AI-based prediction for Ultra Reliable Low Latency service performance in industrial environments

Meriem MHEDHBI
Orange Innovation Networks
Chatillon, France

Salah ELAYOUBI
Université Paris Saclay, CentraleSupélec, CNRS, Laboratoire des Signaux et Systèmes
Gif-Sur-Yvette, France

Galaad LECONTE
Université Paris Saclay, CentraleSupélec, CNRS, Laboratoire des Signaux et Systèmes
Gif-Sur-Yvette, France

***Abstract*—This article investigates the usage of Artificial Intelligence (AI) techniques in the prediction of network performance for Industrial Internet of Things (IIoT). In industrial environments, 5G Ultra Reliable Low Latency Communications (URLLC) are intended for serving critical services with very stringent latency requirements, such as those involving collaborative robots. Even if the flexible 5G New Radio (NR) design is able to achieve the target IIoT performances, the necessary spectrum resources need to be available and reserved for URLLC. A Quality of Service (QoS) prediction scheme is thus needed for anticipating performance degradation and undertake necessary actions, such as network resource provisioning or application adaptation, e.g. by entering an adapted mode. We explore the design of AI algorithms for QoS prediction in industrial environments, and compare different tools for regression and classification, including Neural Networks (NN) and K Nearest Neighbors (K-NN). We explore prediction based on Signal to Interference and Noise Ratio (SINR), or simply based on the position of robots within the plant. As the latency degradation event is rare in general, we observe that the training data is highly imbalanced leading to a low prediction accuracy. We show how the prediction performance can be enhanced by importance sampling techniques and by a modified detection threshold in what we call M-KNN scheme.**

Keywords— Industrial IoT, QoS prediction, AI, URLLC

I. INTRODUCTION

5G is envisioned as a federating communication technology as it is designed to serve three main service categories that are eMBB for enhanced Mobile Broadband, massive Machine Type Communications (mMTC) and Ultra Reliable Low Latency Communications (URLLC). These latter services are suitable for Industrial Internet of Things (IIoT) applications, as they enable an overwhelming proportion of packets (99,999%) to be received within a latency budget ranging from 0.5 to 10 ms, depending on the use case [1]. These very stringent Quality of Service (QoS) requirements are made possible due to several features such as the flexible numerology of 5G New Radio (NR), punctured scheduling in the downlink and the enhanced retransmission schemes in the uplink that allow for blind and repeated retransmission of URLLC packets [2].

These advanced URLLC features have to be accompanied by an adequate provisioning of resources for URLLC slice as the available resources may not be sufficient in high traffic scenarios, leading to a degraded performance of the industrial system. In this context, a reactive scheme where resources are allocated to URLLC upon QoS degradation is not adequate and may lead to a violation of the reliability constraint, especially that the reconfiguration phase of the URLLC slice may take several tens of milliseconds [3]. A solution may be in over-provisioning of resources to the URLLC slice, but this is clearly inefficient and not always possible. We propose in this paper a QoS prediction framework that detects in advance the QoS degradation in order for enabling proactive schemes. On one hand, this will allow the provisioning of adequate resources for the URLLC slice, and on the other hand, in case of impossibility of provisioning sufficient resources in very high traffic regimes, this may allow the industrial system to enter an adapted mode that generates less traffic by relying on local information [4].

The problem of predicting QoS in wireless networks has attracted some attention in the past years. [5] used a Random Forest algorithm for predicting the achievable data rate for mobile broadband applications, based on measurement data including radio and network management measurements. When it comes to specific services, [6] considered the video streaming service and used machine learning tools, namely Generalized Linear Model (GLM) and Support Vector Machine (SVM) for predicting buffer starvation events, based on the user's radio conditions and the system load. [7] developed a random forest based algorithm that estimates the YouTube quality observing the IP packet arrivals. To the best of our knowledge, this work is the first one to consider AI-based prediction for URLLC.

