation (AM) mechanism adjusts the constellation size M over time, aiming to maximize link capacity $c_t^{(k,l)}$ while keeping the BER above a given threshold. To comply with the radios in [5, Sec. II.B], hereafter, we consider a wireless x-haul network that employs the *AM mechanism with hysteresis* represented in Table 1. Table 1 represents a mapping from the evolution of the received signal levels $P_{Rx,t}^{(k,l)}$ over time to the evolution of the link capacities $c_t^{(k,l)}$ over time. In particular, we assume that every link $(k,l) \in E$ uses radios that adapt their constellation size M at each time-step t based on Table 1 and on their measured received signal level $P_{Rx,t}^{(k,l)}$. The *limit up* in Table 1 represents the received signal level in which the adopted M should increase. The *limit down* represents the received signal level in which the adopted M should decrease to keep the BER above the set threshold.

4.2 The Cost of Re-routing

One possible approach to dynamically optimizing the network configuration *without resorting to predictions of links' future conditions* is for the SDN controller to carry out, in each time t, the following procedure: (i) gather information about the current link capacities $c_t^{(k,l)}$; (ii) employ existing solutions to the MCF optimization problem (e.g., [2, 49]) to find the configuration $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ that maximizes the *current* network utilization; and (iii) implement the new routing decisions $f_{n,t}^{(k,l)}$ and admission rates $z_{n,t}$ by sending control packets to the BSs in the x-haul network. Upon reception of these control packets, the BSs add/remove entries from their routing tables and adjust their admission control accordingly. Two important drawbacks of this approach are the delay to recover from performance degradation, which is characteristic of reactive reconfiguration mechanisms, and that it does not take into account the transient congestion that may be caused by re-routing. Both drawbacks may severely affect time-sensitive traffic. The MSNR algorithm proposed in Sec. 4.3 addresses both drawbacks. In this section, *we discuss the negative effects that re-routing may have on the network performance.*

To update routes from $f_{n,t-1}^{(k,l)}$ to $f_{n,t}^{(k,l)}$, the SDN controller may have to send control packets to multiple BSs. Due to communication/processing delays, some BSs may apply the new routes $f_{n,t}^{(k,l)}$ while others still employ old routes $f_{n,t-1}^{(k,l)}$, which may cause significant transient congestion and over-utilization of communication links, namely violation of the capacity constraints (2). Depending on the duration and magnitude of the congestion, data packets may be severely delayed or even lost. Hence, the re-routing process can have performance impacts that should be taken into account when the SDN controller decides to re-route or not.

In order to reduce the transient congestion associated with re-routing, a common approach (e.g., [18, 33, 59]) is to subdivide the re-routing process into multiple stages. In each stage, the SDN

controller updates a small number of BSs. Each stage is designed to generate zero (or little) transient congestion and the complete sequence of stages is designed to lead to the desired final routing configuration. An important constraint is that the time for completing the re-routing process should be shorter than the interval between two consecutive time-steps, e.g., t and $t+1$, which in this paper is $\Delta = 10$ seconds. In [18, 33, 59], the authors propose different route implementation systems that attempt to minimize the transient congestion. Yet, these route implementation systems can only guarantee that re-routing is performed with zero congestion when a fraction of the network capacity is not utilized before the update. Naturally, when all links are fully utilized, the first update to take effect will always congest at least one link.

In this paper, we consider an SDN controller that implements new routes $f_{n,t}^{(k,l)}$ by employing the *SWAN mechanism* [18], which leverages *scratch capacity* in *every link* to perform congestion-free re-routing. In particular, [18] shows that SWAN can update routes, i.e., change from $f_{n,t-1}^{(k,l)}$ to any given $f_{n,t}^{(k,l)}$, with zero transient congestion in *at most* $\lceil 1/s_t \rceil - 1$ stages, where $s_t \in (0,1]$ represents the *scratch capacity of the network* at time t. s_t is given by

$$s_t = \underset{s\in(0,1]}{\operatorname{argmax}} \left\{ \sum_{n=1}^{N-1} z_{n,t-1} d_n f_{n,t-1}^{(k,l)} \leq (1-s) c_t^{(k,l)}, \forall (k,l) \in E \right\}. \tag{4}$$

More details on the SWAN route implementation mechanism are available in [18]. Notice that when the network has no *scratch capacity*, i.e., $s_t \to 0$, the SWAN mechanism needs $\lceil 1/s_t \rceil - 1 \to \infty$ stages to complete a single congestion-free re-routing process. We assume that the period to complete each stage is $\ll 0.5$ seconds and we impose a lower bound of $s_{min} = 0.05$ on the *scratch capacity*, s_t. This way, we ensure that the time to complete a re-routing process is $\ll \Delta_t = 10$ seconds. Hereafter, we assume that the SDN controller is allowed to re-route at time t, if and only if $s_t \geq s_{min} = 0.05$.

The SDN controller employs the MSNR algorithm to compute the optimal sequence of network configurations over time and, when necessary, it employs the SWAN mechanism to implement new routes. In particular, in each time-step t, given the prior routing and admission control decisions, $\{f_{n,t-1}^{(k,l)}, z_{n,t-1}\}$, the SDN controller employs (4) to calculate the current scratch capacity s_t. If $s_t \geq s_{min}$, the SDN controller employs the MSNR algorithm to compute the optimal sequence of network configurations and then it employs SWAN to implement the optimal configuration $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ at the current time t. Alternatively, if $s_t < s_{min}$, the SDN controller is not allowed to re-route at time t, but it can still optimize the admission rates $z_{n,t}$. In this case, the SDN controller employs the MSNR algorithm *with fixed routing parameters* $f_{n,t}^{(k,l)} = f_{n,t-1}^{(k,l)}$ to compute the optimal sequence of network configurations and then it implements the optimal admission rates $z_{n,t}$ at time t. It is easy to see that admission rates can be updated from $z_{n,t-1}$ to $z_{n,t}$ with zero transient congestion in at most two stages, irrespective of the value of s_t. In the first stage, the SDN controller updates all BSs in which $z_{n,t-1} > z_{n,t}$ and, in the last stage, the SDN controller updates all BSs in which $z_{n,t-1} < z_{n,t}$.

The routing and admission control decisions *at time* t determine the scratch capacity s_{t+1} *at time* $t+1$, which determines whether the SDN controller will be allowed to re-route at time $t+1$. Hence, if the SDN controller plans to re-route at time $t+1$, it should select a network configuration $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ that will lead to $s_{t+1} \geq s_{min}$. This can be achieved by employing, at time t, the following capacity constraint for every link $(k,l) \in E$

$$\sum_{n=1}^{N-1} z_{n,t} d_n f_{n,t}^{(k,l)} \leq \min\{c_t^{(k,l)}, (1-s_{min})\hat{c}_{t+1}^{(k,l)}\}. \tag{5}$$

Alternatively, if the SDN controller plans to keep the same routes in the next time-step, i.e., $f_{n,t+1}^{(k,l)} = f_{n,t}^{(k,l)}$, it should attempt to fully utilize the links, leaving no scratch capacity. This can be achieved by employing the capacity constraints in (2). Intuitively, this means that, in order to re-route in the next time-step $t+1$, the SDN controller may need to reduce the admission rates $z_{n,t}$ in the current time-step t. *This reduction of $z_{n,t}$ represents the cost of re-routing*, as illustrated in the following example.

Example: consider the network in Fig. 8(a) with $N = 3$ nodes, three links $\{(1,2),(2,3),(1,3)\}$, and fixed demands $d_1 = 1$ and $d_2 = 0.5$. Assume that this network has capacities $c_t^{(k,l)} = 0.5$ for all links and predicted capacities $\hat{c}_{t+h}^{(k,l)} = 0.5$ for all links and prediction horizons h. Moreover, assume that $s_t \geq s_{min} = 0.05$, meaning that the SDN controller is allowed to re-route at the current decision time t.

Plan to not re-route. If the SDN controller plans to keep the same routes in future time-steps, then it adopts the capacity constraints in (2). In this toy example, it is easy to see that the corresponding max-min fair admission rates are $z_{1,t} = z_{2,t} = 2/3$. Notice that there exists feasible configurations $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ with higher sum $\sum_{n=1}^{2} z_{n,t}$, but their admission rates are not max-min fair. An example of such unfair feasible admission rates are $z_{1,t} = 0.5$, $z_{2,t} = 1$.

Plan to re-route. Alternatively, if the SDN controller plans to re-route in the next time-step, then it adopts the capacity constraints in (5) with $s_{min} = 0.05$. It is easy to see that the corresponding max-min fair admission rates are $z_{1,t} = z_{2,t} = 2/3 * (1 - 0.05)$.

Two important observations about re-routing plans. (i) Planning to re-route at time $t+1$ does not guarantee that the SDN controller will be able to re-route at time $t+1$. In particular, if the capacity prediction is inaccurate and (by chance) $\hat{c}_{t+1}^{(k,l)} > c_{t+1}^{(k,l)}$, the SDN controller may not have enough scratch capacity at time $t+1$ to re-route. In this case, the SDN controller is only allowed to adjust the admitted demand. (ii) Planning to re-route at time $t+1$, can only cause performance degradation at the current time t, due to the provision of the scratch capacity. The potential benefits of planning to re-route at time $t+1$ can only be assessed by computing the performance of the network at future time-steps.

