Baseband-Function Placement with Multi-Task Traffic Prediction for 5G Radio Access Networks

Zorello, Ligia Maria Moreira; Bliek, Laurens; Troia, Sebastian; Guns, Tias; Verwer, Sicco; Maier, Guido

DOI
10.1109/TNSM.2022.3190059

Publication date
2022

Document Version
Final published version

Published in
IEEE Transactions on Network and Service Management

Citation (APA)
Zorello, L. M. M., Bliek, L., Troia, S., Guns, T., Verwer, S., & Maier, G. (2022). Baseband-Function Placement with Multi-Task Traffic Prediction for 5G Radio Access Networks. *IEEE Transactions on Network and Service Management*, *19*(4), 5104 - 5119. https://doi.org/10.1109/TNSM.2022.3190059

Baseband-Function Placement With Multi-Task Traffic Prediction for 5G Radio Access Networks

Ligia Maria Moreira Zorello, *Graduate Student Member, IEEE*, Laurens Bliek, Sebastian Troia, Tias Guns, Sicco Verwer, and Guido Maier

***Abstract*—The 5G Radio Access Network (RAN) virtualization aims to improve network quality and lower the operator's costs. One of its main features is the functional split, i.e., dividing the instantiation of RAN baseband functions into different units over metro-network nodes. However, its optimal placement is non-trivial: it depends on the application requirements and on the expected traffic volume, whose daily variation highly impacts the total power consumption. Current optimization solutions fail to provide a placement solution capable of handling traffic fluctuations. In fact, the standard machine learning algorithms used in the literature for planning the network resources in advance result in an allocation that is inadequate to carry the actual traffic at all the time-slots. Hence, we must reserve an artificial buffer capacity in the nodes to ensure feasibility. Instead, our proposed method exploits a fine-grained two-step multi-task algorithm that predicts the mean and quantile traffic, making the artificial capacity no longer necessary. The subsequent placement uses mixed-integer linear programming and a heuristic. The former considers the expected traffic in the objective function (to estimate costs) and the quantile in the constraints (to enforce capacity limits). The heuristic combines the mean and quantile results to minimize the power and comply with the requirements. While using sufficiently large artificial buffers guarantees robustness with a mild power increase compared to the oracle, the fine-grained multi-task model improves the results, reducing the power consumption compared to the mean and meets all constraints. The heuristic enables significant computational time reduction.**

***Index Terms*—5G radio access network, functional split, network function virtualization, traffic prediction, mathematical optimization, machine learning, energy saving.**

Manuscript received 15 June 2021; revised 30 November 2021 and 9 May 2022; accepted 27 June 2022. Date of publication 12 July 2022; date of current version 31 January 2023. The work leading to these results has been supported by the European Community under grant agreement no. 761727 Metro-Haul project. The associate editor coordinating the review of this article and approving it for publication was M. F. Zhani. *(Corresponding author: Ligia Maria Moreira Zorello.)*

Ligia Maria Moreira Zorello is with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy. He is now with Meta Inc, London, U.K. (e-mail: ligiamaria.moreira@polimi.it).

Sebastian Troia and Guido Maier are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy.

Laurens Bliek is with the Department of Industrial Engineering and Innovation Sciences, TU Eindhoven, 5600 MB Eindhoven, The Netherlands.

Tias Guns was with the Economic and Social Sciences & Solvay Business School, Vrije Universiteit Brussel, 1050 Brussels, Belgium. He is now with KU Leuven, Leuven, Belgium.

Sicco Verwer is with the Faculty of Electrical Engineering, Mathematics and Computer Science, TU Delft, 2628 CD Delft, The Netherlands.

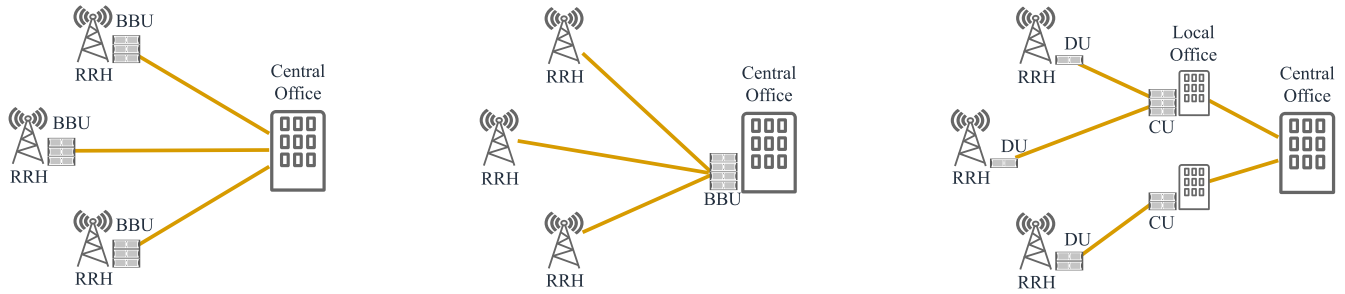
Digital Object Identifier 10.1109/TNSM.2022.3190059

I. Introduction

THE FIFTH generation of mobile networks (5G) was developed to carry services with different Quality of Service (QoS) requirements with improved energy efficiency. It was necessary to improve the Decentralized Radio Access Network (D-RAN) of 4G (Fig. 1(a)), as it would lead to an inefficient and highly costly architecture for mobile operators at scale. 4G further introduced the Centralized RAN (C-RAN) (Fig. 1(b)) to decouple the Remote Radio Head (RRH) and the Baseband Unit (BBU) and fully centralize and virtualize the BBUs in powerful central offices Fig. 1(b). This architecture enhances the network management and control, the processing scalability, and it reduces the overall power consumption due to the better utilization of the computing resources. However, several issues are yet to be addressed to enable its broad deployment. The consolidation of baseband functions in a single central node requires very high traffic capacity over the fronthaul links connecting the RRHs to their corresponding BBU, and it imposes stringent latency requirements between sites [1], [2].

To diminish these issues and find a trade-off between power consumption and processing and bandwidth capacity, part of the baseband functions can be virtualized and redistributed over local offices in the network, as shown in Fig. 1(c). The 3rd Generation Partnership Project (3GPP) proposed a 5G RAN with functional splits, indicating the centralization degree of baseband functions [3]. While the RRH continues in the antenna site, the BBU functions are deployed as Virtual Network Functions (VNFs) into Distributed Unit (DU) and Centralized Unit (CU). The DUs deploy lower-layer network functions in the access network, and the CUs deploy the remaining functions over cloud-enabled nodes in the metro network. The CUs and DUs are directly connected via midhaul links.

The efficient placement of these VNFs in the network nodes becomes a new optimization problem. There is a significant effort in the literature to solve it. Nevertheless, aspects related to the costs and to the functional split constraints were not fully considered. Network operators must minimize their operational costs when deploying baseband VNFs while ensuring that service requirements are always satisfied. For this reason, it is cumbersome to use power models that take into account different aspects of the network. As shown in our previous



(a) Distributed RAN. (b) Centralized RAN. (c) 5G-RAN with splits.

Fig. 1. Evolution of RAN architecture: completely distributed (a), completely centralized (b) and partially centralized with functional splits (c).

work [4], the network power consumption plays a key role in building up the cost of an NFV infrastructure. Therefore, we study the optimal VNF placement to minimize the node and network power consumption. In addition, despite being an important requirement, the split latency is frequency neglected in the literature. It guarantees that all baseband functions are correctly executed; thus, our approach ensures that the solution is always compliant with the split and service latency requirements. We consider the placement of the CUs while the DUs are assumed as fixed in the access network.

This optimization highly depends on the network traffic, which shows fluctuations during the day due to users' movement in urban areas. Therefore, the network operator may need to move the baseband VNFs over the network nodes to adapt to the traffic and comply with split and service requirements at all times. The baseband VNFs may be deployed as Virtual Machines (VMs) or containers over general-purpose servers. Consequently, the migration of the baseband VNFs can be translated into moving their hosting VMs or containers over the network. Independently on the technique applied, the reconfigurating the baseband VNF placement involves the VNF instantiation and migration and lightpath establishment. The entire process is time consuming, making the computation and reconfiguration of the network on a real-time basis impractical and leading to disruptions of service. Therefore, it is necessary to plan the VNF placement in advance. Machine learning algorithms can aid this anticipated optimization taking advantage of the tidal daily traffic variation in urban areas due to the regularity of inhabitants and commuters displacement [5] These algorithms provide accurate traffic forecast and improves the applicability of optimization frameworks in real-time scenarios. However, unpredictable events may create perturbations in these patterns. Therefore, the traditional techniques used in the literature using machine learning to predict the traffic and then optimize the placement fail to provide solutions that ensures feasibility in real-time. Indeed, they do not guarantee that the real traffic could be carried by the anticipated placement.

We model the baseband VNF placement as a modular optimization problem in a metro-area network from the point of view of a mobile operator in the context of the Metro-Haul project [6]. In particular, we assume that the nodes are distributed cloud-enabled, i.e., they are equipped with computing power that can host baseband VNFs. A flexible and high-capacity Wavelength Division Multiplexing (WDM) optical network interconnects these nodes. In this scenario, the first stage of the optimization is an hourly-based traffic forecast consisting of a two-step multi-task prediction. The multi-task machine learning algorithm outputs the mean and the normalized traffic, then scales it using persistent forecast. The forecast traffic is then used as input to calculate the placement to reduce the power consumption for the operator. We propose an optimization algorithm based on Mixed Integer Linear Programming (MILP) to minimize the network and IT power consumption subject to the functional split and service constraints. In order to ensure a robust solution when applying the actual traffic to the configured network, the traffic prediction is divided into two outputs. We apply the expected traffic per node, i.e., the mean predicted value, to the objective function. The optimization constraints use the quantile predictions to consider more strict scenarios and avoid underestimating the network configuration. To mitigate the MILP complexity, we propose a heuristic algorithm to compute the placement considering the predicted traffic.

The remainder of the paper is organized as follows. Section II surveys the related works. Section III details the machine-learning-aided optimization, describing the multi-task two-step traffic prediction and the proposed MILP, including the models that describe the system. Section IV shows the simulation environment, the performance of the proposed traffic prediction tool, and the results when operating the MILP on the predicted traffic. Finally, Section V concludes the paper.

II. Related Works

This section details the works that tackle the baseband VNF placement optimization problem. First, it describes the papers that use mathematical optimization. Then, it highlights the solutions to speed up this task by using heuristic algorithms and machine learning, focusing on traffic prediction.

A. Optimization of Baseband VNF Placement

The baseband VNF placement in 5G-RAN has been considerably studied in the past few years with the development of 5G solutions. Tzanakaki *et al.* introduced in [7] an optimization to select the transport technology (optical or wireless) and then optimize the utilization of the data center network and processing resources. Although authors

considered different split options deployed, they assumed all baseband functions placed in a unique centralized data center. Yu *et al.* proposed in [8] a placement optimization algorithm to reduce the number of active nodes hosting baseband functions. However, this work did not take into account the network components in the optimization, which, as we showed in [4], significantly impacts the power consumption for operators. Al-Quzaweeni *et al.* [9] minimized the power consumption when allocating baseband functions and routing users traffic considering both processing and network components. In the same line, Murti *et al.* [10] developed an algorithm to deploy CUs with different functional splits over a 5G RAN to reduce the overall operational expenses. Tinini *et al.* [11] optimized the BBU migration over cloud and fog nodes in a Passive Optical Network to minimize power consumption. These works studied the baseband consolidation but did not evaluate the impact that different services have on the network. In addition, they ignored or adopted a simplified model for the latency that considers only the split latency and/or average service latency.

Yusupov *et al.* [12] placed baseband functions to carry different categories of services, ensuring low latency. They minimized the number of used nodes and the service latency and maximized the remaining data rate on links to carry these services. Ojaghi *et al.* performed radio spectrum slicing and baseband placement considering different service requirements to minimize uniquely computational cost over network nodes [13]. Similarly, Matoussi *et al.* [14] proposed a joint optimization of RAN slicing and split selection to transport different service types and minimize the power consumption. However, these works did not optimize the location of such functions.