Our proposed framework uses Artificial Intelligence (AI) to predict the packet loss due to delay violation. The considered scenario is Automated Guided Vehicles (AGV) in confined industrial environments where the application layer is able to send reliable trajectory and data traffic predictions to the network controller. The network controller then associates these traffic and trajectory information with network-related information to predict QoS in the near future state. The training phase is based on data from the network with known AGV locations and traffic profiles. We explore two sets of AI methods that are Neural Networks and K-Nearest Neighbors (K-NN) classifier. We explore the prediction schemes with different types of training data, ranging from network level parameters (Signal to Interference and Noise Ratio (SINR)) to application-level information (trajectory-based position estimation). As our target application is URLLC, classical AI performance metrics such the prediction accuracy are not adequate, as the detection of loss events remains far from being suitable for URLLC, even for a high prediction accuracy. This is due to two reasons. First, the loss event is rare for URLLC so that the training set does not contain sufficient data points for learning about loss conditions. And second, the AI methods are classically designed to predict the most likely outcome, while in the case of URLLC detecting risks is primordial. For coping with the rareness of the loss events, we make use of importance sampling for generating data that is less imbalanced. As of algorithm tuning, we modify the classical K-NN clustering algorithm and propose a new scheme, named (M,K)-NN that predicts a loss event when at least *M* among the *K* nearest neighbors is a loss. We show how the values of *M* and *K* can be tuned so that losses are detected with an acceptable price on false negatives.

The contributions of this paper are the following:

- We develop an AI-based QoS prediction framework for URLLC in industrial environments.
- We compare the performances of neural network and K-NN algorithms depending on the use case and the available network data.
- We propose an important sampling framework for accelerating the convergence of the learning phase.
- We propose an enhanced (M,K)-NN algorithm that is tailored towards detecting risks of losses in critical use cases.

The remainder of this paper is organized as follows. In section II, we describe the problem of QoS prediction in industrial environments and the corresponding data. Section III develops a Neural Network based method for predicting URLLC QoS when the SINR is known and shows how importance sampling enhances the prediction performance. Section IV considers a more realistic setting where only the positions of AGVs are available and not their SINR. We compare the NN performance with a modified K-NN algorithm. Section V discusses the impact of the proposed QoS prediction framework on the network management. Section VI eventually concludes the paper.

II. QoS PREDICTION SCENARIO

The predictive QoS is a mechanism that enables the mobile network to provide notifications about the potential QoS changes in order to adjust the system in advance. This feature is particularly interesting to use cases of critical nature. For instance the industry automation and automotive applications have very demanding QoS requirements and the network may not always guarantee the required QoS. In such cases, a notification in advance of a potential change in the QoS may avoid or reduce undesired behavior and damages. We present in this section the QoS prediction scenario and describe the dataset that will be the basis of our prediction framework.

A. Studied scenario

We consider a scenario of a typical factory including a number of wireless connected robots equipped with URLLC transceiver and communicate with a central controller via a number of gNodeBs that are deployed within the factory. Regarding the spectrum aspects, the mmWave 5G band is privileged in such scenario as it offers extreme capacity and very low latency [10].

As of the traffic pattern, we consider per-robot Poisson traffic generation with a 1 ms interval between two consecutive packets. Each packets is of size 12 Bytes. In critical IoT services, a latency budget is defined which corresponds to the maximum allowed delay to correctly receive and decode a packet. Beyond the latency budget the packet transmission is failed and the packet is considered as lost. In this study the latency budget is equal to 1 ms. For the link adaptation, the MCS is chosen so that the target BLER equal to 10^{-5}.

The goal is to predict the degradation of the quality of service of URLLC users in order to adapt the configuration of the network and / or the application function and the selected key performance indicator to trigger the adapted mode is the latency. The adopted Key Performance Indicator (KPI) is the maximal achieved latency for each user that is compared to a given threshold (depending on the latency budget). The prediction task is then a classification problem that predicts whether the latency exceeds the threshold or not.

B. Dataset acquisition and exploitation

Our dataset is acquired using a 3GPP-compliant system level simulator. Table 1 presents the detailed system parameters.