4.3 Optimal Sequence of Configurations

In this section, we develop the MSNR algorithm which leverages information about current and future predicted link capacities $\{c_t^{(k,l)}, \hat{c}_{t+1}^{(k,l)}, \ldots, \hat{c}_{t+H}^{(k,l)}\}$ to dynamically optimize routing and admission control decisions aiming to maximize the cumulative sum of admission rates $\sum_{t=1}^{T}\sum_{n=1}^{N-1} z_{n,t}$, while ensuring that, *in every time-step t*, the selected feasible set $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ is max-min fair and can be implemented by the SDN controller without inducing transient congestion. An important feature of the MSNR algorithm is that it guarantees (as shown in Proposition 4) that the selected network configuration is always *max-min fair*.

Before describing the MSNR algorithm, we introduce the concept of a re-routing plan. For a given time t and a prediction window size H, let $r_{t,h}$ be an indicator function that is equal to 1, if the plan is to re-route in time-step $t+h$, $\forall h \in \{0,1,\ldots,H\}$, and $r_{t,h} = 0$, otherwise. The *re-routing plan* at time t is given by the vector $\mathbf{r_t} = (r_{t,0}, r_{t,1}, \ldots, r_{t,H})$. Notice that if $s_t < s_{min}$, then $r_{t,0} = 0$ and if $s_t \geq s_{min}$, then $r_{t,0} \in \{0,1\}$. The capacity constraint associated with link $(k,l) \in E$ at time-step $t+h$ depends on whether the plan is to re-route at time-step $t+h+1$ or not, and is represented by

$$\sum_{n=1}^{N-1} z_{n,t+h} d_n f_{n,t+h}^{(k,l)} \leq \min\{\hat{c}_{t+h}^{(k,l)}, (1 - s_{min} r_{t,h+1}) \hat{c}_{t+h+1}^{(k,l)}\}\,, \tag{6}$$

which is a generalization of (2) and (5).

The optimization problem associated with a given re-routing plan $\mathbf{r_t}$ can be subdivided at the re-routing times (i.e., times $t+h$ in which $r_{t,h}=1$) *without loss of optimality*. Let $\{t+h_1, t+h_1+1, \ldots, t+h_2\}$ represent a subdivision of a re-routing plan $\mathbf{r_t}$. The Generalized-MCF (G-MCF) algorithm described in Algorithm 1 jointly optimizes the routing decisions $f_{n,t+h_1}^{(k,l)}, \forall n, \forall (k,l)$ at the initial time $t+h_1$ and the admission rates $z_{n,t+h}, \forall n, \forall h \in \{h_1, \ldots, h_2\}$. Notice that routes remain fixed within a subdivision. To address this joint optimization, G-MCF solves a sequence of MCF problems with increasing admission rates $z_{n,t+h}$ until all commodities in the network become saturated. To compute the optimal routing and admission control decisions *associated with an entire re-routing plan* $\mathbf{r_t}$, the G-MCF algorithm is employed in each of its subdivisions.

Algorithm 1: Generalized-MCF (G-MCF) algorithm

1 **Input**: subdivision under consideration $\{t+h_1, t+h_1+1, \ldots, t+h_2\}$, re-routing plan $\mathbf{r_t}$, network topology $G=(V,E)$, demands d_n, current link capacities $c_t^{(k,l)}$, and predicted link capacities $\hat{c}_{t+h}^{(k,l)}$;
2 **Initialization**: iteration $k=0$, unsaturated commodities $U=\{(n,h)\}, \forall (n,h)$, and admission rates that saturate each commodity $z_{(n,h)}^S = 0, \forall n \in \{1,\ldots,N-1\}, \forall h \in \{h_1,\ldots,h_2\}$;
3 **while** $U \neq \emptyset$ **do**
4 | % Next, we find $\bar{z}$ that solves the joint optimization described in Line 7;
5 | **for** $n \in \{1,\ldots,N-1\}$ *and* $h \in \{h_1,\ldots,h_2\}$ **do**
6 | | **if** $(n,h) \in U$ **then** $z_{n,t+h} \leftarrow \bar{z}$; **else** $z_{n,t+h} \leftarrow z_{(n,h)}^S$;
7 | Solve: max $\bar{z}$, s.t. $\bar{z} \in [0,1]$ and constraints (1), (6), $\forall h$;
8 | Output 1: values of $\bar{z}^*$ and $f_{n,t+h}^{(k,l)}$;
9 | % Next, we identify the new saturated commodities (n,h);
10 | Determine the set D of disconnected commodities in the residual graph associated with Output 1;
11 | *Saturation Flag* $= \emptyset$;
12 | **for** $(n,h) \in D$ **do**
13 | | % Next, we find $\bar{z}_{(n,h)}$ that solves the joint optimization described in Line 16;
14 | | Assign: $z_{n,t+h} \leftarrow \bar{z}_{(n,h)}$;
15 | | **for** $(m,j) \in U \setminus (n,h)$ **do** $z_{m,t+j} \leftarrow \bar{z}^*$;
16 | | Solve: max $\bar{z}_{(n,h)}$, s.t. $\bar{z}_{(n,h)} \in [0,1]$ and (1), (6), $\forall h$;
17 | | Output 2: values of $\bar{z}_{(n,h)}^*$ and $f_{n,t+h}^{(k,l)}$;
18 | | **if** $\bar{z}_{(n,h)}^* = \bar{z}^*$ **then**
19 | | | *Saturation Flag* $\leftarrow$ *Saturation Flag* $\cup (n,h)$;
20 | **for** $(n,h) \in$ *Saturation Flag* **do**
21 | | Assign: $z_{(n,h)}^S \leftarrow \bar{z}^*$ and $U \leftarrow U \setminus (n,h)$;
22 | $k \leftarrow k+1$
23 % Next, we find the max-min fair feasible configuration;
24 **for** $n \in \{1,\ldots,N-1\}$ *and* $h \in \{h_1,\ldots,h_2\}$ **do**
25 | Assign: $z_{n,t+h} \leftarrow z_{(n,h)}^S$;
26 Obtain: $f_{n,t+h}^{(k,l)}$ that satisfy (1) and (6) for $h \in \{h_1,\ldots,h_2\}$;
27 Output 3: values of $z_{n,t+h} = z_{(n,h)}^S$ and $f_{n,t+h}^{(k,l)}$;

MSNR algorithm. To find the optimal sequence of network configurations *at time-step* t, the MSNR algorithm selects the plan $\mathbf{r_t^*}$ with highest cumulative sum of admission rates $\sum_{h=0}^{H-1}\sum_{n=1}^{N-1} z_{n,t+h}$. A naive implementation of the MSNR algorithm computes and compares the performance of the (at least) 2^H admissible re-routing plans. To reduce the computational complexity from exponential

$O(2^H)$ to polynomial $O(H^4)$, we develop a principled implementation of the MSNR algorithm which leverages the fact that the G-MCF algorithm optimizes subdivisions of the re-routing plan $\mathbf{r_t}$ separately. This principled implementation based on backward induction is described next.

First Case. Consider the space of plans $\mathbf{r_t}$ that re-route *for the first time* at step $t+H-1$, i.e., $\mathbf{r_t} \in \{(0,\ldots,0,1,0),(0,\ldots,0,1,1)\}$. The MSNR algorithm employs the G-MCF algorithm to compute the optimal routing and admission control decisions for these 2 re-routing plans and then selects the plan $\mathbf{r_t^{(1)}}$ with highest cumulative sum of admission rates at time $t+H-1$, namely $\sum_{n=1}^{N-1} z_{n,t+H-1}$.

Second Case. Consider the space of plans that re-route for the first time at step $t+H-2$, i.e., $\mathbf{r_t} \in \{(0,\ldots,0,1,0,0),(0,\ldots,0,1,0,1),(0,\ldots,0,1,1,0),(0,\ldots,0,1,1,1)\}$. Notice that in the *subspace of plans* that re-route both at times $t+H-2$ and $t+H-1$, we know from the First Case that $\mathbf{r_t} = \mathbf{r_t^{(1)}} + (0,\ldots,0,1,0,0)$ has the best performance and, hence, all other plans in this particular subspace can be excluded from consideration. The MSNR algorithm computes the optimal routing and admission control decisions for the remaining 3 re-routing plans and selects the plan $\mathbf{r_t^{(2)}}$ with highest cumulative sum of admission rates at times $t+H-2$ and $t+H-1$, namely $\sum_{h=H-2}^{H-1}\sum_{n=1}^{N-1} z_{n,t+h}$.

Third Case. Consider the space of plans that re-route for the first time at step $t+H-3$, i.e., $\mathbf{r_t} \in \{(0,\ldots,0,1,0,0,0),\ldots,(0,\ldots,0,1,1,1,0),(0,\ldots,0,1,1,1,1)\}$. Notice that in the *subspace of plans* that re-route both at times $t+H-3$ and $t+H-2$, we know from the Second Case that $\mathbf{r_t} = \mathbf{r_t^{(2)}} + (0,\ldots,0,1,0,0,0)$ has the best performance and, hence, all other plans in this particular subspace can be excluded from consideration. Similarly, in the *subspace of plans* that re-route both at times $t+H-3$ and $t+H-1$, but do not re-route at time $t+H-2$, we know from the First Case that $\mathbf{r_t} = \mathbf{r_t^{(1)}} + (0,\ldots,0,1,0,0,0)$ has the best performance and, hence, all other plans in this particular subspace can be excluded from consideration. The MSNR algorithm computes the optimal routing and admission control decisions for the remaining 4 re-routing plans and selects the plan $\mathbf{r_t^{(3)}}$ with highest cumulative sum of admission rates from times $t+H-3$ to $t+H-1$, namely $\sum_{h=H-3}^{H-1}\sum_{n=1}^{N-1} z_{n,t+h}$.