These papers proposed optimization methods based on linear programming to compute the baseband VNF placement. However, this type of problem is typically NP-Hard, making it hard to provide a fast reaction to traffic changes. Consequently, these solutions cannot be deployed in real-time scenarios.

B. Heuristic Techniques

Some solutions speed up the computation of the optimal placement and enable a more dynamic placement. Gupta *et al.* [15] propose an optimization and a heuristic algorithm to place CUs based on the energy consumption and the number of handovers, considering different functional splits. Nevertheless, it does not evaluate the impact of traffic variation. Singh *et al.* [16] optimized the baseband VNF placement to minimize the energy consumption. For this, they developed an integer quadratic programming model and a more computationally efficient decomposition-based heuristic model. Xiao *et al.* proposed in [17] a MILP model and a heuristic algorithm to minimize the power consumption when placing the baseband functions and provisioning lightpaths over a flexible optical transport network. The authors compared the impact when different services must be carried over the network separately. Sigwele *et al.* [18] maximized the energy efficiency of the C-RAN radio and cloud parts. On the transport side, the proposed heuristic algorithms aim to reduce the number of active nodes.

None of these works evaluated the effect of multiple services sharing the same network infrastructure. In addition, although heuristic algorithms have reduced complexity and can solve the optimization problem much faster than linear programming, the VNF migration and the lightpath establishment are time-consuming. Hence, it is necessary to compute the placement ahead of time.

C. Machine-Learning Techniques for Dynamic Optimization

The integration of machine learning is in improving the dynamicity and applicability of optimization in real-time solutions. In the context of 5G RAN, Pelekanou *et al.* [19] deploy machine learning algorithms first to forecast the traffic and then select which baseband function to place in a single centralized data center. Yu *et al.* [20] allocate, migrate and scale 5G RAN slices over metro-aggregation networks based on traffic prediction. Gao *et al.* present in [21] a deep reinforcement learning approach to dynamically adjust the baseband function placement and routing to minimize the used nodes, the bandwidth, and the transport latency. Zhu *et al.* [22] propose a heuristic-assisted deep reinforcement learning-based algorithm to decide whether to aggregate BBUs or to avoid traffic migration. Chen *et al.* [23] optimized the RRH-to-BBU association based on the traffic predicted using a multivariate long short term memory. Guerra-Gómez *et al.* presented in [24] a dynamic computational resource allocation of BBU-related functions using machine learning with an error-shifting technique to define a margin of computational resources to consider in the optimization.

Several works on traffic prediction have been published in the past years. Zhang *et al.* [25] use machine learning techniques based on Feed Forward Neural Networks (FFNN) and Recurrent Neural Networks (RNN) to cope with network resource utilization and handover by predicting users' traffic and mobility. Other works use more modern machine learning algorithms for traffic forecasting. Andreoletti *et al.* [26] deploy diffusion convolutional recurrent neural networks to capture topological properties from the network and predict the load over network links. Bega *et al.* [27] describe a data analytics tool based on three-dimensional convolutional neural networks to learn spatio-temporal features and forecast the load of different services. Nevertheless, the former two methods predict the traffic load over the links, which are associated with routing the demands in the network independently on the baseband function placement.

These works aim to obtain the most accurate model to predict the network traffic considering extensive topology and statistics information. However, the high prediction accuracy does not guarantee a robust optimization regarding the actual traffic. For this, we propose a two-step multi-task approach. It outputs the mean expected traffic to estimate the optimization costs and the traffic uncertainty (measured as quantile) to guarantee that the constraints are respected. Moreover, this prediction model is based only on two parameters, the hour of the day and the aggregated traffic of previous days, significantly reducing the model size and complexity.

D. Paper Contribution

In this work, we propose a traffic prediction algorithm tailored to the optimization of baseband VNF placement. Thanks to the traffic prediction, the network operators can reconfigure the placement of such functions in advance according to the expected demands. The contributions of this paper are summarized as follows:

1) We introduce an optimization framework that forecasts the traffic and computes the baseband VNF placement for different services to efficiently plan and reconfigure the 5G-RAN network in advance.
2) We propose a two-step multi-task traffic predictor capable of significantly reducing the model complexity, while ensuring robustness to the optimization.
3) We formulate the placement as a MILP problem, in which we minimize the system power consumption considering multiple services and splits.
4) We develop a heuristic algorithm to reduce the MILP computational time.
5) We conduct a series of experiments and prove that the proposed framework provides fast and accurate optimization of the baseband VNF placement while satisfying the actual demand.

III. Machine-Learning-Aided Baseband VNF Placement

The optimal baseband VNF placement depends on the requirements related to the node selection and traffic routing. Such requirements are strictly related to the service type and to the selected functional split. This work considers two typical mobile services: Voice over IP (VoIP) and content delivery. The former does not generate a large volume of traffic but requires a strict latency budget to guarantee the quality of the calls. Therefore, it enters the category of ultra- Reliable and Low Latency (uRLLC) services [28]. Instead, the latter demands greater traffic capacity over network links but looser latency requirement, entering the category of enhanced Mobile Broadband (eMBB) services. After this, we refer to these services as uRLLC and eMBB. In this work we did not consider the mMTC services as they did not match with the traffic dataset we had at our disposal. In this work we did not consider the mMTC services as they did not match with the traffic dataset we had at our disposal.

Some splits are more appropriate to support a given service so that the split constraints can match the application requirements in terms of bandwidth and latency [29]. Physical-layer splits compel more strict latency requirements and high midhaul traffic but enable greater coordination thanks to a more centralized network control. Therefore, low-layer split options are more adapted to uRLLC applications, which typically imply low traffic loads but stringent latency and high coordination. In contrast, high-layer functional splits centralize fewer data link-layer functions from the baseband protocol stack. This more distributed architecture leads to lower CU computational complexity and to looser latency constraints. Moreover, it is capable of minimizing the traffic load over the metro network links. Hence, they are more suitable for eMBB services.[1]

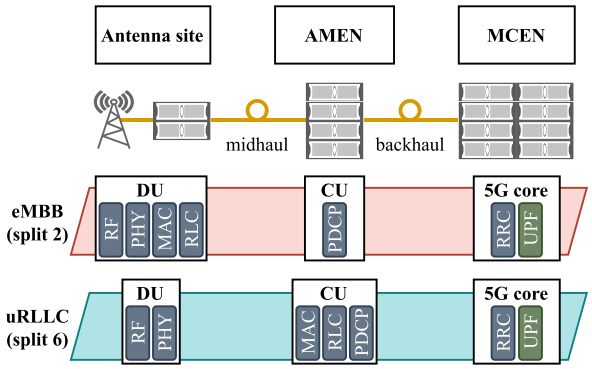


Fig. 2. Location of UPF and baseband functions over AMENs and MCENs according to split and service types. Radio Resource Control (RRC), Packet Data Convergence Protocol (PDCP), Radio Link Control (RLC), Medium Access Control (MAC), Physical layer (PHY), Radio Frequency (RF).

We assume in this work that traffic generated by services belonging to the eMBB class is processed by the CU using split #2 of 3GPP split set. [3][2] Conversely, traffic generated by uRLLC applications is processed by the CU using split #6. Moreover, we assume that the User Plane Functions (UPFs) related to each service are located in the core network. Fig. 2 depicts the division of baseband functions among DU and CU for each service.

We place the baseband VNFs over a flexible IP-over-optical metro network. In order to host these VNFs, the network nodes must be equipped with general-purpose servers capable of processing such functions. High-capacity fiber links connect these nodes to carry the demands from all RRHs to the gateway. In addition, the network infrastructure must enable the reconfiguration of lightpaths to establish them according to the baseband VNF placement. For this, we take advantage of the metro network infrastructure proposed by the Metro-Haul project [6]. It is composed of a set of metro and core nodes interconnected by a high-capacity and flexible WDM network. We assume that the core nodes, or Metro Core Edge Nodes (MCENs), host all UPFs. The metro nodes, called Access Metro Edge Nodes (AMENs), are enabled with computing capacity to support the instantiation of VNFs. Each of them is connected to a set of DUs using optical links. Each DU serves to a pool of adjacent RRHs via the access network. Based on this configuration, we also consider a set of demands for eMBB and uRLLC services arriving at the nodes. These demands generate hourly traffic corresponding to the upstream (downstream) traffic generated by (destined to) all mobile users connected to the RRHs connected to this node.

Based on this architecture, we detail in the remaining of this section a machine learning-aided optimization to place the baseband VNFs depicted in Fig. 3. As previously explained, the baseband VNF placement and routing must be computed beforehand to ensure an efficient network configuration.

[1]To reduce the DU computational complexity, the baseband functions could be divided between the DUs and RRHs using a low-layer split.

[2]The DU placement in the access network is not object of this work.

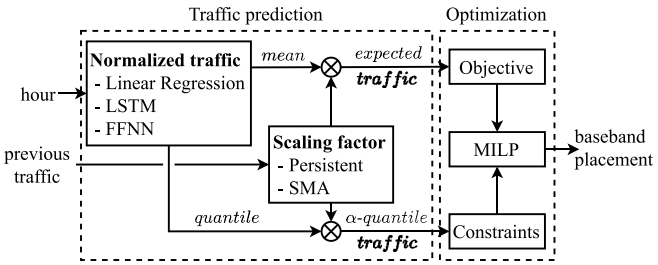


Fig. 3. Traffic prediction and baseband VNF placement optimization.

Therefore, the proposed framework first estimates the expected traffic and the α-quantile uncertainty using a two-step multi-task algorithm. For this, it predicts the mean and the quantile normalized traffic using traditional machine learning algorithms and then multiplies them by a scaling factor. Then, the predicted traffic is given as input to the MILP model responsible for optimizing the baseband VNF placement. In particular, it applies the expected traffic to the objective function to estimate the costs and the quantile in the constraints to ensure feasibility in the worst-case scenario.

A. Traffic Prediction in Metro Nodes

This section describes the two-step traffic prediction model depicted on the left-hand side of Fig. 3. Predicting traffic is a classical problem of time series forecasting. Therefore, regression supervised machine learning techniques can be applied to predict a real value based on the information of previously seen data. In this regard, as mentioned in Section II, several approaches have been proposed in the literature. Most of these works consider multiple characteristics related to the historical traffic and the network topology to model the prediction. The approach proposed in this work is tailored to the baseband VNF placement optimization problem and reduces the complexity of the models, improving training time and storage requirements.

The insight behind our approach is the tidal movement of large populations in a city during the day that displays a fairly stable seasonality [30]. The users' traffic arrives at each node of the metro network from the RRHs to which they are connected regularly during weekdays, resulting in a very similar traffic shape and a slow variation of the traffic volume with respect to the previous days. This phenomenon is better illustrated in Section IV.

In contrast with the traditional approaches in which the expected traffic is predicted directly based on previous traffic information in a single step, our proposal consists of a two-step prediction. We divide the traffic forecast into two parts, normalized and scaling factor prediction, as shown in Fig. 3. We perform the normalization over the total daily traffic to avoid a substantial impact of the spikes. Hence, we consider the summation of the hourly traffic in each day as the scaling factor. We exploit the time series techniques of persistent forecast and Simple Moving Average (SMA) to compute this value. The persistence forecast strategy, or naive forecast, uses the previous observation directly as the forecast value. The SMA calculates the average of previous n observations. More specifically, in this work, we deploy persistent forecast to output the total traffic volume observed in the previous day, and SMA to consider the total traffic of the previous 5 days.