TABLE I. SYSTEM AND TRAFFIC PARAMETERS

Radio parameters	
Frequency band	26 GHz
TTD pattern	DUDU
Bandwidth	100 MHz
Sub carrier spacing	120 kHz
Mini slot size	7 OFDM symbols
Traffic parameters	
Traffic pattern	Poisson
Traffic intensity	1 packet/ms
Packet size	12 Bytes
Target radio latency	1 ms
Network parameters	
Number of gNodeBs	2
Number of sectors	6
Number of AGVs	5

The factory communication service area size is equal to 160 m*160 m and includes 5 robots with uniformly random positions. We consider a deployment of 2 ceiling-mounted gNodeBs located at height of 10 m and equipped each with three directional antennas. The channel model is based on the 3GPP Indoor Factory Dense clutter High BS (InF-DH) model defined in [9].The prediction module is fed by information from the network side about the SINR or the positions of the robots.

We conducted a large number of downlink system level simulations of the studied scenario to evaluate the performance of URLLC in terms of latency and packet loss probability. The above mentioned deployment options (frequency band, 2 gNodeBs in a small area…) and the low target BLER (used in link adaptation and physical layer abstraction) are adapted for critical IoT use cases, leading to a very low percentage of lost packets, as expected for URLLC.
During the simulations, we collected a training dataset that contains network level information (path loss, SINR, load) and QoS information (per packet average and maximal delay) associated to the different positions in the network.
This training data set is exploited to learn the association between network parameters and the QoS, as will be explained in the next section. In the exploitation phase, a subset of the input information is available to the QoS prediction module, provided by the application layer or measured by the network. Note that detailed architecture for QoS prediction and subsequent network management is presented in section V. Two practical options are studied:

- SINR available: In this option, detailed radio conditions are available when predicting the URLLC QoS, leading to known SINR. Note that the SINR embeds the path loss and the interferences and is thus expected to provide accurate predictions. However, when QoS is predicted based on SINR knowledge, degradation is already there or will closely occur, and any subsequent network reconfiguration scheme can be seen as *reactive*.
- Position available: In this option, only the position of the AGV is known when predicting the QoS. No radio information (SINR or interference) is known and the prediction may be less effective. However, such a prediction is suitable for proactive system reconfiguration, as the position may be obtained from trajectory predictions provided by the IIoT controller.
  The SINR and positions data represent two extreme prediction features and should give an insight about the lower and upper bounds of the prediction performance, and on the performances of the reactive vs. proactive schemes.

III. QoS Prediction Based on SINR

As state before, the choice of the SINR as input feature is motivated by the high correlation observed between the users SINR and the maximum achieved latency. This is illustrated by Figure 1 and Figure 2 representing the users' SINR values and the users' maximum latency, respectively. Compared both figures indicates that users with high latency values suffer from low SINR values.