Subsequent Cases. The MSNR algorithm considers the space of plans that re-route for the first time at steps $t+H-4$, $t+H-5$, ..., t and employs an analogous procedure in order to determine the best plans $\mathbf{r_t^{(4)}}$, $\mathbf{r_t^{(5)}}$, ..., $\mathbf{r_t^{(H)}}$.

Last Case. The MSNR algorithm compares the performance of the best plans $\mathbf{r_t^{(h)}}, \forall h \in \{1,2,\cdots,H\}$ with the performance of the plans $(0,\ldots,0,0)$ and $(0,\ldots,0,1)$, and then selects the plan $\mathbf{r_t^*}$ with highest cumulative sum of admission rates $\sum_{h=0}^{H-1}\sum_{n=1}^{N-1} z_{n,t+h}$ in the entire prediction window. The routing and admission control decisions associated with $\mathbf{r_t^*}$ are the optimal sequence of network configurations.

Remark 3 (Computational Complexity). *To find the best plans $\mathbf{r_t^{(1)}}$, $\mathbf{r_t^{(2)}}$, ..., $\mathbf{r_t^{(H)}}$ in each of the corresponding backward induction cases, the MSNR algorithm computes and compares the performance of $2,3,\ldots,H+1$ re-routing plans, respectively. Then, in the last case of the induction, the MSNR algorithm computes and compares the performance of $H+2$ re-routing plans in order to find the plan $\mathbf{r_t^*}$ and the associated optimal sequence of network configurations $\{f_{n,t+h}^{(k,l)}, z_{n,t+h}\}, \forall h \in \{0,1,\ldots,H-1\}$, at time t. In total, the MSNR algorithm employing backward induction computes the performance of $(H+1)(H+4)/2$ re-routing plans, as opposed to the (at least) 2^H computations associated with the naive implementation. Notice from Algorithm 1 that to compute the performance of any given re-routing plan $\mathbf{r_t}$, the G-MCF algorithm solves $O(H^2N^2)$ MCF optimization problems, each of which can be solved in polynomial time [2, 49]. It follows that the MSNR algorithm has polynomial computational complexity which grows as $O(H^4)$.*

Proposition 4 (Max-Min Fairness of the MSNR algorithm). *The optimal sequence of network configurations* $\{f_{n,t+h}^{(k,l)}, z_{n,t+h}\}$ *given by the MSNR algorithm has admission rates* $\{z_{n,t+h}\}_{n=1}^{N-1}$ *that are max-min fair in every time-step* $t+h$ *and for any given* $h \in \{0,1,\ldots,H-1\}$*, irrespective of the topology* $G=(V,E)$*, demands* d_n*, and current and predicted link capacities* $\{c_t^{(k,l)}, \hat{c}_{t+1}^{(k,l)}, \ldots, \hat{c}_{t+H}^{(k,l)}\}$.

Proof. The complete proof is in Appendix A. □

5 PERFORMANCE EVALUATION

In this section, we evaluate the performance of the PNR framework. In particular, in Sec. 5.1 we evaluate the prediction accuracy of the AP mechanism and compare it with two benchmark time series prediction methods. Then, in Sec. 5.2, we evaluate the performance of the MSNR algorithm and compare it with two reactive algorithms using a small network with $N=3$ nodes, synthetically generated attenuation levels $x_t^{(k,l)}$ and synthetically generated attenuation predictions $\hat{x}_{t+h}^{(k,l)}$ with tunable prediction accuracies. The goal is to gain insight from this small and controllable network. Finally, in Sec. 5.3, we evaluate the PNR framework (with both the AP mechanism and the MSNR algorithm) using the dataset from the backhaul network with $N=13$ nodes, illustrated in Fig. 1.

5.1 Evaluation of the AP mechanism

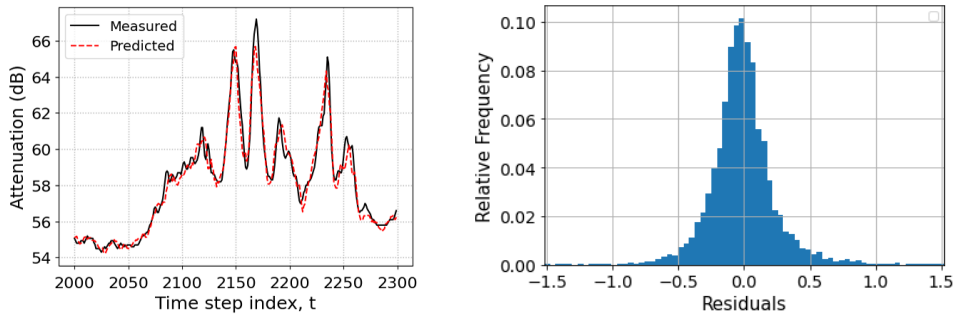
The prediction accuracy of the AP mechanism is evaluated using the test sequences of attenuation measurements described in Sec. 2. In this section, we show the results associated with *Test Seq. I* and *Test Seq. II*, both of which include a period of rain. We first assess the prediction error of a given link, then we analyze the prediction RMSE of the entire network and, finally, we assess the empirical probability of large prediction errors. Additional evaluation, including results associated with different test sequences and different links, are shown in Appendix B and in [30].

Let $e_{t,h}^{(k,l)} = x_{t+h}^{(k,l)} - \hat{x}_{t+h}^{(k,l)}$ be the *h-steps-ahead prediction error* associated with link (k,l) at time t. In Fig. 6(a), we compare the evolution of the attenuation measurements $x_{t+3}^{(9,13)}$ from link $(9,13)$ with the 3-steps-ahead predictions $\hat{x}_{t+3}^{(9,13)}$ generated by the AP mechanism during an interval of 300 time-steps from *Test Seq. I*. In Fig. 6(b), we display the relative frequency distribution of the 3-steps-ahead prediction error $e_{t,3}^{(9,13)}$ from link $(9,13)$ associated with the entire *Test Seq. I*. The results in Fig. 6 suggest that: (i) the attenuation predictions accurately track the measurements; (ii) the distribution of the prediction error $e_{t,h}^{(k,l)}$ is similar to a normal distribution with zero mean; and (iii) at high measured attenuation levels, predictions tend to (slightly) underestimate the attenuation. This underestimation is further assessed in Appendix B.

Weather-induced attenuation varies over time and location, and also depends on links' characteristics such as frequency, polarization, and length. Therefore, prediction errors may differ considerably across different links. To capture the prediction error in the entire network, we employ

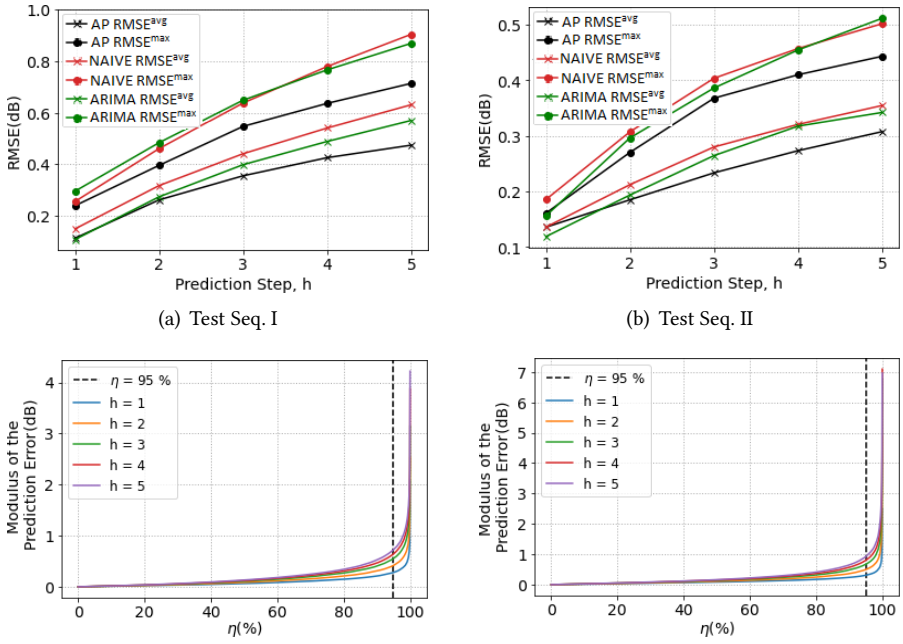
$$RMSE_h^{\text{avg}} = \sqrt{\frac{1}{T-H}\sum_{t=1}^{T-H}\frac{1}{|E|}\sum_{(k,l)\in E}\left(e_{t,h}^{(k,l)}\right)^2}\ ,\quad RMSE_h^{\max} = \sqrt{\frac{1}{T-H}\sum_{t=1}^{T-H}\max_{(k,l)\in E}\left\{\left(e_{t,h}^{(k,l)}\right)^2\right\}} \tag{7}$$

which calculate the RMSE associated with the h-steps-ahead prediction errors of all links over the entire time-horizon and the RMSE associated with the *largest* h-steps-ahead prediction error among all the links in each time-step t, respectively. In Fig. 7(a)-(b), we display the $RMSE_h^{\text{avg}}$ and $RMSE_h^{\max}$ (in dB) as a function of the prediction horizon $h \in \{1,\ldots,H\}$ for *Test Seq. I* and *II* and for three prediction mechanisms: (i) the AP mechanism; (ii) the naive AP method, also called random-walk method, which is a commonly used benchmark [20] that employs the latest measurement as future predictions, i.e., $\hat{x}_{t+h}^{(k,l)}(naive) = x_t^{(k,l)}, \forall h$; and (iii) the ARIMA model, which is a well-known time



(a) (b)

Fig. 6. (a) Comparison of the attenuation from link (9, 13) with the 3-steps-ahead predictions. (b) Relative frequency distribution of the 3-steps-ahead prediction error.