To determine the normalized traffic, the unique input parameter necessary is the hour. For this, we employ traditional regression machine learning techniques. The simplest available approach is linear regression, whose model obtains linear functions of the parameters used in this algorithm. They offer simple analytical properties; nevertheless, they often present significant limitations in practical use cases. Alternatively, artificial neural networks were shown to reach high accuracy in traffic forecast applications. In this work, we consider two algorithms: FFNN and RNN. FFNN consists of a series of nodes, called neurons, interconnected through weighted edges in a feed-forward way [31]. RNN introduces a recurring structure to traditional FFNNs such that each neuron forwards feedback to the next steps. This characteristic makes RNN convenient for time-series predictions. Among the different algorithms available, this work focuses on Long Short-Term Memory (LSTM) [32], typically capable of learning long-term dependencies in sequence prediction problems and extensively used in traffic prediction use cases.

In any machine learning model, the training phase consists of minimizing a loss function that evaluates the output with respect to the expected value. In traditional regression methods, the best model is found by minimizing the Mean Squared Error (MSE) (Eqn. (1)). The minimization of such loss function defines the conditional mean of the target over the feature values. However, it is not always sufficient for a machine learning model to make accurate predictions. In applications in which the forecast value is further used in an optimization problem, it is equally important to understand the uncertainty of these predictions. This behavior can be achieved by applying quantile regression. As its name says, this model estimates the conditional quantiles, giving priority to the *i-th* percentile of the input data. Consequently, its predictions are more robust against outliers. Eqn. (2) computes the loss function of quantile regression.

$$L_{MSE}(Y, \hat{Y}) = \frac{1}{n}\sum\Big(Y_i - \hat{Y}_i\Big)^2 \tag{1}$$

$$L_{quant}(Y, \hat{Y}) = \begin{cases} (1-\alpha)(\hat{Y} - Y), & \text{if } \hat{Y} \geq Y, \\ \alpha(\hat{Y} - Y), & \text{if } \hat{Y} < Y \end{cases} \tag{2}$$

where Y is the true value, $\hat{Y}$ is the predicted value for the normalized traffic and α gives the αth percentile of Y. In particular, the loss function utilized for the quantile regression presents an asymmetric behavior, such that for $\alpha < 0.5$ it penalizes positive errors.

This work integrates the mean and the quantile predictions to the optimization setup. We apply the mean traffic prediction to the objective function because it provides a good estimate of expected traffic, whose relation with the objective we wish to minimize. Instead, the optimization could benefit from the quantile regression as it provides an upper bound on the expected traffic. This allows us to optimize the baseband VNF placement considering the expected value while ensuring the feasibility of the constraints. In order to predict both values, we use a multi-task learning approach. In this setting, the neural

network model has two distinct outputs: the mean prediction value and the quantile prediction. We model each output, or task, with its own loss function, i.e., L_{MSE} and $L_{quantile}$ for the mean and the quantile outputs, respectively. During the training phase, the model learns a shared representation such that the loss function becomes a combination of both tasks:

$$L = \alpha \cdot L_{MSE}(Y, \hat{Y}_{MSE}) + (1-\alpha) \cdot L_{quant}(Y, \hat{Y}_{quant}), \tag{3}$$

where $\hat{Y}_{MSE}$ and $\hat{Y}_{quant}$ represent the output of the mean and quantile tasks, and α weights the importance of each loss. In particular, we set it to 0.5 as they are equally important.

B. Optimization of Baseband VNF Placement

The second part of the optimization (right block of Fig. 3) consists of the baseband VNF placement problem, extending our work in [4]. The goal is to minimize the overall power consumption when deploying baseband functions to carry different services over the network. Because of the stringent latency constraints of the uRLLC service, the optimization gives it priority over eMBB when being processed. Moreover, the optimization relies on the constraints related both to the split and to the services, as described as follows.

We consider a metro network composed of *N* nodes and *E* links, in which one node is assumed as the gateway and the remaining are candidate CU nodes to host the split 2 and split 6 baseband functions. Nodes have a computing capacity C_n and the links can carry a maximum bandwidth C_e. We also assume that there are *D* uplink demands arriving from the RRHs connected to each of the network nodes ($n_d \in N$), and whose destination is the gateway (n_{gw}). Each demand $d \in D$ requests a certain amount of traffic described by λ_d^m (mean) and λ_d^q (quantile) associated to a service and, thus, to a functional split $f \in F = \{2, 6\}$. All parameters and variables are described in Table I.

Processing Load: The baseband processing load depends on the split option selected. The model used in this paper assumes the processing load of a split to be proportional to the capacity required to process a Common Public Radio Interface (CPRI) flow [33]:

$$\pi_f = \sigma_f \left(n_a^2 + 3 \cdot n_a + \frac{M_b \cdot C \cdot L}{3} \right) \cdot \frac{R}{10}, \tag{4}$$

where n_a is the number of antennas, M_b is the modulation, *C* is the coding rate, *L* is the number of MIMO layers, and *R* is the number of resource blocks per user. The scaling factor σ_f is calculated as a ratio of the split and the CPRI computational complexity according to [34]. In the particular case of the functional splits used in this paper, the CU scaling factor is 0.13 and 0.42 for split 2 and split 6, respectively, while the values for the DU are 0.87 and 0.58.

Latency: The computation of the maximum acceptable service and split latency is impacted by the delays drawn by both network and IT components. This paper considers the following parameters:

- Propagation delay (t_e): product of the distance of the fiber links through which the traffic of a demand needs to be transported and the specific propagation delay per unit length in a fiber (approximately 5 μs/km).
- Switching latency (t^{sw}): product of the number of electronic switches crossed by the demand and the specific switch delay, assumed as 20 μs per traversed switch [35].
- Baseband processing delay (t_f): depends on the server characteristics and on the processing load required to process a demand at a certain split [36], [37]:

$$t_f = \frac{\pi_f}{\pi_{CPU} \cdot f_{CPU}}, \tag{5}$$

where π_f is the split processing capacity from Eqn. (4), π_{CPU} is the server maximum processing load in GOPS, and f_{CPU} is CPU operating frequency in GHz.

This work ensures that the baseband VNF placement always meets the QoS requirements, i.e., the placement must be compliant to the maximum acceptable latency of the splits and service requested by the different demands. For this, we assume the split latency as the propagation and switching delay between the DU and the CU, summed to the processing delay of the requested split. The maximum split latency is 1.5 ms for split 2 and 0.25 ms for split 6 [2]. For the service latency, we consider the propagation along the whole path from the RRHs to the gateway, and the processing delay of DU and CU nodes. The maximum service latency accepted by the uRLLC service is 0.5 ms and by the eMBB service is 5 ms [38].

Power Consumption: To determine the overall power consumed by the system, we divide it into two parts: network and processing. The network consumption P_{net} is given by the power consumed by the transponders, being the transponder

TABLE I
PARAMETERS AND VARIABLES USED IN THE MILP FORMULATION

Parameters	
P^s_{idle}:	Server idle power consumption
P^s_{max}:	Server maximum power consumption
P^t_{max}:	Transponder power consumption
η_n:	Servers available at node
C_n:	Node maximum processing capacity
C_e:	Link maximum number of wavelengths
λ^m_d:	Mean predicted traffic load of demand d
λ^q_d:	Quantile predicted traffic load of demand d
π_f:	Processing workload of functional split f
t_e:	Link e propagation delay
t^{sw}:	Switching delay
t_f:	Processing latency with functional split f at CU
t^{DU}_f:	Processing latency with functional split f at DU
t^{split}_f:	Maximum allowed latency of functional split f
$t^{service}_f$:	Maximum allowed latency of service using split f
n_d:	Node where demand d starts
n_{gw}:	Gateway node
$\delta_{e,n}$:	1 if link e ends at node n
$\delta_{d,f}$:	1 if demand d requires functional split f
$\mathcal{M}$:	Very large number
Variables	
w_n:	1 if node n hosts a CU
$y_{d,n}$:	1 if node n is assigned to demand d
$x_{d,e}$:	1 if link e carried demand d
$x^{CU}_{d,e}$:	1 if link e carried demand d until CU
$T_{d,n}$:	Processing delay for demand d at node n

an optical device responsible for performing optical-electrical-optical conversion. We assume that whenever a lightpath is created, the two transponders associated with it are active and consuming $2P^t_{max}$. The processing power P_{proc} depends on the node utilization, i.e., the node consumes its idle power P^s_{idle} whenever it is active, and its consumed power increases according to its utilization up to the maximum power P^s_{max}. The node utilization is directly related to the baseband VNFs hosted by this node and by the total processed traffic:

$$\Pi_n = \sum_{d \in D_n} \sum_{f \in F} \frac{\lambda_d \delta_{d,f} \pi_f}{C_n}, \tag{6}$$

where D_n is the set of demands processed by node n, π_f is the split processing capacity from Eqn. (4), $\delta_{d,f}$ indicates that demand d requires functional split f, and C_n is the node maximum processing capacity.

1) Objective Function: The optimization goal is to minimize the power consumption related to processing and network components. Based on the power consumption description provided above, the objective function is represented by Eqn. (7). As shown in Fig. 3, in order to estimate the expected placement cost, the node power consumption related to its utilization considers the mean predicted traffic λ^m_d. The weight β determines the impact of each component. In particular, we set it to 0.5 to give equal importance to both metrics.

$$\min \beta P_{net} + (1-\beta) P_{proc}, \tag{7}$$

where

$$P_{net} = \sum_{d \in D} \sum_{e \in E} \sum_{n \in N} \left(P^t_{max} \delta_{e,n} x_{d,e} \right) \tag{8a}$$

$$P_{proc} = \sum_{n \in N} \eta_n \left(P^s_{idle} w_n + \frac{\left(P^s_{max} - P^s_{idle}\right)}{C_n} \times \sum_{d \in D} \sum_{f \in F} \left(\lambda^m_d \delta_{d,f} \pi_f \right) y_{d,n} \right) \tag{8b}$$

2) Constraints: The aforementioned objective function is subject to the following constraints. The first set of constraints determine the path from the origin to the destination node. Eqn. (9) denotes the flow conservation, such that the number of links assigned to a demand arriving at a certain node (E^+_n) is equal to the number of outgoing links (E^-_n), unless the demand starts or ends at this node. The assignment of a demand to a link is denoted by $x_{d,e}$.

$$\sum_{e \in E^+_n} x_{d,e} - \sum_{e \in E^-_n} x_{d,e} = \begin{cases} -1, & \text{if } n = n_d \\ 1, & \text{if } n = n_{gw} \\ 0, & \text{otherwise} \end{cases}, \quad \forall d \in D, \forall n \in N \tag{9}$$

Eqn. (10) ensures that the path that carries the demand passes through the node assigned to it, indicated by $y_{d,n}$.

$$x_{d,e} \cdot \delta_{e,n} \geq y_{d,n}, \forall d \in D, \forall e \in E, \forall n \in N \tag{10}$$

Eqn. (11) establishes the path from origin to the CU, while Eqn. (12) determines that the links assigned to this demand until the CU ($x^{CU}_{d,e}$) belong to the path from the origin node to the gateway.

$$\sum_{e \in E^+_n} x^{CU}_{d,e} - \sum_{e \in E^-_n} x^{CU}_{d,e} = \begin{cases} 1, & \text{if } n = n_d \\ -y_{d,n}, & \text{otherwise} \end{cases}, \quad \forall d \in D, n \in N \tag{11}$$

$$x^{CU}_{d,e} \leq x_{d,e}, \forall d \in D, e \in E \tag{12}$$

We define that each demand is assigned to a single node:

$$\sum_{n \in N} y_{d,n} = 1, \forall d \in D. \tag{13}$$

Using the Big M method, constraint (14) specifies the nodes which host a CU, denoted by w_n, and ensures that the node processes a demand if and only if it hosts a CU.

$$\mathcal{M} \cdot w_n \geq \sum_{d \in D} y_{d,n} \geq w_n, \forall n \in N \tag{14}$$

The next two constraints delimit the capacity over the links and nodes. Eqn. (15) restricts the traffic of the services over the links to the total link capacity, whilst Eqn. (16) limits the processing load of each node to the maximum node capacity.

$$\sum_{d \in D} x^{CU}_{d,e} \leq C_e, \forall e \in E \tag{15}$$

$$\sum_{d \in D} \left(\sum_{f \in F} \left(\delta_{d,f} \pi_f \right) \lambda^q_d y_{d,n} \right) \leq C_n, \forall n \in N \tag{16}$$

Finally, constraints (17) and (18) ensure that the split and service latencies are limited to their maximum allowed values. Eqn. (19) and Eqn. (20) determine the processing delay of DU and CU nodes, respectively. We consider the propagation (t_e), switching (t^{sw}) and processing delays (t_f) previously defined. The split latency is the sum of the propagation delay from the DU to the node hosting a CU, the switching delay and the CU processing delay. Instead, the service latency sums up the propagation and switching delay until the gateway, and the processing delay in the DU and in the CU. Moreover, we assume that uRLLC have priority over the eMBB services, i.e., they are processed first in the CU. To guarantee such priority, we consider that $F' = \{6\}$ if demand d is of type uRLLC and $F' = F = \{2, 6\}$ otherwise in Eqn. (19) and Eqn. (20).

$$\sum_{e \in E} x^{CU}_{e,d} t_e + t^{sw} \sum_{e \in E} x^{CU}_{d,e} + \sum_{n \in N} T_{d,n} \leq \sum_{f \in F} \delta_{d,f} t^{split}_f, \quad \forall d \in D \tag{17}$$

$$\sum_{e \in E} x_{e,d} t_e + t^{sw} \sum_{e \in E} x_{d,e} + T^{DU}_d + \sum_{n \in N} T_{d,n} \leq \sum_{f \in F} \delta_{d,f} t^{serv}_f, \quad \forall d \in D \tag{18}$$

$$T^{DU}_d = \sum_{f \in F'} \lambda^q_d t^{DU}_f, \forall d \in D \tag{19}$$

$$T_{d,n} = y_{d,n} \sum_{d' \in D} \left(\sum_{f \in F'} \left(\delta_{d',f} t_f \right) \lambda^q_{d'} y_{d',n} \right), \forall d \in D, n \in N \tag{20}$$

Because Eqn. (20) involves a quadratic term, this optimization is a Mixed-Integer Quadratic Constraint Program, which cannot be solved by libraries such a CPLEX. Therefore, we linearize as follows to reduce it to a MILP:

$$0 \leq T_{d,n} \leq 1.5 y_{d,n}, \forall d \in D, n \in N \tag{21}$$

$$\sum_{d' \in D} \left(\sum_{f \in F'} (\delta_{d',f} t_f) \lambda^q_{d'} y_{d',n} \right) - 1.5(1 - y_{d,n}) \leq T_{d,n} \leq \sum_{d' \in D} \left(\sum_{f \in F'} (\delta_{d',f} t_f) \lambda^q_{d'} y_{d',n} \right), \forall d \in D, n \in N \tag{22}$$

C. Heuristic Algorithm

The problem described by the MILP model is a typical bin-packing problem, which is NP-hard [39]. Thus, we developed a greedy heuristic algorithm to place the baseband VNFs and establish lightpaths to minimize power consumption and ensure latency and capacity constraints compliance. The algorithm pseudo-code is described in Algorithm 2 and Algorithm 1.

Algorithm 1 gets as inputs the network graph $G(N, E)$ and the set of demands D. It starts by placing baseband VNFs in all nodes so that each one processes its incoming demands. It then routes the demands following the shortest path computed using Dijkstra algorithm, taking the number of hops as the weight metric. The next step consists of a greedy algorithm that searches a node among the neighbors to move the least utilized nodes. For this, it sorts the nodes reversely according to the node utilization U_n calculated as follows:

$$U_n = \frac{\sum_{d \in D} y_{d,n} \lambda_d \delta_{d,f} \pi_f}{C_n} \tag{23}$$

Then, it gets the set of demands D_{low} assigned to the node with the i^{th} lowest utilization n_{low}, i.e., where $y_{d,n_{low}} = 1$, and saves the current state of variables $x^{CU}_{d,e}$, $x_{e,d}$ and $y_{d,n}$ to $x^{CU,temp}_{d,e}$, $x^{temp}_{e,d}$ and $y^{temp}_{d,n}$. For each demand $d \in D_{low}$, the algorithm searches among the set of neighbors $N_{neighbors}$ for the node with the highest utilization that respects all constraints. Hence, for each node $n \in N_{sorted} \cap N_{neighbors}$, it tries to assign demand d to this node and sets $y_{d,n} = 1$ and $y_{d,n_{low}} = 0$. Then, it calculates the shortest path between n_d and n ($E(n_d, n)$), and between n and n_{gw} ($E(n, n_{gw})$), and it sets variables $x^{CU}_{e,d}$ and $x_{e,d}$ as follows:

$$x^{CU}_{e,d} = \begin{cases} 1, \text{if } e \in E(n_d, n) \\ 0, \text{otherwise} \end{cases} \tag{24}$$

$$x_{e,d} = \begin{cases} 1, \text{if } e \in E(n_d, n) \cup E(n, n_{gw}) \\ 0, \text{otherwise} \end{cases} \tag{25}$$

Next, it verifies if the solution respects capacity and latency constraints (Eqn. (16), Eqn. (17) and Eqn. (18)). In case the solution is not compliant with any of them, it sets the variables back to the previous solution ($x^{CU,temp}_{d,e}$, $x^{temp}_{e,d}$ and $y^{temp}_{d,n}$) and tries the next node $n \in N_{sorted} \cap N_{neighbors}$. When the algorithm finishes this procedure for all demands $d \in D_{low}$, it verifies whether the node is empty ($\sum_{d \in D_{low}} y_{d,n_{low}} = 0$) and sets $w_{n_{low}} = 0$ to indicate it is not used. If at least one demand is assigned to this node, it sets the variables back to the previous solution for all demands. In addition, it increments i to evaluate the next node with the lowest utilization. The stop condition is when i is over the length of set N_{sorted}.

Algorithm 2 provides a solution that consumes the least energy while respecting all constraints. For this, it computes the baseband VNF placement using the mean traffic λ^m_d. Then, it calculates the total power consumption and utilization following Eqn. (7) and Eqn. (23) applying the quantile traffic (λ^q_d). After this, it computes the baseband VNF placement and power consumption using the quantile traffic λ^q_d. If the power consumed by the mean placement (P_{mean}) is lower than the quantile (P_{quant}) and the maximum utilization U_{mean} does

Algorithm 1 Baseband VNF Placement and Routing

Input: Network graph $G(N, E)$, demands D, traffic λ_d
Output: CU assignment $y_{d,n}$, w_n, traffic routing $x_{d,e}$, $x^{CU}_{d,e}$

for $d \in D$ **do**
  Assign demand d to node n_d where demand d starts: $y_{d,n_d} \leftarrow 1$, $w_{n_d} \leftarrow 1$
  $x_{e,d} \leftarrow 1$ if e is in shortest path between n_d and n_{gw}
end for
$i \leftarrow 1$
while true do
  Sort nodes reversely based on their utilization: $N_{sorted} \leftarrow \text{sort}(N, U_n)$
  $n_{low} \leftarrow$ node with i^{th} lowest utilization
  $D_{low} \leftarrow$ demands assigned to n_{low}
  $x^{CU,temp}_{d,e} \leftarrow x^{CU}_{d,e}$, $x^{temp}_{e,d} \leftarrow x_{e,d}$ $y^{temp}_{d,n_{low}} \leftarrow y_{d,n_{low}}, \forall d \in D_{low}, \forall e \in E$
  for all $d \in D_{low}$ **do**
    for all $n \in N_{sorted} \cap N_{neighbor}$ **do**
      $y_{d,n} \leftarrow 1$, $y_{d,n_{low}} \leftarrow 0$
      $x^{CU}_{e,d} \leftarrow$ Eqn. (24)
      $x_{e,d} \leftarrow$ Eqn. (25)
      if Eqn. (16), Eqn. (17) **or** Eqn. (18) not satisfied **then**
        $x^{CU}_{d,e} \leftarrow x^{CU,temp}_{d,e}$, $x_{e,d} \leftarrow x^{temp}_{e,d}$ $y_{d,n_{low}} \leftarrow y^{temp}_{d,n_{low}}, \forall e \in E$
        next n
      end if
    end for
  end for
  if $\sum_{d \in D_{low}} y_{n_{low},d} = 0$ **then**
    $w_{n_{low}} \leftarrow 0$
  else
    $x^{CU}_{d,e} \leftarrow x^{CU,temp}_{d,e}$, $x_{e,d} \leftarrow x^{temp}_{e,d}$ $y_{d,n_{low}} \leftarrow y^{temp}_{d,n_{low}}, \forall d \in D_{low}, \forall e \in E$
    $i \leftarrow i + 1$
    if $i > |N_{sorted}|$ **then**
      break
    end if
  end if
end while

Algorithm 2 Baseband VNF Placement Based on Multi-Task Traffic Prediction

Input: Network graph $G(N, E)$, demands D, mean and quantile predicted traffic λ_d^m, λ_d^q

Output: CU assignment $y_{d,n}$, w_n, traffic routing $x_{d,e}$

1: Get baseband VNF placement with mean traffic λ_d^m: $y_{d,n}^{mean}, w_n^{mean}, x_{d,e}^{mean}, x_{d,e}^{CU,mean} \leftarrow$Algorithm 1

2: $P_{mean} \leftarrow$ Eqn. (7)$(y_{d,n}^{mean}, w_n^{mean}, x_{d,e}^{mean}, \lambda_d^q)$
$U_{mean} \leftarrow$ Eqn. (23)$(y_{d,n}^{mean}, \lambda_d^q)$

3: Get baseband VNF placement with quantile traffic λ_d^q: $y_{d,n}^{quant}, w_n^{quant}, x_{d,e}^{quant}, x_{d,e}^{CU,quant} \leftarrow$Algorithm 1

4: $P_{quant} \leftarrow$ Eqn. (7)$(y_{d,n}^{quant}, w_n^{quant}, x_{d,e}^{quant}, \lambda_d^q)$

5: **if** $P_{mean} < P_{quant}$ **and** $\max(U_{mean}) \leq 1$ **then**

6: $y_{d,n} \leftarrow y_{d,n}^{mean}, x_{d,e} \leftarrow x_{d,e}^{mean}$, $x_{d,e}^{CU} \leftarrow x_{d,e}^{CU,mean}, w_n \leftarrow w_n^{mean}, \forall d \in D, \forall e \in E, \forall n \in N$

7: **else**

8: $y_{d,n} \leftarrow y_{d,n}^{quant}, x_{d,e} \leftarrow x_{d,e}^{quant}$, $x_{d,e}^{CU} \leftarrow x_{d,e}^{CU,quant}, w_n \leftarrow w_n^{quant}, \forall d \in D, \forall e \in E, \forall n \in N$

9: **end if**

not surpass the node capacity, the algorithm returns the mean placement. Otherwise, it outputs the quantile placement.

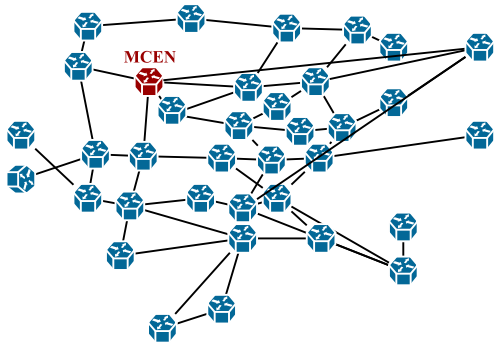


Fig. 4. Metro-area network topology with 35 AMENs and one MCEN.