(a) Test Seq. I (b) Test Seq. II

(c) AP mechanism, Test Seq. I (d) ARIMA model, Test Seq. I

Fig. 7. (a)-(b): $RMSE_h^{\text{avg}}$ and $RMSE_h^{\text{max}}$ of the prediction error for different horizons h and for the AP mechanism, naive AP method, and ARIMA model. (c)-(d): Percentile for the modulus of the prediction error.

series prediction model [44]. *It is important to emphasize that both benchmark methods (i.e., naive and ARIMA) consider each link in isolation when predicting future attenuation levels and, thus, they do not capture the spatial correlation that is typical of weather-induced attenuation.* The results in Fig. 7(a)-(b) suggest that the AP mechanism outperforms the benchmark methods in both *Test Seqs. I* and *II* and that this performance improvement increases as the prediction horizon h increases.

When the network condition is steady (e.g., during a dry period or when the rain intensity does not vary significantly) we expect the accuracy of any sound prediction mechanism to be high. Therefore, *time-average improvements* such as the ones displayed in Fig 7(a)-(b) can seem modest. To capture the improvement of AP in critical moments, we analyze the empirical probability of *large prediction errors* by computing the percentile associated with the modulus of the h-steps-ahead predictions errors $|e_{t,h}^{(k,l)}|$. In particular, for a given test sequence with measurements $x_{t+h}^{(k,l)}$ and associated attenuation predictions $\hat{x}_{t+h}^{(k,l)}$, the η th percentile value represents the lowest $|e_{t,h}^{(k,l)}|$ that is larger than or equal to η% of all the values of $|e_{t,h}^{(k,l)}|$ in the considered dataset. In Figs. 7(c) and (d), we show the percentile values for different prediction horizons $h \in \{1, \ldots, 5\}$ for the *AP mechanism* and for the *ARIMA model*, respectively. The results in Figs. 7(c) and (d) suggest that, as expected, the percentile values increase with the prediction horizon h. More importantly, the results also show that 99.9% of the one-step-ahead errors for ARIMA and AP are lower than 0.96 dB and 0.84 dB, respectively, and the 5-step-ahead errors for ARIMA and AP are lower than 3.26 dB 2.39 dB, respectively, highlighting the significant improvement of AP over ARIMA, which can be attributed at least in part to the capability of the AP mechanism of capturing the spatial correlation of weather-induced attenuation.

In summary, the results in Figs. 6 and 7 show that *the AP mechanism predicts future link attenuation with high accuracy* even when network's conditions change abruptly. This high prediction accuracy is what allows us to use the predicted link capacities as input to the MSNR algorithm. The impact of the prediction accuracy on the performance of the MSNR algorithm is discussed next.

5.2 Evaluation of reconfiguration algorithms in a small network

We compare the performance of the MSNR algorithm with that of two reactive network reconfiguration algorithms in a small network with $N = 3$ nodes, synthetically generated link attenuation levels, and tunable accuracy of attenuation predictions. The goal is to leverage the small tunable network to gain insight into the behavior of the different reconfiguration algorithms. The drawback is that the limited re-routing options restrict the performance benefit of the reconfiguration algorithm. To evaluate the performance benefit in a large network, in the next section we consider the city-scale backhaul network.

The reconfiguration algorithms are evaluated in terms of their network utilization, which is captured by the evolution of the node-average admission rate $\sum_{n=1}^{N-1} z_{n,t}/(N-1)$ over time t. The considered reconfiguration algorithms are:

i) *MSNR algorithm*: as described in Sec. 4.3.
ii) *Never re-route algorithm*: attempts to maximize the admission rates $z_{n,t}$ by never provisioning scratch capacity and, thus, fully utilizing links whenever possible. Under this algorithm, the SDN controller is rarely[6] allowed to re-route, but it continuously optimizes the admission rates.
iii) *Always re-route algorithm*: attempts to provision scratch capacity $s_t \geq s_{min} = 0.05$ at every time-step t, allowing the SDN controller to optimize routing decisions often.

All three network reconfiguration algorithms select max-min fair admission rates $z_{n,t}$ *at every time-step* t. The main difference is that only the MSNR algorithm employs the predictions of the links' future conditions to decide when to re-route. Both the Never re-route and Always re-route algorithms simply react to the time-varying conditions of the network. Unfortunately, a comparison with the predictive SDN-based routing framework developed in [59] is impractical

[6]If the predicted capacities $\hat{c}_{t+1}^{(k,l)}$ are inaccurate, in particular if $\hat{c}_{t+1}^{(k,l)} < c_{t+1}^{(k,l)}$, it may happen that $s_{t+1} \geq s_{min}$ and the Never re-route algorithm is allowed to re-route at time $t+1$.

due to the incompatible assumptions. The framework in [59] can *only* be employed during periods of rain, it allows flows to temporarily exceed the link capacity, and it does not take fairness into account.

The results in this section are associated with the small network with $N = 3$ nodes and three links $\{(1,2),(2,3),(1,3)\}$ illustrated in Fig. 8(a). The *normalized*[7] demands associated with nodes 1 and 2 remain fixed at $d_1 = 1$ and $d_2 = 0.5$, respectively, during the time-horizon of 1,000 time-steps. The attenuation levels $x_t^{(k,l)}$ and predicted attenuation levels $\hat{x}_{t+h}^{(k,l)}$ are synthetically generated according to the following stochastic processes

$$x_t^{(k,l)} = \min\{\max\{x_{t-1}^{(k,l)} + \delta_t^{(k,l)}; -100\}; -50\}\ ; \tag{8}$$

$$\hat{x}_{t+h}^{(k,l)} = \min\{\max\{x_{t+h}^{(k,l)} + \tilde{\delta}_{t,h}^{(k,l)}; -100\}; -50\}\ , \tag{9}$$

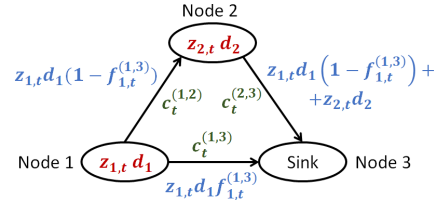
for all links $(k,l) \in E$, for all time-steps $t \in \{1,\ldots,1,000\}$, for all values of $h \in \{1,\ldots,H\}$, and with $x_0^{(k,l)}$ sampled from a uniform distribution in the interval $(-100,-50)$. Notice that (8) establishes the variation of the attenuation $x_t^{(k,l)}$ over time, while (9) establishes the noise in the prediction $\hat{x}_{t+h}^{(k,l)}$ of the future attenuation $x_{t+h}^{(k,l)}$. The sequence of Gaussian random variables $\delta_t^{(k,l)}$ is i.i.d. over time t, independent across links, and sampled according to $\mathcal{N}(0, 6.25)$. Similarly, the sequence of random variables $\tilde{\delta}_{t,h}^{(k,l)}$ are Gaussian $\mathcal{N}(0,\tilde{\sigma}^2)$ with positive variance $\tilde{\sigma}^2$, i.i.d. over time, and independent across different links. Notice from (9) that, a high variance $\tilde{\sigma}^2$ represents an AP mechanism with poor accuracy, i.e. large prediction error. The choice of Gaussian distribution for $\tilde{\delta}_{t,h}^{(k,l)}$ was inspired by the relative frequency distribution of the prediction error shown in Fig. 6(b).

To determine the capacities $c_t^{(k,l)}$ and the predicted capacities $\hat{c}_{t+h}^{(k,l)}$ associated with the synthetic values of $x_t^{(k,l)}$ and $\hat{x}_{t+h}^{(k,l)}$, respectively, we adopt a constant transmission signal level of $P_{Tx,t}^{(k,l)} = 0$ dBm and use the AM mechanism described in Sec. 4.1. In Fig. 8(b), we display the evolution of the *normalized* values of $c_t^{(k,l)}$ employed to obtain the results in this section. Notice that this is a network with highly dynamic link capacities $c_t^{(k,l)}$.

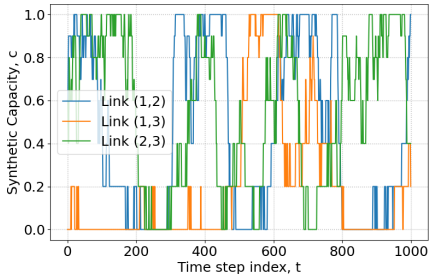
In Fig. 8(c), we compare the evolution of the node-average admission rate $(z_{1,t} + z_{2,t})/2$ over time t for different reconfiguration algorithms operating with ideal attenuation predictions, i.e., with $\hat{x}_{t+h}^{(k,l)} = x_{t+h}^{(k,l)}$ and, as a result, $\hat{c}_{t+h}^{(k,l)} = c_{t+h}^{(k,l)}$. In Fig. 8(d) and in Table 2, we show the time-average admission rates $\sum_{t=1}^{T}(z_{1,t}+z_{2,t})/2T$ for different reconfiguration algorithms operating with attenuation predictions with different accuracies $\tilde{\sigma}^2 \in \{0, 0.0025, 0.25, 1, 4, 9, 25\}$ and different prediction window sizes $H \in \{2,3,4,5\}$.

The results in Fig. 8(c) show that, as expected, Never re-route has the worse performance, while MSNR with prediction window size $H = 5$ has the best performance in terms of network utilization. The poor performance of Never re-route, especially between time-steps 500 and 800, results from the SDN controller not being allowed to re-route. The lower performance of Always re-route when compared to MSNR is due to the frequent provisioning of scratch capacity $s_{min} = 0.05$. By leveraging the prediction of links' future conditions, MSNR can assess the potential future benefits of re-routing[8], which allows it to choose when is the best time to re-route. Throughout the 1,000 time-steps, the SDN controller re-routes 31, 30, 28, and 29 times when employing MSNR with prediction window sizes $H \in \{2,3,4,5\}$, respectively.

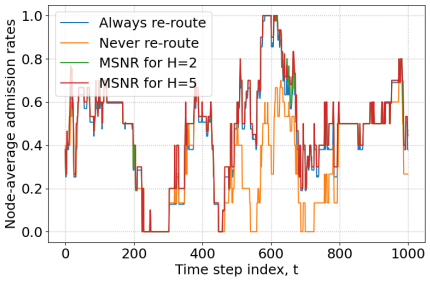
[7] Both demands and capacities are normalized with respect to the maximum achievable bitrate of 225 Mbps from Table 1.
[8] Recall from the discussion in Sec. 4.2 that planning to re-route at the next time-step $t+1$, can only degrade the network performance at the current time t due to the provisioning of the scratch capacity.



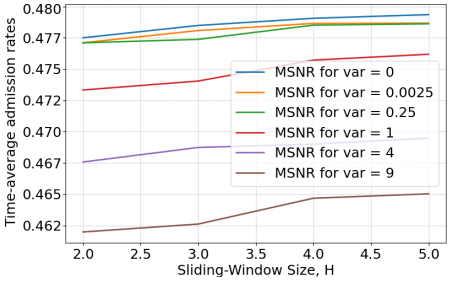
(a) Small network with $N = 3$ nodes



(b) Normalized link capacity $c_t^{(k,l)}$



(c) Node-average rate $(z_{1,t} + z_{2,t})/2$



(d) Time-average $\sum_{t=1}^{T}(z_{1,t} + z_{2,t})/2T$

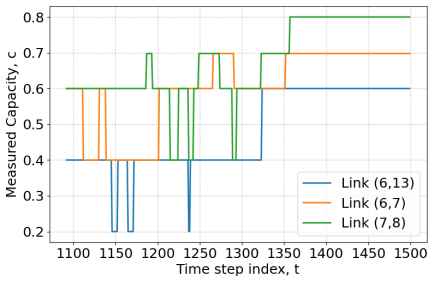
Fig. 8. Performance of the MSNR algorithm for a small network with $N = 3$ nodes. (a) Small network with two commodities and a destination. The admitted demands $z_{n,t}d_n$ at time-step t are shown within the corresponding nodes. The total flows and capacities at time-step t are shown next to the corresponding links. (b) Evolution of the normalized link capacity $c_t^{(k,l)}$ over time. (c) Node-average admission rate $(z_{1,t} + z_{2,t})/2$ for different reconfiguration algorithms with ideal attenuation prediction ($\tilde{\sigma}^2 = 0$). (d) Time-average admission rate $\sum_{t=1}^{T}(z_{1,t} + z_{2,t})/2T$ for MSNR with different prediction window sizes $H \in \{2, 3, 4, 5\}$ and attenuation prediction accuracies $\tilde{\sigma}^2 \in \{0, 0.0025, \ldots, 9\}$.

The results in Fig 8(d) and Table 2 suggest that: (i) The performance of MSNR improves[9] as the prediction accuracy improves and as the window size H increases. Intuitively, this is because accurate information about future network conditions allows the MSNR algorithm to better prepare the network for imminent disturbances. (ii) The performance gain of improving the prediction accuracy is more significant than the performance gain of increasing the prediction window size H. This is because the MSNR algorithm uses predicted network conditions as the ground-truth. This design choice was made based on the high accuracy of the AP mechanism[10].

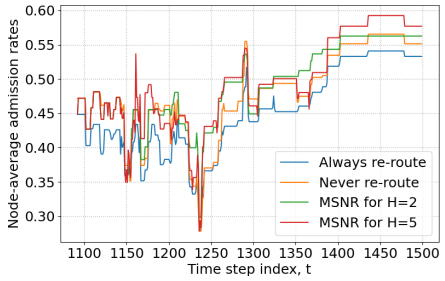
5.3 Evaluation of the PNR Framework with real-world network data

We now evaluate the performance of the PNR framework using the data collected from the backhaul network in Fig. 1 with $N = 13$ BSs (12 commodities and one destination) and 17 links. The normalized demands assigned to the commodities are chosen according to a uniform distribution in the interval $(0, 2)$. In particular, the twelve demand values[11] are $\boldsymbol{d} = [1.111, 0.557, 1.124, 1.266, 0.174, 1.485, 0.947,$

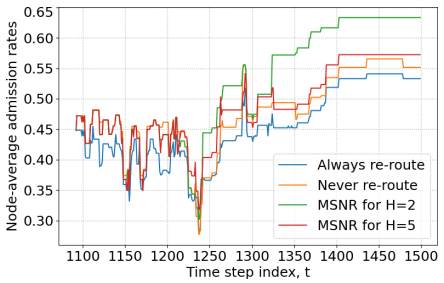
[9]Notice that a larger network with more re-routing options could have shown a more substantial performance improvement.
[10]In case low prediction accuracy is detected, a viable alternative is for the SDR to fall back on *reactive* network reconfiguration algorithms such as the one proposed by NEC in [40] that do not rely on future predictions.
[11]Notice that similar results can be obtained for different vectors of demands.



(a) Normalized link capacity $c_t^{(k,l)}$



(b) Ideal attenuation prediction



(c) Prediction from the AP mechanism

Fig. 9. Performance of the PNR framework using data collected from the backhaul network in Fig. 1. (a) Evolution of the normalized measured link capacity $c_t^{(k,l)}$ over time. (b) Evolution of the node-average admission rate $\sum_{n=1}^{N-1} z_{n,t}/(N-1)$ over time for different network reconfiguration algorithms with ideal attenuation prediction. (c) Evolution of $\sum_{n=1}^{N-1} z_{n,t}/(N-1)$ over time for different network reconfiguration algorithms using predictions from the AP mechanism.

Table 2. Time-average admission rates $\sum_{t=1}^{T}(z_{1,t}+z_{2,t})/2T$ for different network reconfiguration algorithms and for attenuation predictions with different accuracies.

Prediction accuracy	Ideal	$\tilde{\sigma}^2 = 1$	$\tilde{\sigma}^2 = 25$
Never re-route	0.362	0.359	0.343
Always re-route	0.453	0.449	0.427
MSNR for $H = 2$	0.477	0.473	0.447
MSNR for $H = 3$	0.478	0.474	0.449
MSNR for $H = 4$	0.479	0.476	0.449
MSNR for $H = 5$	0.479	0.476	0.450

0.067, 0.140, 0.596, 1.413, 0.999]. The values of the (actual) capacities $c_t^{(k,l)}$ and future predicted capacities $\hat{c}_{t+h}^{(k,l)}$ are determined by the link attenuation measurements in the dataset, by the AM mechanism described in Sec. 4.1, and by the AP mechanism. To assess the performance of the PNR framework in a *challenging scenario*, we choose a sequence of more than 400 measurements (from *Test Seq. I* described in Sec. 2) that *includes a period with high attenuation variability due to a rain event*. Moreover, we consider transmission signal levels $P_{Tx,t}^{(k,l)}$ that are 10 dBm lower than

Table 3. Performance gain of MSNR with $H \in \{2, 5\}$ when compared to a reactive algorithm: Never re-route or Always re-route. The performance gain metrics in columns 3, 4, and 5 are defined in (10).