TABLE II
SIMULATION INPUT PARAMETERS

Parameters	Description	Value
η_n	# servers at AMENs	1
$\eta_{n_{sw}}$	# servers at MCEN	70
C_s	Server capacity	534.6 GFLOPS
f_{CPU}	CPU frequency	3.7 GHz
P_{idle}^s	Server idle power consumption	130 W
P_{max}^s	Server maximum power consumption	870 W
n_a	Number of antennas	2
L	MIMO layers	2
M_b	QAM modulation	64
C	Code rate	1
R	Resource blocks	100
P_{max}^t	Transponder power consumption	110.4 W
C_e	Wavelengths per fiber	80

IV. RESULTS

This section discusses the performance of the prediction and optimization algorithms presented in Section III. First, we describe the simulation environment, detailing the parameters used. Next, we describe the outcomes of the traffic prediction. After this, we evaluate the robustness of the baseband VNF placement with the predicted traffic. Then, we compare the results of the network optimization using the predicted traffic to the optimal MILP solution and to two baseline scenarios. Finally, we analyze the computational time and model size.

A. Implementation and Simulation Environment

For the simulations, we consider a network topology inspired by a metro network from Metro-Haul project [6] depicted in Fig. 4. It contains a total of 36 nodes, among which one MCEN and 35 AMENs. The MCEN is the only node connected to the core and is considered to be the gateway node. Instead, the AMENs are treated as potential CU nodes. Each AMEN is equipped with a Intel Xeon Gold 6134 servers with 8 cores operating at 3.7 GHz and with a processing capacity C_s of 537.6 GFLOPS. The total capacity in each node is obtained by multiplying the number of servers per node by the server processing capacity. The idle power consumption P_{idle}^s is 130 W, and we consider that it represents 15% of the maximum power P_{max}^s, which is set to 870 W. Also, the MCEN contains 70 servers with the same specifications.

As anticipated, we assume that each AMEN is connected to a set of DU nodes serving the local RRHs, whose radio configuration follows the description in [2]. This scenario consists of a 64QAM modulation, 2x2 MIMO, and a coding scheme of 1 in the uplink. We also consider that a single user per transmission time interval sends 100 resource blocks of data. Furthermore, the nodes are connected using optical fiber links with 80 wavelengths. Each node is equipped with 100 Gbit/s transponders, with a power consumption of 110.4 W [40]. Table II summarizes the input parameters used in the simulations.

We assume that the topology of Fig. 4 represents a realistic metro-network infrastructure of a network operator. We combined this topology with the OpenCellid database [41] to map the RRHs of the city of Milan to the nodes. We also incorporated the TIM Big Data Challenge dataset [42] to obtain the traffic per node. This dataset describes the anonymized Call Detail Record over two months, indicating the interaction of users with the network through SMS, phone calls, and Internet connections every ten minutes. As explained in Section III, we assume that the uRLLC and eMBB traffic are modeled as VoIP and content delivery services, respectively. Therefore, we apply the phone calls and Internet connections from the TIM Big Data Challenge dataset to the uRLLC and eMBB demands, respectively. Furthermore, we assume that traffic information regarding uRLLC and eMBB services comes from distinct cells to avoid coordination issues when using different splits for each service. We divided the weekday traffic information into three parts: the training set with 20 consecutive weekdays, the validation set with 5 days, and the test set with 3 days.

TABLE III
HYPERPARAMETERS OF FFNN, LSTM AND LINEAR REGRESSION ALGORITHMS FOR THE TWO- AND SINGLE-STEP APPROACHES

		Two step	Single step
FFNN	Layers	2	2
	Neurons	25, 115	140, 85
	Activation	ReLu, ReLu, tanh	ReLu, tanh, ReLu
LSTM	Layers	2	2
	Neurons	50, 130	130, 70
LR	Degree	18	1

The traffic prediction tool was developed using the Keras library from Python. The MILP formulation was modeled using Net2Plan open-source network planner with the NFV over IP over WDM library [43]. We ran the simulations on a CPU Intel Core i7 with 32GB of RAM, 64-bit Windows 10.

B. Traffic Prediction

This section compares the results for the traffic prediction. As explained in Section III-A, the machine learning algorithms use the hour of the day to predict the normalized traffic, which is then scaled to obtain the total traffic. First, we show the results for the mechanisms proposed for computing the scaling factor. Then, we describe the performance of traffic prediction. In order to assess the performance of the proposed method, we compare its results to another state of the art model to predict the mean traffic proposed by Alvizu *et al.* [44]. That paper approach uses the following features: day, hour, average traffic of the previous days, traffic value for the same hour of seven days before, and traffic value for the same hour of the previous day. In contrast to the two-step approach proposed in this paper, we refer to the traffic prediction of [44] as *single-step*.

We selected the algorithms hyperparameters using the hyperopt algorithm [45] to minimize the MSE over the validation set. Even if the nodes and the services differ in their traffic patterns, they present fairly similar characteristics. For this reason, we optimized the hyperparameters considering the eMBB traffic of a single central node. According to the hyperparameter optimization results, we built the FFNN, LSTM, and linear regression models for the two- and single-step approaches, as shown in Table III.

Next, we trained the Linear Regression, LSTM, and FFNN algorithms in all scenarios using Adam optimizer [46] and MSE as the loss function. To assess the algorithms accuracy on the test set, we use the coefficient of determination (R^2), Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) (Eqn. (26)).

$$R^2 = 1 - \frac{\sum(Y_i - \hat{Y}_i)^2}{\sum(Y_i - \frac{1}{n}\sum Y_i)^2} \tag{26a}$$

$$RMSE = \sqrt{\frac{1}{n}\sum(Y_i - \hat{Y}_i)^2} \tag{26b}$$

$$MAPE = \frac{1}{n}\sum\left(\frac{|Y_i - \hat{Y}_i|}{Y_i}\right) \tag{26c}$$

where Y_i is the real value, $\hat{Y}_i$ is the predicted value, and n is the number of samples.

TABLE IV
RMSE AND MAPE OF PERSISTENCE FORECAST AND SMA

		Persistent		SMA	
		RMSE	MAPE	RMSE	MAPE
eMBB	Mean	11.856	0.031	13.691	0.044
	Max	29.719	0.068	36.169	0.118
uRLLC	Mean	1.191	0.025	1.171	0.034
	Max	2.875	0.066	2.532	0.151

TABLE V
R^2 SCORE AND RMSE OF NORMALIZED, SCALED AND SINGLE-STEP TRAFFIC PREDICTION FOR EMBB AND URLLC SERVICES

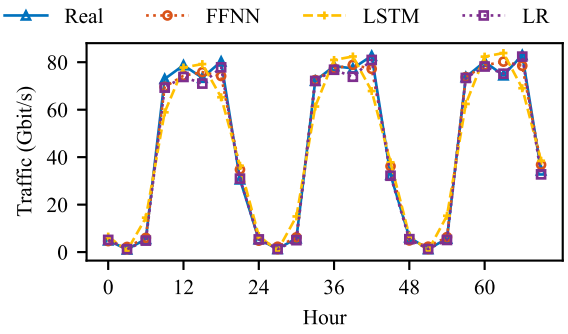
		FFNN		LSTM		LR	
		R^2	RMSE	R^2	RMSE	R^2	RMSE
eMBB	Normalized	0.86	0.005	0.79	0.005	0.89	0.004
	Scaled	0.91	0.965	0.85	1.156	0.95	0.745
	Single step	0.75	0.993	0.84	0.907	0.78	1.213
uRLLC	Normalized	0.96	0.007	0.92	0.009	0.98	0.004
	Scaled	0.97	0.205	0.93	0.298	0.99	0.099
	Single step	0.98	0.151	0.98	0.146	0.98	0.141

Table IV describes the RMSE and MAPE for eMBB and uRLLC services using the persistent forecast strategy and SMA over 5 days for the test set. The mean value represents the average traffic forecast error over all nodes, and the maximum value indicates the highest error among all nodes.

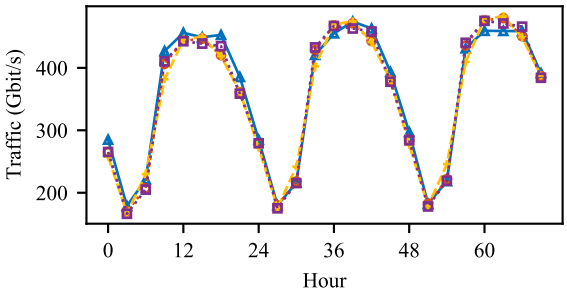
By applying the SMA over the previous 5 days as the scaling factor, the RMSE over the three days for all nodes is under 14 Gbit/s and 1.2 Gbit/s for eMBB and uRLLC traffic, respectively. This corresponds to an average MAPE of 4.4% (eMBB) and 3.4% (uRLLC). Persistent forecast, instead, enables reducing the error in both traffic scenarios, with the maximum MAPE dropping to 6.8% for eMBB and to 6.6% for uRLLC. Furthermore, evaluating the MAPE over the complete dataset, the maximum error for a node is under 5.5% with persistent forecast, while the SMA surpasses 7.5%. The low MAPE score confirms the slow day-by-day traffic variation, corroborating our idea of using it as a scaling factor.

We used the persistent forecast strategy to obtain the final expected traffic in the two-step prediction because it achieves the lowest error. Table V summarizes the R^2 scores and the RMSE of the normalized and final scaled traffic (i.e., the normalized traffic multiplied by the total traffic of the previous day), compared to the results of the single-step traffic prediction. The results in this subsection consider only the mean predicted value because the quantile regression provides an overestimation of the traffic; hence, it reduces the accuracy with respect to the real value.

Comparing the single-step traffic forecast to the approach proposed in this paper, the two-step prediction outperforms the former. The single-step technique provides high accuracy to compute the uRLLC traffic, surpassing the scores of the scaled traffic with LSTM. Nonetheless, the performance of the single-step algorithms significantly drops if used to forecast the eMBB traffic. The R^2 score decreases to less than 0.85 with any algorithm, and the RMSE surpasses 1 Gbit/s for the Linear Regression. The two-step approach, instead, presents significantly higher accuracy in the eMBB scenario when applying FFNN and Linear Regression, with RMSE values under 0.97 Gbit/s. Indeed, these algorithms provide an



(a) uRLLC service traffic.



(b) eMBB service traffic.

Fig. 5. Total predicted traffic using LSTM, FFNN and LR using the scaled normalized traffic compared to the real traffic.

accuracy of over 0.93 after the traffic was scaled. On the other hand, LSTM does not perform as well as the other algorithms, with accuracy values of 0.85 and 0.93 for the eMBB and uRLLC services, respectively. Evaluating the services separately, it is worth mentioning that the calls traffic shape in the dataset is more stable than Internet traffic. This is translated to the results presented in Table V. The uRLLC traffic achieves an accuracy over 0.97 when using the two-step prediction with FFNN and Linear Regression, and 0.93 with LSTM.

Figures 5(a) and 5(b) depict, respectively, the uRLLC and eMBB traffic of all nodes using the two-step traffic prediction in comparison to the real traffic. These graphs illustrate that the mispredictions are related to the occurrence of unpredictable events that generate a sudden traffic growth or reduction of traffic at certain time slots. Indeed, the traffic predicted with the machine learning algorithms draws a smoother traffic curve, not detecting unexpected traffic spikes.

The results in this subsection prove that our model has a better performance than traditional single traffic prediction approaches. Moreover, because the model requires uniquely the hour of the day and the total traffic received in the previous day, we can substantially simplify the pre-processing of the acquired data.

C. Robustness of Optimization With Traffic Prediction

This subsection evaluates the robustness of the optimization when using the traffic obtained in Section IV-B in the MILP. In particular, we selected the two-step multi-task FFNN algorithm for the network optimization.