MSNR	Reactive	Time-aver.	Node-aver.	Instant.
$H = 5$	Always	7.74%	18.00%	170.19%
$H = 2$	Always	15.49%	26.84%	263.58%
$H = 5$	Never	1.67%	10.37%	68.37%
$H = 2$	Never	8.98%	22.04%	208.01%

the dataset measurements. In Fig. 9(a), we display the evolution of the normalized capacities $c_t^{(k,l)}$ from three selected links. Notice that the variation is significant. In Fig. 9(b), we show the evolution of the node-average admission rates $\sum_{n=1}^{N-1} z_{n,t}/(N-1)$ for different reconfiguration algorithms employing ideal attenuation prediction, i.e., $\hat{c}_{t+h}^{(k,l)} = c_{t+h}^{(k,l)}$. In Fig. 9(c), we display the node-average admission rates for algorithms employing the AP mechanism to predict $\hat{c}_{t+h}^{(k,l)}$ over time. The results in Figs. 9(b) and 9(c) show that MSNR outperforms both Never re-route and Always re-route.

In Table 3, we display the performance gain of MSNR with $H \in \{2, 5\}$ employing the AP mechanism when compared to reactive algorithms: Never re-route or Always re-route. Let $z_{n,t}^{(M,R)} = z_{n,t}^{(M)} - z_{n,t}^{(R)}$ be the difference between the admission rates associated with MSNR and the reactive algorithm. The third, fourth, and fifth columns of Table 3 are associated with

$$\frac{\sum_{t=1}^{T}\sum_{n=1}^{N-1} z_{n,t}^{(M,R)}}{\sum_{t=1}^{T}\sum_{n=1}^{N-1} z_{n,t}^{(R)}} \quad , \quad \max_{t}\left\{\frac{\sum_{n=1}^{N-1} z_{n,t}^{(M,R)}}{\sum_{n=1}^{N-1} z_{n,t}^{(R)}}\right\} \quad , \quad \max_{n,t}\left\{\frac{z_{n,t}^{(M,R)}}{z_{n,t}^{(R)}}\right\}; \tag{10}$$

which are the *time-average performance gain*, *maximum node-average performance gain*, and *maximum instantaneous performance gain*, respectively. The results in Table 3 show that the MSNR algorithm can improve the time-average admission rate $\sum_{t=1}^{T}(z_{1,t} + z_{2,t})/2T$ by more than 7% when compared to either Always re-route or Never re-route. More importantly, they also show that the **gain in terms of the instantaneous per commodity admission rate $z_{n,t}$ can exceed** 200%. These significant instantaneous gains occur during severe rain-induced attenuation events, thus suggesting that the *PNR framework can indeed alleviate the impact of abrupt disturbances on the network performance, which can be paramount to time-sensitive applications.*

An important observation from the results in Secs. 5.2 and 5.3 is that, when the AP mechanism has high accuracy, the performance gap between MSNR with $H = 2$ and reactive algorithms is significantly larger than the performance gain obtained from increasing the prediction window size H. Adding to this observation the fact that the computational complexity of MSNR grows with H, as discussed in Sec. 4.3, makes the PNR framework with $H = 2$ an attractive choice both in terms of performance and complexity.

6 CONCLUSION

We developed the PNR framework that has two components. The first is the AP mechanism that uses historical data to predict the sequence of future attenuation levels, *without incorporating any specific weather-related models.* The second is the MSNR algorithm that dynamically optimize routing $f_{n,t}^{(k,l)}$ and admission control $z_{n,t}$ decisions over time aiming to maximize the cumulative sum of admission rates $\sum_{t=1}^{T}\sum_{n=1}^{N-1} z_{n,t}$. The MSNR algorithm guarantees that the selected feasible set $\{f_{n,t}^{(k,l)}, z_{n,t}\}$ is max-min fair in every time-step t and can be implemented without inducing transient congestion. We used a real-world dataset to thoroughly evaluate the PNR framework and

to show that it allows the SDN controllers to prepare the x-haul for imminent (and possibly severe) weather-induced disturbances.

An interesting extension of this work is the introduction of a mechanism to predict time-varying traffic demands d_n. It is worth emphasizing that (i) time-varying demands can be easily accounted for using the network model proposed in Sec. 2; and that (ii) the prediction uncertainty of future demands would mainly affect the capacity constraint (2) of the network reconfiguration problem, similarly to the prediction uncertainty of link attenuation levels. An important challenge is obtaining compatible data for both high frequency link attenuation and traffic demands.

There are several additional open problems that will be considered in our future work, including consideration of downlink/uplink traffic, consideration of alternative approaches for predicting link attenuation (e.g., Transformer models), consideration of Reinforcement Learning for dynamic network reconfiguration with fairness guarantees, application to 5G slice admission and provisioning, and experimental evaluation in city-scale testbeds [41, 46].

ACKNOWLEDGEMENTS

We thank our shepherd Prof. Suzan Bayhan and the anonymous referees for their helpful comments. This research was supported in part by NSF-BSF grant CNS-1910757 and NSF grants CNS-2148128, EEC-2133516.

REFERENCES

[1] N. Abedini and S. Shakkottai. 2014. Content caching and scheduling in wireless networks with elastic and inelastic traffic. *IEEE/ACM Trans. Netw.* 22, 3 (2014), 864–874.
[2] M. Allalouf and Y. Shavitt. 2008. Centralized and distributed algorithms for routing and weighted max-min fair bandwidth allocation. *IEEE Trans. Netw. Service Manag.* 16, 5 (2008), 1015–1024.
[3] J. Andersson, J. Olsson, R. van de Beek, and J. Hansryd. 2022. OpenMRG: Open data from Microwave links, Radar, and Gauges for rainfall quantification in Gothenburg, Sweden. *Earth System Science Data Discussions* (2022), 1–26.
[4] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, and H. Zhang. 2013. Developing a predictive model of quality of experience for Internet video. In *Proc. ACM SIGCOMM.*
[5] L. Bao, J. Hansryd, T. Danielson, G. Sandin, and U. Noser. 2015. Field trial on adaptive modulation of microwave communication link at 6.8 GHz. In *Proc. IEEE EuCAP.*
[6] L. Bao, C. Larsson, M. Mustafa, J. Selin, J. Andersson, J. Hansryd, M. Riedel, and H. Andersson. 2017. A brief description on measurement data from an operational microwave network in Gothenburg, Sweden. In *Proc. CEST.*
[7] D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Perez. 2019. DeepCog: Optimizing resource provisioning in network slicing with AI-based capacity forecasting. *IEEE J. Sel. Areas Commun* 38, 2 (2019), 361–376.
[8] N. Bui, M. Cesana, S Hosseini, Q. Liao, I. Malanchini, and J. Widmer. 2017. A survey of anticipatory mobile networking: Context-based classification, prediction methodologies, and optimization techniques. *IEEE Commun. Surveys Tuts.* 19, 3 (2017), 1790–1821.
[9] N. Bui and J. Widmer. 2015. Mobile network resource optimization under imperfect prediction. In *Proc. IEEE WoWMoM.*
[10] M. Caesar, D. Caldwell, N. Feamster, J. Rexford, A. Shaikh, and J. van der Merwe. 2005. Design and implementation of a routing control platform. In *Proc. NSDI.*
[11] A. Chattopadhyay, B. Blaszczyszyn, and E. Altman. 2018. Location aware opportunistic bandwidth sharing between static and mobile users with stochastic learning in cellular networks. *IEEE Trans. Mobile Comput.* (2018), 1802–1815.
[12] K. Cho and D. Yoon. 2002. On the general BER expression of one- and two-dimensional amplitude modulations. *IEEE Trans. Commun.* 50, 7 (2002), 1074–1080.
[13] A. Doff, K. Chandra, and R. Prasad. 2015. Sensor assisted movement identification and prediction for beamformed 60 GHz links. In *Proc. IEEE CCNC.*
[14] J. Edstam, J. Hansryd, S. Carpenter, T. Emanuelsson, Y. Li, and H. Zirath. 2017. Microwave backhaul revolution - reaching beyond 100 GHz. In *Ericsson Technology Review.*
[15] U. Ehsan, B. Harrison, L. Chan, and M. Riedl. 2018. Rationalization: A neural machine translation approach to generating natural language explanations. In *Proc. AAAI/ACM Conference on AI, Ethics, and Society.*
[16] A. Graves and N. Jaitly. 2014. Towards end-to-end speech recognition with recurrent neural networks. In *Proc. ICML.*
[17] C. Gutterman, E. Grinshpun, S. Sharma, and G. Zussman. 2019. RAN resource usage prediction for a 5G slice broker. In *Proc. ACM MobiHoc.*