First, we follow the technique widely used in the literature [19], [20], [23], which blindly uses the predicted traffic into the optimization framework to determine the future baseband VNF placement. We ran the optimization algorithm applying the mean forecast traffic (λ_d^m) to the objective function and constraints of the MILP. Then, we applied the real traffic data to understand whether the baseband VNF placement is feasible in the actual scenario. Analyzing the power consumption of the network configuration using the forecast traffic, we observed a slight decrease in power consumption (at most 0.27% at certain time slots) with respect to the optimal solution. This result clearly shows a misplacement of the baseband functions, as the solution calculated using the real traffic is optimal. Indeed, analyzing other system aspects carefully, we noticed that the node utilization surpasses its capacity when the real traffic is applied to the computed placement, reaching more than 110% with the mean FFNN traffic. Despite the high accuracy of the traffic prediction, it underestimates the traffic in almost all nodes. Consequently, simply applying the mean forecast traffic as done in most research works available in the literature does not guarantee a robust solution for the baseband VNF placement and, thus, cannot be directly used in this scenario. To overcome this issue, we evaluated two solutions: adding an extra artificial capacity buffer and applying the multi-task prediction.

1) Artificial Buffer: Consists of varying the total node capacity perceived by the MILP model. This technique is frequently used in the literature to ensure that the predicted resource allocation is compliant with the constraints in the real-time scenario. It is a similar solution to the one proposed by Guerra-Gomez *et al.* in [24], and hereinafter we refer to this setup as buffer. The algorithm computes the baseband VNF placement assuming that the capacity over the nodes is lower than the actual value, i.e., considering a capacity buffer of *X%*. Then, the real traffic is applied to the computed configuration with the full node capacity. By adding a buffer of 5%, i.e., reducing the computing capacity to 95%, the baseband VNF placement complies in most time-slots with the total node capacity. However, some time-slots require higher computational capacity installed in the nodes, with the node utilization arriving to 100.4%. Therefore, we must assume a lower capacity threshold, with a buffer of 10%. Indeed, the solution obtained becomes compliant with all the constraints at all time-slots when applying the actual traffic. These results show that we can obtain a risk-averse model that operators can use without loss of QoS by adding an artificial buffer to the optimization constraints. Nevertheless, this solution provides a static modification of the system, not considering the fact that each node presents different incoming traffic volumes. Furthermore, not all time-slots present the violation of the computing capacity constraint.

2) Multi-Task Prediction: Consists of using the mean traffic to estimate the costs in the objective function and the quantile prediction to ensure constraints compliance as explained in Section III-A. This solution improves the robustness and considers the traffic variability. Fig. 6 depicts the total real traffic and the predictions considering the mean, the 75th percentile and the 85th percentile. These figures show that the greatest is the quantile value, the more the traffic is overestimated. This translates into an overall RMSE over the total traffic of

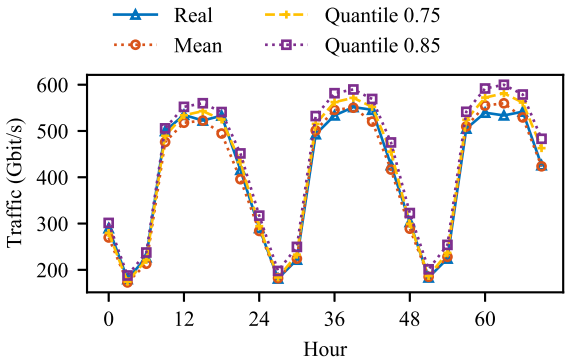
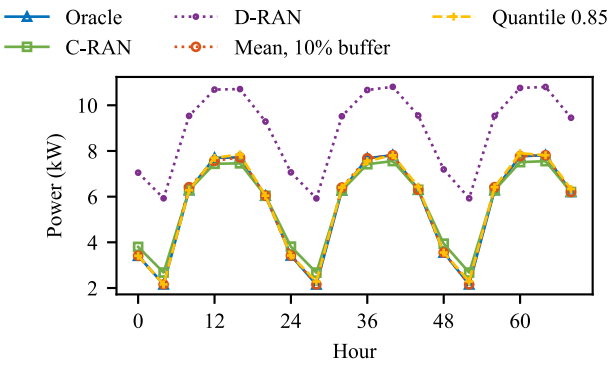


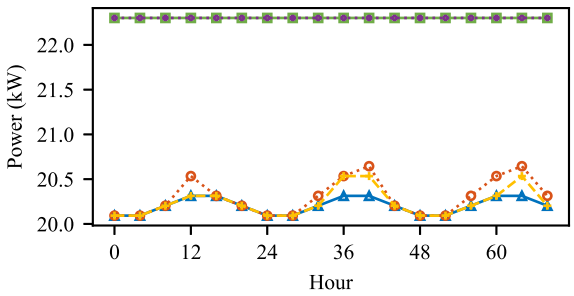
Fig. 6. Total traffic predicted with mean, 75th and 85th percentile compared to the real traffic.

TABLE VI
MAXIMUM NODE UTILIZATION WHEN REDUCING NODE CAPACITY AND WHEN APPLYING QUANTILE PREDICTION TO THE CONSTRAINTS

Prediction in constraints	Node capacity buffer	Maximum node utilization
Mean	–	1.101
Mean	5%	1.008
Mean	10%	**0.956**
Quantile 0.75	–	1.04
Quantile 0.85	–	**0.992**



(a) Nodes power consumption.



(b) Network power consumption.

Fig. 7. Power consumed by the oracle, by the placement with predicted traffic using artificial buffer and quantile regression, and by the baseline scenarios.

19.83 Gbit/s and 31.1 Gbit/s with 75th and 85th percentile predictions against 16.2 Gbit/s with the mean value. First, we trained the quantile model using the 75th percentile. The baseband VNF placement in this scenario reduces the overall node utilization; however, a single time-slot still requires more capacity. In particular, the placement violates the capacity of one node, whose utilization reaches 104%. Consequently, we trained the model using a greater quantile, i.e., 0.85. This approach enables the obtained solution to be feasible at all time-slots, reaching a maximum utilization of 99.2%.

Table VI summarizes the results when applying 5% and 10% artificial node capacity buffer and multi-task prediction with 75th and 85th quantile in the constraints.

This section shows that calculating the network configuration using the predicted traffic as-is does not provide a robust solution, as it does not respect the node capacity constraints when the actual traffic is applied. Hence, it is necessary to apply other techniques to guarantee a feasible baseband VNF placement with real traffic. For this, we proposed two solutions: add an artificial node capacity buffer and use quantile prediction in the MILP constraints. In the former, we added a 10% buffer to the node capacity constraints. The latter takes into account the traffic variability by using quantile regression to overestimate the predicted traffic.

D. MILP Optimization With Predicted Traffic

Based on the outcomes of Section IV-C, this section shows that the MILP with predicted traffic places the baseband VNF efficiently. For this, we compare the performance of the optimization algorithm using real traffic (hereinafter called *oracle*) to the traffic predicted by FFNN with the robustness modifications. We also compare the solution to two baseline scenarios: fully centralized and fully decentralized. The first corresponds to the earliest proposal of the baseband unit separation to achieve a completely centralized processing. For this reason, we call it the *C-RAN* scenario, and the MCEN is the only node containing the baseband VNFs. The second scenario designates the current RAN in 4G networks, i.e., the *D-RAN*, in which all AMENs in the topology described in Section IV-A host a CU and process the demands from the DUs to which they are connected. Both baseline scenarios route the demands on the path with the shortest propagation delay. All results in this section assume that the real traffic is applied to the different baseband VNF placement computation.

Fig. 7 presents the overall power consumption of the oracle, the baselines (C-RAN and D-RAN), the placement using the mean predicted traffic with 10% capacity buffer (Mean, 10% buffer) and using the multi-task prediction with 0.85 quantile (Quantile 0.85). The box plot in Fig. 8 illustrates the difference in power consumption with respect to the oracle.

Comparing the results to the baseline scenarios, the oracle enables reducing the overall power consumed on average 21.3% in comparison with the D-RAN. As expected, when processing is fully distributed, the AMENs are active at all times, leading to greater node power. Rather, the oracle baseband VNF placement maintains at most 11 nodes active simultaneously. As a result, the oracle reduces the IT power consumption by 76.93%. The C-RAN scenario, instead, presents the lowest consumption, with an average decrease of 6.46%, as a consequence of using a unique node. With respect to both baseline scenarios, the MILP reduces the total

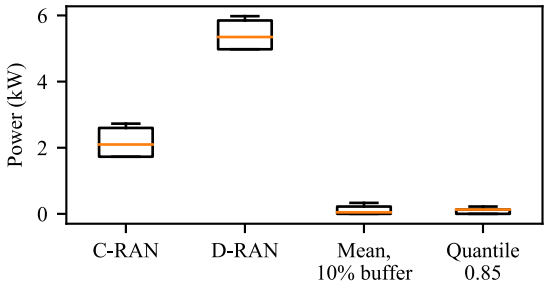


Fig. 8. Difference in power consumption with respect to the oracle of the placement based on the predicted traffic, and of the baseline scenarios.

TABLE VII
SPLIT AND SERVICE LATENCY FOR URLLC SERVICES. MAXIMUM ALLOWED LATENCY: 0.25 MS (SPLIT 6) AND 0.5 MS (URLLC SERVICE)

	Split latency (ms)			Service latency (ms)		
	min	max	mean	min	max	mean
Oracle	0.179	0.210	0.200	0.330	0.376	0.35
C-RAN	0.188	**0.283**	0.240	0.188	0.291	0.242
D-RAN	0.063	0.130	0.095	0.186	0.268	0.216
Mean, 10% buffer	0.179	0.230	0.204	0.330	0.399	0.361
Quantile 0.85	0.179	0.226	0.200	0.330	0.376	0.358

transponder consumption by more than 10% on average. This result shows that, although the baseline scenarios route the demands using the shortest path, they do not guarantee the use of the minimum amount of network resources.

Evaluating the results using the predicted traffic, we observe a mild increase in power compared to the oracle. More specifically, the consumption rose on average 0.39% (106 W) when applying the mean traffic prediction with 10% buffer. Using quantile regression, the increase represents 0.33% (87 W) on average; thus, we can save 2% with respect to the mean with 10% buffer. The solution with the quantile regression tends to use more nodes than the one with the buffer and the oracle. During the evaluated time slots, it uses on average 8.88 and at maximum 12 nodes, while these values decrease to 8.61/11 nodes with capacity buffer and to 8.5/11 nodes for the oracle. This translates into an increase of IT power consumption of 1.22% and 0.35% for the quantile and artificial buffer solutions with respect to the oracle. However, the network represents the most important component when computing the total power consumption. Since the solution using quantile prediction in the constraints uses fewer transponders, it enables finding a solution closer to the oracle.

We also evaluate the oracle, the baseline scenarios and the placement using the predicted traffic in terms of the maximum latency to understand whether the different scenarios comply with the service and split constraints. Table VII presents the minimum, maximum, mean of the demands with highest latency obtained for uRLLC service, and Table VIII show the results for the eMBB service.

The fully distributed scenario presents the lowest latency and the lowest deviation, reaching at most 130 μs and 268 μs of split and service delays for uRLLC services, and 338 μs and 1.67 ms for eMBB. This is explained by the fact that the CU is as close as possible to the DU, and it uses the shortest path to the gateway, leading to a modest propagation delay.