[18] C. Hong, S. Kandula, R. Mahajan, M. Zhang, V. Gill, M. Nanduri, and R. Wattenhofer. 2013. Achieving high utilization with software-driven WAN. In *Proc. ACM SIGCOMM*.
[19] L. Huang, S. Zhang, M. Chen, and X. Liu. 2014. When backpressure meets predictive scheduling. In *Proc. ACM MobiHoc*.
[20] R. Hyndman and A. Koehler. 2006. Another look at measures of forecast accuracy. *International Journal of Forecasting* 22, 4 (2006), 679–688.
[21] ITU-R P.530. 2017. Propagation data and prediction methods required for the design of terrestrial line-of-sight systems. *ITU* 530-17 (2017).
[22] ITU-R P.676. 2016. Attenuation by atmospheric gasses. *ITU* 676-11 (2016).
[23] ITU-R P.838. 2005. Specific attenuation model for rain for use in prediction methods. *ITU* 838-3, 1992-1999-2003-2005 (2005).
[24] ITU-R P.840. 2017. Attenuation due to clouds and fog. *ITU* 840-7 (2017).
[25] A. Jabbar, J. Rohrer, A. Oberthaler, E. Cetinkaya, V. Frost, and J. Sterbenz. 2009. Performance comparison of weather disruption-tolerant cross-layer routing algorithms. In *Proc. IEEE INFOCOM*.
[26] D. Jacoby, J. Ostrometzky, and H. Messer. 2020. Short-term prediction of the attenuation in a commercial microwave link using LSTM-based RNN. In *Proc. EUSIPCO*.
[27] D. Jacoby, J. Ostrometzky, and H. Messer. 2021. Adaptive fuzzy-based models for attenuation time series forecasting. In *Proc. COMCAS*.
[28] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, J. Zolla, U. Hölzle, S. Stuart, and A. Vahdat. 2013. B4: Experience with a globally-deployed software defined WAN. In *Proc. ACM SIGCOMM*.
[29] N. Javed, E. Lyons, M. Zink, and T. Wolf. 2013. Adaptive wireless mesh networks: Surviving weather without sensing it. In *Proc. IEEE ICCCN*.
[30] I. Kadota, D. Jacoby, H. Messer, G. Zussman, and J. Ostrometzky. 2022. Switching in the Rain: Predictive Wireless x-haul Network Reconfiguration. *arXiv preprint arXiv:2203.03383* (2022).
[31] D. Kingma and J. Ba. 2015. Adam: A method for stochastic optimization. In *Proc. ICLR*.
[32] B. Lian, Z. Wei, X. Sun, Z. Li, and J. Zhao. 2022. A Review on Rainfall Measurement Based on Commercial Microwave Links in Wireless Cellular Networks. *Sensors* 22, 12 (2022).
[33] H. Liu, X. Wu, M. Zhang, L. Yuan, R. Wattenhofer, and D. Maltz. 2013. zUpdate: Updating data center networks with zero loss. In *Proc. ACM SIGCOMM*.
[34] Z. Lu and G. de Veciana. 2013. Optimizing stored video delivery for mobile networks: The value of knowing the future. In *Proc. IEEE INFOCOM*.
[35] J. Maciejowski. 2002. *Predictive Control With Constraints*. Prentice Hall.
[36] T. Mangla, N. Theera-Ampornpunt, M. Ammar, E. Zegura, and S. Bagchi. 2016. Video through a crystal ball: Effect of bandwidth prediction quality on adaptive streaming in mobile environments. In *Proc. ACM MoVid*.
[37] R. Margolies, A. Sridharan, V. Aggarwal, R. Jana, N. Shankaranarayanan, V. Vaishampayan, and G. Zussman. 2016. Exploiting mobility in proportional fair cellular scheduling: Measurements and algorithms. *IEEE/ACM Trans. Netw.* 24, 1 (2016), 355–367.
[38] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. 2008. OpenFlow: Enabling innovation in campus networks. *SIGCOMM Comput. Commun. Rev.* 38, 2 (2008), 69—-74.
[39] S. Naimi, A. Busson, V. Vèque, L. Slama, and R. Bouallegue. 2014. Anticipation of ETX metric to manage mobility in ad hoc wireless networks. In *Proc. Ad-hoc, Mobile, and Wireless Networks*.
[40] NEC iPASOLINK Series. 2016. Intelligent and high capacity wireless transport solution. https://pdf.aeroexpo.online/pdf/nec-corporation/intelligent-high-capacity-wireless-transport-solution/171108-4848.html
[41] NYC Mesh. 2022. https://www.nycmesh.net/
[42] J. Ostrometzky, G. Zussman, H. Messer, D. Jacoby, and I. Kadota. US Patent Application No. 17/551,643. December, 2021. Predictive weather-aware communication network management. https://patents.google.com/patent/US20220110012A1/en
[43] K. Papagiannaki, N. Taft, Z. Zhang, and C. Diot. 2003. Long-term forecasting of Internet backbone traffic: Observations and initial models. In *Proc. IEEE INFOCOM*.
[44] D. Patel, M. Patel, and D. Patel. 2014. Implementation of ARIMA model to predict rain attenuation for KU-band 12 GHz frequency. *IOSR Journal of Electronics and Communication Engineering* 9, 1 (2014), 83–87.
[45] J. Rak. 2016. A new approach to design of weather disruption-tolerant wireless mesh networks. *Telecommunication Systems* 61, 2 (2016), 311–323.
[46] D. Raychaudhuri, I. Seskar, G. Zussman, T. Korakis, D. Kilper, T. Chen, J. Kolodziejski, M. Sherman, Z. Kostic, X. Gu, H. Krishnaswamy, S. Maheshwari, P. Skrimponis, and C. Gutterman. 2020. Challenge: COSMOS: A City-Scale Programmable Testbed for Experimentation with Advanced Wireless. In *Proc. of ACM MobiCom*.

[47] M. Schleiss and A. Berne. 2010. Identification of dry and rainy periods using telecommunication microwave links. *IEEE Geoscience and Remote Sensing Letters* 7, 3 (2010), 611–615.
[48] M. Shafiq, L. Ji, A. Liu, J. Pang, S. Venkataraman, and J. Wang. 2013. A first look at cellular network performance during crowded events. *SIGMETRICS Perform. Eval. Rev.* 41, 1 (2013), 17–28.
[49] F. Shahrokhi and D. Matula. 1990. The maximum concurrent flow problem. *Journal of the ACM* 37, 2 (1990), 318–334.
[50] J. Sosa, C. Sosa, and B. Paz. 2000. ARIMA models in the rain attenuation prediction in a Mexican tropical area. In *Proc. IEEE AP-S.*
[51] Y. Sun, X. Yin, J. Jiang, V. Sekar, F. Lin, N. Wang, T. Liu, and B. Sinopoli. 2016. CS2P: Improving video bitrate selection and adaptation with data-driven throughput prediction. In *Proc. ACM SIGCOMM.*
[52] S. Sur, X. Zhang, P. Ramanathan, and R. Chandra. 2016. BeamSpy: Enabling robust 60 GHz links under blockage. In *Proc. USENIX NSDI.*
[53] S. Tarsa, M. Comiter, M. Crouse, B. McDanel, and H. Kung. 2015. Taming wireless fluctuations by predictive queuing using a sparse-coding link-state model. In *Proc. ACM MobiHoc.*
[54] X. Tie, A. Seetharam, A. Venkataramani, D. Ganesan, and D. Goeckel. 2011. Anticipatory wireless bitrate control for blocks. In *Proc. ACM CoNEXT.*
[55] X. Wang, Z. Zhou, F. Xiao, K. Xing, Z. Yang, Y. Liu, and C. Peng. 2018. Spatio-temporal analysis and prediction of cellular traffic in metropolis. *IEEE Trans. Mobile Comput.* 18, 9 (2018), 2190–2202.
[56] P. Werbos. 1990. Backpropagation through time: What it does and how to do it. *Proc. IEEE* 78, 10 (1990), 1550–1560.
[57] H. Xie, S. Fang, Z. Zha, Y. Yang, Y. Li, and Y. Zhang. 2019. Convolutional attention networks for scene text recognition. *ACM Transactions on Multimedia Computing, Communications, and Applications* 15, 1s (2019), 1–17.
[58] X. Xu, I. Broustis, Z. Ge, R. Govindan, A. Mahimkar, N. Shankaranarayanan, and J. Wang. 2015. Magus: Minimizing cellular service disruption during network upgrades. In *Proc. ACM CoNEXT.*
[59] F. Yaghoubi, M. Furdek, A. Rostami, P. Öhlén, and L. Wosinska. 2018. Consistency-aware weather disruption-tolerant routing in SDN-based wireless mesh setworks. *IEEE Trans. Netw. Service Manag.* 15, 2 (2018), 582–595.
[60] Z. Yang, P. Pathak, Y. Zeng, and P. Mohapatra. 2015. Sensor-assisted codebook-based beamforming for mobility management in 60 GHz WLANs. In *Proc. IEEE MASS.*
[61] S. Yin, D. Chen, Q. Zhang, and S. Li. 2011. Prediction-based throughput optimization for dynamic spectrum access. *IEEE Trans. Veh. Technol.* 60, 3 (2011), 1284–1289.
[62] C. Yue, R. Jin, K. Suh, Y. Qin, B. Wang, and W. Wei. 2018. Linkforecast: Cellular link bandwidth prediction in LTE networks. *IEEE Trans. Mobile Comput.* 17, 7 (2018), 1582–1594.

A PROOF OF PROPOSITION 4

Proposition 4 (Max-Min Fairness of the MSNR algorithm). *The optimal sequence of network configurations* $\{f^{(k,l)}_{n,t+h}, z_{n,t+h}\}$ *given by the MSNR algorithm has admission rates* $\{z_{n,t+h}\}_{n=1}^{N-1}$ *that are max-min fair in every time-step* $t+h$ *and for any given* $h \in \{0, 1, \ldots, H-1\}$*, irrespective of the topology* $G = (V, E)$*, demands* d_n*, and current and predicted link capacities* $\{c^{(k,l)}_t, \hat{c}^{(k,l)}_{t+1}, \ldots, \hat{c}^{(k,l)}_{t+H}\}$.