TABLE VIII
SPLIT AND SERVICE LATENCY FOR EMBB SERVICES. MAXIMUM ALLOWED LATENCY: 1.5 MS (SPLIT 2)AND 5 MS (EMBB SERVICE)

	Split latency (ms)			Service latency (ms)		
	min	max	mean	min	max	mean
Oracle	0.351	0.537	0.453	0.559	1.666	1.268
C-RAN	0.249	0.542	0.422	0.492	1.661	1.214
D-RAN	0.095	0.388	0.263	0.487	1.666	1.205
Mean, 10% buffer	0.351	0.509	0.438	0.559	1.618	1.245
Quantile 0.85	0.308	0.521	0.446	0.549	1.666	1.252

TABLE IX
POWER AND LATENCY RESULTS AT TIME-SLOT 40

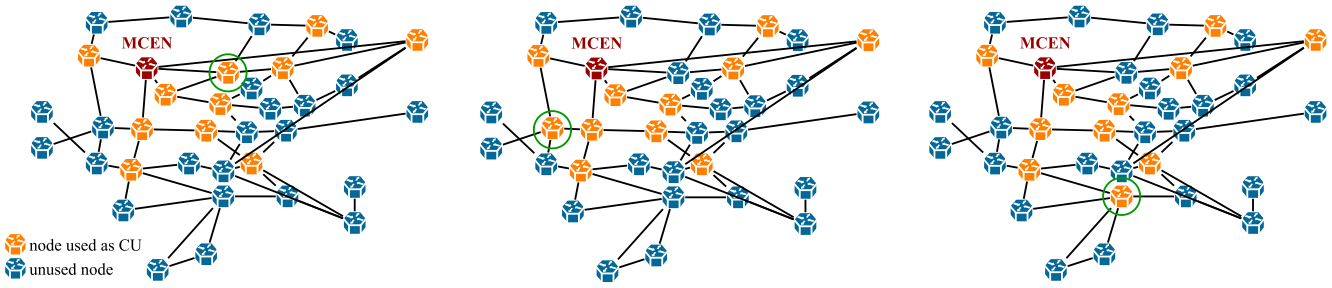
	Power (kW)			Split / Service latency (ms)	
	Total	IT	Network	uRLLC	eMBB
Oracle	28.13	7.82	20.31	0.209 / 0.375	0.510 / 1.552
Mean	28.46	7.82	20.64	0.229 / 0.399	0.432 / 1.552
Quantile	28.35	7.82	20.53	0.206 / 0.375	0.510 / 1.552

Moreover, each CU processes a few traffic demands, representing a small contribution as processing delay. In contrast, the C-RAN scenario reaches the highest latency values and greatest variability. The results of eMBB service show that, although its delay is higher than the D-RAN service latency, it is compliant with all constraints. However, as observed in bold in Table VII, it presents the highest variance and does not satisfy the split latency constraint of uRLLC service (250 μs) in several time-slots, with maximum latency of more than 280 μs. Hence, even if it enables reducing the overall power consumption, it penalizes the demands because of the latency.

The oracle baseband placement latency results present considerably higher values with respect to the D-RAN baseline scenario, mainly considering the uRLLC service. Nevertheless, it is compliant with the split and service latency constraints at all time slots: for the uRLLC service, the highest latency is 210 μs (split) and 376 μs (service), and for the eMBB service, it does not surpass 537 μs (split) and 1.67 ms (service). Comparing these results to the placement calculated using the predicted traffic, we observe very similar values. The solution with mean prediction and node capacity buffer presents more important differences, with maximum deviations of 11.4% (split 2), 9.9% (split 6), 2.8% (service 2) and 6.2% (service 6). These results confirm that the network configuration computed by the MILP for the predicted traffic presents a variation to the oracle solution. Nevertheless, we observe that the requirements are respected in all scenarios, i.e., the split 6, split 2, uRLLC, and eMBB delays are under the specified limits of 0.25 ms, 1.5 ms, 0.5 ms, and 5 ms, respectively.

Fig. 9 exemplifies the difference graphically in baseband VNF placement over the metro-network topology at 4 PM of the second day (time-slot 40). Table IX summarizes the power and latency results for each optimization.

The number of nodes that host the baseband VNFs is the same for all scenarios (11 nodes). Ten of these nodes remain the same, with a difference of a single node in each scenario. Because of the different settings, the power consumption of the placement with quantile and with mean prediction increases of 331.2 W and 220.8 W, respectively, compared to the oracle. Since the number of used nodes is the same, this deviation is



(a) oracle placement. (b) Placement with mean and buffer 10%. (c) Placement with multi-task.

Fig. 9. Baseband function placement over AMENs at 4 PM of the first day in the scenarios with the real traffic (a), with the mean prediction and artificial buffer (b) and with the multi-task prediction (c). The circled nodes highlight the nodes that are active only one of the scenarios.

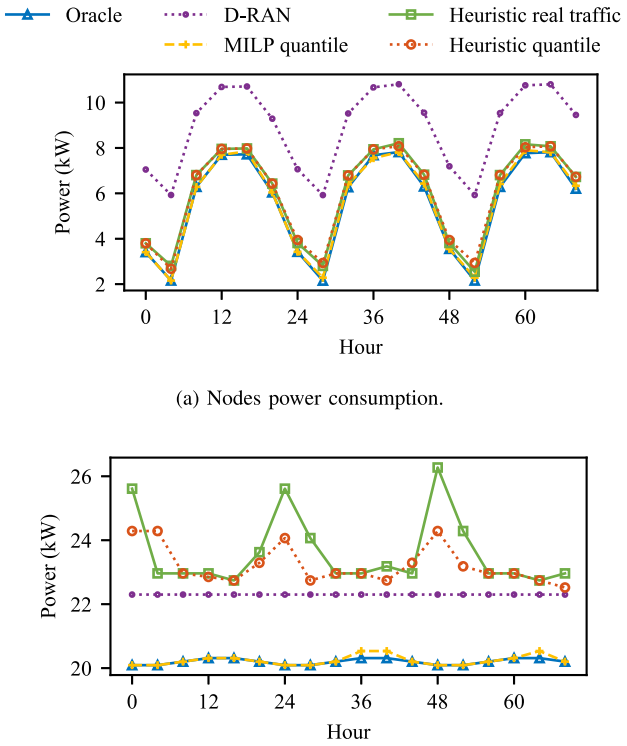
driven only by the transponder as the demands must be routed through different paths. The different routes also affect the split and service latency of each optimization. As a result, the placement computed using the mean predicted traffic introduces about 20 μs to the uRLLC split and service latency and reduces 75 μs from the eMBB split delay with respect to the oracle. On the other hand, the placement with quantile lowers the uRLLC split delay by only 3 μs, while the other latency values remain the same.

The results presented in this subsection show that the optimization proposed in Section III-B provides a solution that significantly reduces the power consumption while respecting all constraints. In addition, when applying the predicted traffic including the robustness modifications of Section IV-C, it is possible to achieve a quasi-optimal solution, being compliant with all requirements with a minor increase in power consumption. Comparing the two robustness proposals, quantile regression reaches a final result that is closer to the oracle placement even if it degrades the traffic prediction.

E. Heuristic Algorithm With Predicted Traffic

After evaluating the MILP model with predicted traffic, this section presents the results using the heuristic algorithm from Section III-C. For this, Figure 10 depicts the node and network power consumption of the oracle, the D-RAN scenario, the placement using MILP with the multi-task prediction with 0.85 quantile (MILP quantile), and the placement with the heuristic algorithm applying the real (Heuristic real traffic) and the multi-task prediction with 0.85 quantile (Heuristic quantile). Figure 11 illustrates the difference in power consumption of the same scenarios in a box plot compared to the oracle.

The heuristic algorithm with real traffic enables finding an intermediate solution between the oracle and the D-RAN scenario. It presents similar node power consumption compared to the oracle, with an increase of 10% on average. On the other hand, the network power consumption is significantly higher, with a maximum increment of 30.77%. This difference is because the node placement with the heuristic algorithm does not ensure that the CU is in a path among the shortest ones from the node receiving the demands and the gateway. Indeed, the network power is higher than the D-RAN scenario. Nevertheless, the overall power presents an average increase compared to the oracle of 15.2%, ensuring a better solution



(a) Nodes power consumption.

(b) Network power consumption.

Fig. 10. Power consumed by the oracle, by the D-RAN scenario, by the MILP placement with quantile regression, and by the heuristic algorithm with real and quantile predicted traffic.

during the entire day than the D-RAN, which is 21.27% higher than the oracle.

Applying the multi-task traffic prediction to the heuristic algorithm slightly decreases the maximum power consumption relative to real traffic. It consumes in total 1.3% less than the heuristic computed with the real traffic. Note that the heuristic algorithm does not compute the optimal solution; hence, the predicted traffic error further improves the final result. Compared to the MILP results, the multi-task-based heuristic algorithm raises the power consumption by 13.68% and 13.31% regarding the oracle and the MILP with multi-task traffic prediction, respectively. As in the heuristic with real traffic, this growth is a consequence of the higher network power.

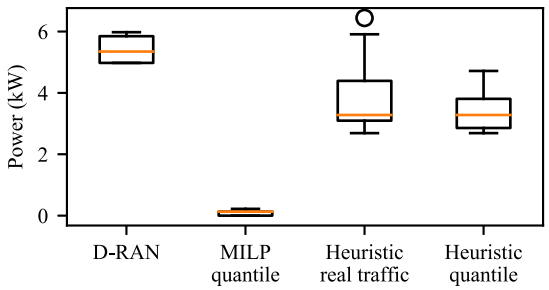


Fig. 11. Difference in power consumption with respect to the oracle of the D-RAN scenario, the MILP placement based on the predicted traffic, and the heuristic placement with the real and the predicted traffic.

TABLE X
HEURISTIC SPLIT AND SERVICE LATENCY. MAXIMUM ALLOWED LATENCY: 0.25 MS (SPLIT 6), 1.5 MS (SPLIT 2),0.5 MS (URLLC SERVICE), 5 MS (EMBB SERVICE)

		Split latency (ms)			Service latency (ms)		
		min	max	mean	min	max	mean
urLLC	Real	0.11	0.22	0.16	0.40	0.499	0.45
	Quantile	0.12	0.22	0.17	0.40	0.499	0.46
eMBB	Real	0.25	0.39	0.33	0.55	1.59	1.20
	Quantile	0.21	0.39	0.32	0.58	1.59	1.21

TABLE XI
TRAINING TIME AND MODEL SIZE FOR LINEAR REGRESSION, LSTM AND FFNN ALGORITHMS IN THE TWO- AND SINGLE-STEP APPROACHES

Algorithm	Two-step		Single-step	
	Time (s)	Size (MiB)	Time (s)	Size (MiB)
LR	0.544	0.027	1.002	0.021
LSTM	590.8	41.9	937.8	50.4
FFNN	211.9	2.32	242.7	5.82

Table X shows the minimum, maximum and mean split and service latency obtained for the uRLLC and eMBB services using the heuristic algorithm. Please refer to Tables VII and VIII for the latency results of the other approaches.

The maximum uRLLC-related latency values are very close to the accepted limit by the split 6 (0.25 ms) and the uRLLC service (0.5 ms). Nonetheless, the heuristic algorithm is always capable of providing a feasible solution. Concerning the eMBB service, since it presents very loose split and service latency budgets (1.5 ms and 5 ms), the algorithm using the real and the predicted traffic does not have any problem in finding a feasible solution. Therefore, the heuristic approach proposed is compliant with all requirements.

F. Computational Time

Table XI details the average training time and the model size of each algorithm analyzed in Section IV-B.

As expected, Linear Regression requires the shortest training time and occupies the least memory, followed by FFNN. Instead, LSTM is about 3 times slower and occupies at least 8 times more memory than FFNN. Comparing the two approaches analyzed in this section, the two-step traffic forecast training is considerably faster than the single-step prediction. Indeed, the time to train the one-step linear regression, LSTM, and FFNN increases 84%, 58%, and 15%, respectively, with respect to the two-step prediction. In addition, the approach proposed in this paper reduces significantly the model size of LSTM and FFNN, occupying less memory (17% and 60%, respectively). The model size of Linear Regression using the two-step approach is slightly larger than the one-step, with a difference of 7 kiB. Besides improving the prediction accuracy as shown in Section IV-B, these results prove that the proposed solution considerably improves the training time and the model size. Consequently, it facilitates the retraining for network operators and also the storage of these models.

As previously mentioned, MILP models with similar formulations to the one proposed in this paper are NP-hard. Thanks to the approach used in this paper of placing the baseband VNFs considering only the metro-network nodes, we can maintain the execution time low. The MILP spends on average 112.47 s to compute the hourly placement of baseband VNFs considering the proposed topology. Although the computing time is fairly low with the topology used in this paper, larger cities with a greater number of nodes could require many hours to compute a solution. As previously mentioned, the heuristic algorithm was developed to mitigate the NP-Hardness of MILP models. The heuristic algorithm proposed in this paper spends on average 182.61 ms to compute the solution for each time slot. Therefore, it is more than 600 times faster than the MILP.

V. FINAL REMARKS

This paper presents the optimization of baseband VNF placement in a metro network to minimize operators' costs. This approach envisages first the forecast of the traffic of different services and then the optimization of the network configuration. Exploiting the well-defined shape of the daily traffic, we developed a two-step multi-task predictive model prior to the baseband VNF optimization. First, Linear Regression, LSTM and FFNN algorithms predict the mean and quantile normalized traffic. Then, we use persistent forecast to determine the scaling factor to calculate the expected traffic. This information is then fed to a MILP formulation that computes the network configuration minimizing the power consumption, subject to the functional split and service requirements. In particular, we apply the mean predicted traffic to the objective function to estimate the costs and the quantile value to the constraints to ensure that the capacity bounds are respected. We also propose a heuristic algorithm to reduce the computational time. It analyzes the placement using the mean and the quantile traffic to minimize the power consumption and guarantee constraint compliance. The results show that the two-step traffic prediction using FFNN algorithm provides the best prediction accuracy and outperforms the state-of-the-art. We also demonstrate that the two-step multi-task traffic prediction in the optimization is the approach that achieves the most similar results with respect to the oracle placement, i.e., the placement calculated using the real traffic. Indeed, our solution enables achieving a robust solution capable of carrying the actual traffic at all time slots, and the power consumption is only 0.33% higher than the oracle. The heuristic algorithm significantly reduces the computational time at the expense of increasing the power consumption by 13%.

ACKNOWLEDGMENT

The authors would like to thank Dr. Marco Quagliotti for his valuable support.

REFERENCES

[1] L. M. Larsen, A. Checko, and H. L. Christiansen, "A survey of the functional splits proposed for 5G mobile crosshaul networks," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 146–172, 1st Quart., 2018.

[2] *Small Cell Virtualization Functional Splits and Use Cases*, Small Cell Forum, Dursley, U.K., Jan. 2016.

[3] *Radio Access Architecture and Interfaces (Release 14)*, 3GPP, Sophia Antipolis, France,, 2017.

[4] L. M. M. Zorello, S. Troia, M. Quagliotti, and G. Maier, "Power-aware optimization of baseband-function placement in cloud radio access networks," in *Proc. IEEE/IFIP Int. Conf. Opt. Netw. Design Model.*, 2020, pp. 1–6.

[5] Z. Zhong, N. Hua, H. Liu, Y. Li, and X. Zheng, "Considerations of effective tidal traffic dispatching in software-defined metro IPover optical networks," in *Proc. IEEE Opto Electron. Commun. Conf.*, 2018, pp. 1–9.

[6] "Metro-Haul." [Online]. Available: https://metro-haul.eu/ (Accessed: May 2022).

[7] A. Tzanakaki, M. P. Anastasopoulos, and D. Simeonidou, "Optical, wireless, and data center network infrastructures for 5G services," *J. Opt. Commun. Netw.*, vol. 11, no. 2, pp. A111–A122, 2019.

[8] H. Yu, F. Musumeci, J. Zhang, Y. Xiao, M. Tornatore, and Y. Ji, "DU/CU placement for C-RAN over optical metro-aggregation networks," in *Proc. IEEE/IFIP Int. Conf. Opt. Netw. Design Model.*, 2019, pp. 82–93.

[9] A. N. Al-Quzweeni, A. Q. Lawey, T. E. H. Elgorashi, and J. M. H. Elmirghani, "Optimized energy aware 5G network function virtualization," *IEEE Access*, vol. 7, pp. 44939–44958, 2019.

[10] F. W. Murti, A. Garcia-Saavedra, X. Costa-Perez, and G. Iosifidis, "On the optimization of multi-cloud virtualized radio access networks," in *Proc. IEEE Int. Conf. Commun.*, 2020, pp. 1–7.

[11] R. I. Tinini, D. M. Batista, G. B. Figueiredo, M. Tornatore, and B. Mukherjee, "Energy-efficient vBBU migration and wavelength reassignment in cloud-fog RAN," *Trans. Green Commun. Netw.*, vol. 5, no. 1, pp. 18–28, 2021.

[12] J. Yusupov, A. Ksentini, G. Marchetto, and R. Sisto, "Multi-objective function splitting and placement of network slices in 5G mobile networks," in *Proc. IEEE Conf. Stand. Commun. Netw.*, 2018, pp. 1–6.

[13] B. Ojaghi, F. Adelantado, E. Kartsakli, A. Antonopoulos, and C. Verikoukis, "Sliced-RAN: Joint slicing and functional split in future 5G radio access networks," in *Proc. IEEE Int. Conf. Commun.*, 2019, pp. 1–6.

[14] S. Matoussi, I. Fajjari, N. Aitsaadi, and R. Langar, "User slicing scheme with functional split selection in 5G cloud-RAN," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2020, pp. 1–8.

[15] H. Gupta, M. Sharma, A. Franklin, and B. R. Tamma, "Apt-RAN: A flexible split-based 5G RAN to minimize energy consumption and handovers," *IEEE Trans. Netw. Service Manag.*, vol. 17, no. 1, pp. 473–487, Mar. 2020.

[16] R. Singh, C. Hasan, X. Foukas, M. Fiore, M. Marina, and Y. Wang, "Energy-efficient orchestration of metro-scale 5G radio access networks," in *Proc. IEEE Int. Conf. Comput. Commun.*, 2021, pp. 1–10.

[17] Y. Xiao, J. Zhang, and Y. Ji, "Energy-efficient DU-CU deployment and Lightpath provisioning for service-oriented 5G metro access/aggregation networks," *J. Lightw. Technol.*, vol. 39, no. 17, pp. 5347–5361, Sep. 1, 2021.

[18] T. Sigwele, Y. Hu, and M. Susanto, "Energy-efficient 5G cloud RAN with virtual BBU server consolidation and base station sleeping," *Comput. Netw.*, vol. 177, Aug. 2020, Art. no. 107302.

[19] A. Pelekanou, M. Anastasopoulos, A. Tzanakaki, and D. Simeonidou, "Provisioning of 5G services employing machine learning techniques," in *Proc. IEEE/IFIP Int. Conf. Opt. Netw. Design Model.*, 2018, pp. 200–205.

[20] H. Yu, F. Musumeci, J. Zhang, M. Tornatore, L. Bai, and Y. Ji, "Dynamic 5G RAN slice adjustment and migration based on traffic prediction in WDM metro-aggregation networks," *J. Opt. Commun. Netw.*, vol. 12, no. 12, pp. 403–413, 2020.

[21] Z. Gao *et al.*, "Deep reinforcement learning-based policy for baseband function placement and routing of RAN in 5G and beyond," *J. Lightw. Technol.*, vol. 40, no. 2, pp. 470–480, Jan. 15, 2022.

[22] M. Zhu, J. Gu, T. Shen, C. Shi, and X. Ren, "Energy-efficient and QoS guaranteed BBU aggregation in CRAN based on heuristic-assisted deep reinforcement learning," *J. Lightw. Technol.*, vol. 40, no. 3, pp. 575–587, Feb. 1, 2022.

[23] L. Chen, T. M. T. Nguyen, D. Yang, M. Nogueira, C. Wang, and D. Zhang, "Data-driven C-RAN optimization exploiting traffic and mobility dynamics of mobile users," *IEEE Trans. Mobile Comput.*, vol. 20, no. 5, pp. 1773–1788, May 2021.

[24] R. Guerra-Gomez, S. Ruiz-Boque, M. Garcia-Lozano, and J. O. Bonafe, "Machine learning adaptive computational capacity prediction for dynamic resource management in C-RAN," *IEEE Access*, vol. 8, pp. 89130–89142, 2020.

[25] H. Zhang, Y. Hua, C. Wang, R. Li, and Z. Zhao, "Deep learning based traffic and mobility prediction," in *Machine Learning for Future Wireless Communications*, F. Luo, Ed. Hoboken, NJ, USA: Wiley, 2020, ch. 7, pp. 119–136.

[26] D. Andreoletti, S. Troia, F. Musumeci, S. Giordano, G. Maier, and M. Tornatore, "Network traffic prediction based on diffusion convolutional recurrent neural networks," in *Proc. IEEE Conf. Comput. Commun. Workshops*, 2019, pp. 246–251.

[27] D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Perez, "DeepCog: Optimizing resource provisioning in network slicing with AI-based capacity forecasting," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 2, pp. 361–376, Feb. 2020.

[28] *IMT Vision—Framework and Overall Objectives of the Fugure Development of IMT for 2020 and Beyond*, ITU, Geneva, Switzerland, 2015.

[29] C. Song *et al.*, "Hierarchical edge cloud enabling network slicing for 5G optical Fronthaul," *J. Opt. Commun. Netw.*, vol. 11, no. 4, pp. B60–B70, Apr. 2019.

[30] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, "Limits of predictability in human mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, Feb. 2010.

[31] T. Kohonen, "An introduction to neural computing," *Neural Netw.*, vol. 1, no. 1, pp. 3–16, 1988.

[32] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[33] M. Shehata, A. Elbanna, F. Musumeci, and M. Tornatore, "Multiplexing gain and processing savings of 5G radio-access-network functional splits," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 4, pp. 982–991, Dec. 2018.

[34] B. Debaillie, C. Desset, and F. Louagie, "A flexible and future-proof power model for cellular base stations," in *Proc. IEEE Veh. Technol. Conf.*, 2015, pp. 1–7.

[35] F. Musumeci, O. Ayoub, M. Magoni, and M. Tornatore, "Latency-aware CU placement/handover in dynamic WDM access-aggregation networks," *J. Opt. Commun. Netw.*, vol. 11, no. 4, pp. B71–B82, 2019.

[36] A. D. Domenico, Y. Liu, and W. Yu, "Optimal computational resource allocation and network slicing deployment in 5G hybrid C-RAN," in *Proc. IEEE Int. Conf. Commun.*, 2019, pp. 1–6.

[37] X. Wang, L. Wang, S. E. Elayoubi, A. Conte, B. Mukherjee, and C. Cavdar, "Centralize or distribute? A techno-economic study to design a low-cost cloud radio access network," in *Proc. IEEE Int. Conf. Commun.*, 2017, pp. 1–7.

[38] *Technical Specification Group Services and System Aspects (Release 16)*, 3GPP, Sophia Antipolis, France, 2020.

[39] M. Qian, W. Hardjawana, J. Shi, and B. Vucetic, "Baseband processing units virtualization for cloud radio access networks," *IEEE Wireless Commun. Lett.*, vol. 4, no. 2, pp. 189–192, Apr. 2015.

[40] J. M. H. Elmirghani *et al.*, "GreenTouch GreenMeter core network energy-efficiency improvement measures and optimization," *J. Opt. Commun. Netw.*, vol. 10, no. 2, pp. A250–A269, 2018.

[41] U. Labs. "OpenCellid." [Online]. Available: http://opencellid.org/ (Accessed: Sep. 2019).

[42] T. Italia. "Big data challenge." 2019. [Online]. Available: https://dandelion.eu/datamine/open-big-data/ (Accessed: Sep. 2019).

[43] J. L. Romero-Gázquez, M. Garrich, F. M. Muro, M. B. Delgado, and P. P. Mariño, "NIW: A Net2Plan-based library for NFV over IP over WDM networks," in *Proc. Int. Conf. Transp. Opt. Netw.*, 2019, pp. 1–4.

[44] R. Alvizu, S. Troia, G. Maier, and A. Pattavina, "Matheuristic with machine-learning-based prediction for software-defined mobile metro-core networks," *J. Opt. Commun. Netw.*, vol. 9, no. 9, pp. D19–D30, 2017.

[45] J. Bergstra, D. Yamins, and D. D. Cox, "Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures," in *Proc. JMLR Int. Conf. Mach. Learn.*, 2013, pp. 115–123.

[46] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–6.