Proof. Proposition 4 holds by the design of the MSNR algorithm. In the first iteration, Algorithm 1 finds the *lowest* admission rate $\bar{z}^*$ that saturates at least one commodity (n, h), assigns $z_{n,t+h} \leftarrow \bar{z}^*$, and removes the new saturated commodities from the set of unsaturated commodities, i.e., $U \setminus (n, h)$. Similarly, in each subsequent iteration k, Algorithm 1 finds the *lowest* admission rate $\bar{z}^*$ that saturates at least one *unsaturated* commodity $(n, h) \in U$, assigns $z_{n,t+h} \leftarrow \bar{z}^*$, and performs $U \setminus (n, h)$. The algorithm terminates when all commodities are saturated, i.e. $U = \emptyset$.

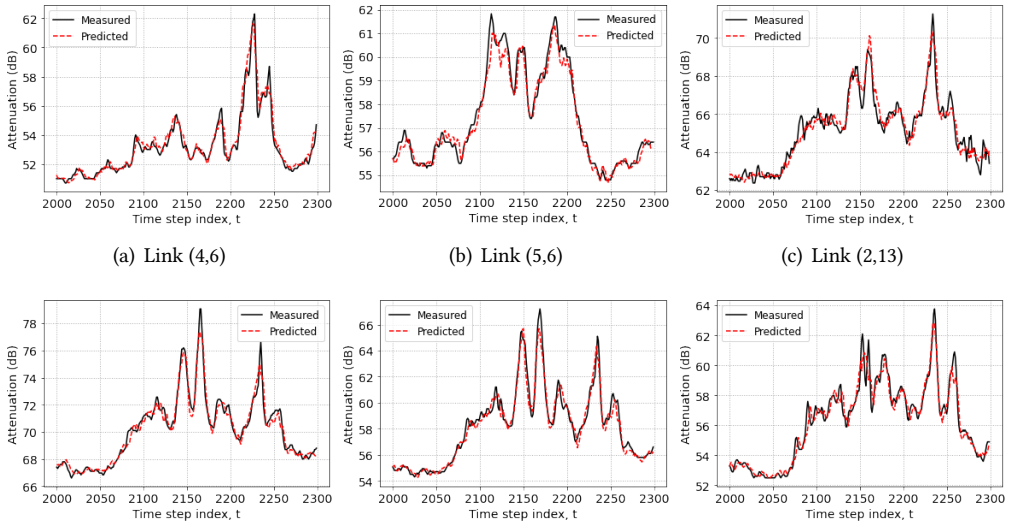
Consider one of the commodities (n, h) that became saturated during iteration k. To increase its admission rate beyond saturation $z_{n,t+h}$, we would have to reduce the admission rate of at least one other commodity (n', h) that became saturated either in iteration k or in a previous iteration[12]. Notice that, by the design of Algorithm 1, the saturation admission rate of commodity (n', h) is lower or equal to $z_{n,t+h}$. This means that, in each iteration k, the set of saturated admission rates $\{z_{n,t+h}\}_{(n,h)\notin U}$ is max-min fair. It follows that, upon termination, Algorithm 1 yields admission rates $\{z_{n,t+h}\}_{n=1}^{N-1}$ that are max-min fair. □

[12]Notice that if we could increase the admission rate of (n, h) beyond saturation $z_{n,t+h}$ without reducing the admission rates of another saturated commodity (n', h), then (n, h) was not saturated.

B EVALUATION OF THE AP MECHANISM

In this appendix, we complement the evaluation of the AP mechanism presented in Sec. 5.1. Specifically, we include: (i) results associated with *Test Seq. III*; (ii) results associated with different links in the network; and (iii) scatter plots that relate the measured attenuation values with the corresponding predictions.

In Fig. 10(a)-(f), we compare the evolution of the attenuation measurements $x_{t+3}^{(k,l)}$ with the 3-steps-ahead predictions $\hat{x}_{t+3}^{(k,l)}$ generated by the AP mechanism during an interval of 300 time-steps from *Test Seq. I*. The results in Fig. 10 suggest that the attenuation predictions accurately track the measurements.



(a) Link (4,6) (b) Link (5,6) (c) Link (2,13)

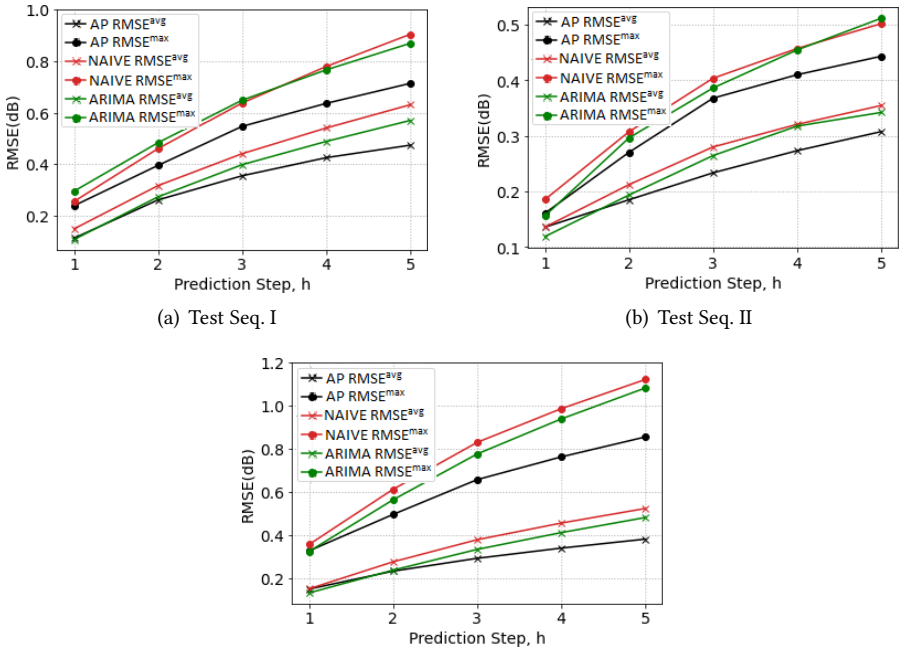
(d) Link (6,13) (e) Link (9,13) (f) Link (10,13)

Fig. 10. Comparison of the actual (measured) attenuation from different links in the network with the corresponding 3-steps-ahead predictions, i.e., predictions of 30 seconds into the future.

In Figs. 11(a)-(c), we display the $RMSE_h^{\text{avg}}$ and $RMSE_h^{\max}$ (in dB), defined in (7), as a function of the prediction horizon $h \in \{1, \ldots, H\}$ for *Test Seq. I*, *II*, and *III* and for three prediction mechanisms: (i) the AP mechanism; (ii) the naive method; and (iii) the ARIMA model. The results in Fig. 11 suggest that the AP mechanism outperforms the benchmark methods in all three test sequences and that this performance improvement increases as the prediction horizon h increases.

In Figs. 12(a)-(b), we show scatter plots that relate the measured attenuation values with the corresponding 1-step-ahead predictions $\hat{x}_{t+1}^{(k,l)}$ and 5-steps-ahead predictions $\hat{x}_{t+5}^{(k,l)}$ for edges (10,13) and (12,13) during *Test Seq. I*. The dashed line represents zero prediction error. The results in Fig. 12 suggests that: (i) prediction errors increase with the prediction horizon h; (ii) prediction errors increase at higher measured attenuation levels (i.e., when the rain is stronger); and (iii) at high measured attenuation levels, predictions tend to underestimate the attenuation.

Received August 2022; revised October 2022; accepted November 2022

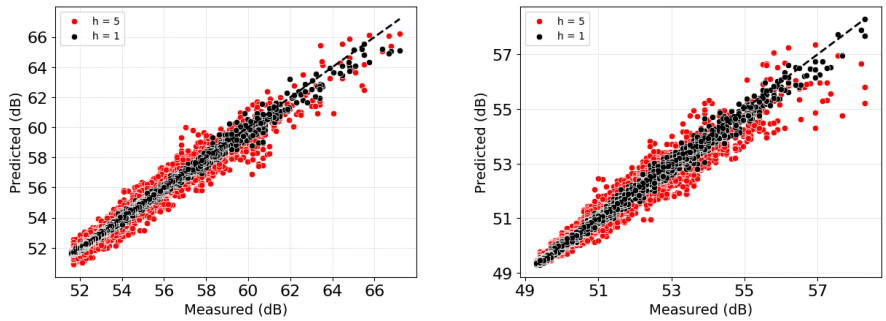


(a) Test Seq. I

(b) Test Seq. II

(c) Test Seq. III

Fig. 11. $RMSE_h^{\text{avg}}$ and $RMSE_h^{\max}$ of the prediction error for different horizons h and for the AP mechanism, naive method, and ARIMA model.



(a) Link (10,13)

(b) Link (12,13)

Fig. 12. Scatter plots that relate the measured attenuation values with the corresponding 1-step-ahead predictions $\hat{x}_{t+1}^{(k,l)}$ and 5-steps-ahead predictions $\hat{x}_{t+5}^{(k,l)}$ for edges (10,13) and (12,13) during Test Seq. I.