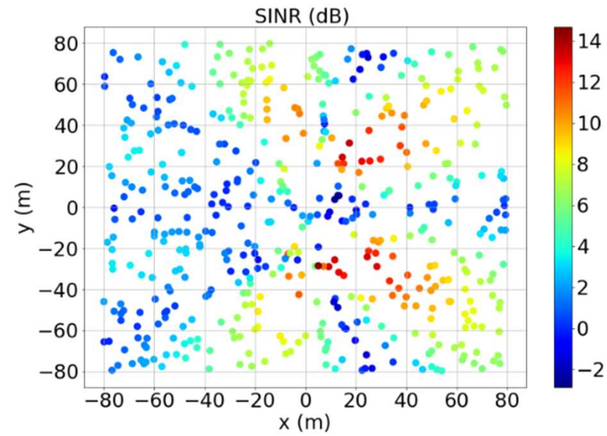

Figure 1: SINR map

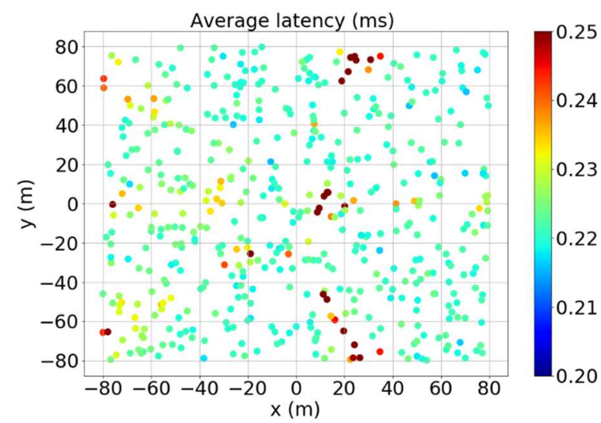

Figure 2: Latency map

A. *Neural Network prediction on the original dataset*

We start by a prediction technique based on neural network (NN) algorithms. The NN takes as input the predicted SINR of the user and provides as output a classification into two classes: a normal class where no latency degradation is predicted, and a degraded class.

We perform a fine tuning of the neural network hyper parameters. We evaluate different neural network parameters such as the number of hidden layers, ranging from 1 to 8 with pyramidal architecture, number of neurons and optimizers. The simulation data is split and used for training and testing. Throughout the remainder of the paper, Table II will present the prediction performances for the different models. Each line corresponds to a prediction model or scenario. The first two columns describe the input data (SINR or positions, simulation method). The next two columns describe the prediction model (principle and parameters) and the last columns give the prediction performance. Three metrics are considered:

- Accuracy: the percentage of successful predictions over the whole test set. We count as a success each time the prediction corresponds to what is observed in simulation.
- False Negatives (FN): the percentage of packets where a good delay is predicted, while there was an outage in reality.

TABLE II. PREDICTION RESULTS

	Input features	Data type		Prediction Algorithm	Prediction parameters	Prediction performance		
		Training	Test			Acc.	FN	FP
1	SINR	No IS	No IS	Neural network	Basic-NN: Layers: (16) Batch size: 1 Optimizer: SGD	99%	60%	0.1%
2	SINR	IS	No IS	Neural network	Basic-NN	98%	0%	2%
3	Positions	IS	No IS	Neural network	Basic-NN	83%	12%	16%
4	Positions	IS	No IS	Neural network	Layers: (64,64) Batch size: 16 Optimizer: RMSProp	97.4%	8.3%	2.5%
5	SINR	IS	No IS	Neural network	Layers: (64,64) Batch size: 16 Optimizer: RMSProp	98%	0%	2 %
6	Positions	IS	No IS	KNN	K = 1	96.4%	5.7%	3.6%
7	Positions	IS	No IS	KNN	K = 3	96.5%	5.7%	3.4%
8	Positions	IS	No IS	M-KNN	K = 3, M= 1	93.4%	0%	6.7%

- False Positives (FP): the percentage of packets where an outage is predicted, while the actual delay was below the threshold.

Our trials led us to the estimator that we call "NN-basic": a neural network with one hidden layer including 16 neurons and stochastic gradient descent optimizer. We have chosen this model because adding layers or neurons does not bring additional accuracy for the considered scenario.

Line 1 of Table II presents the performance of the basic model. The results present an interesting global accuracy of 99%. However the false negatives percentage (FN) is 60% meaning that only 40% of the outage cases were predicted . The reason behind the misdetection of the latency is the rareness of the loss event since the network deployment configuration is well adapted for the latency constraint. In our simulations, the latency problems happen in around 2% of cases. The data is thus largely imbalanced towards the class of "good QoS" that the NN tends to predict a good QoS in most of the cases.

B. Prediction using importance sampling data

To get around the problem of prediction bias towards the dominant event, an importance sampling (IS) approach was applied to the simulator. The IS process consists in biasing the simulator for generating more input data that belong to the rare event class. To do so, a weighting on the probability of occurrence was added, thus increasing the number of points in the problematic areas where latency issues occur the most often. The biasing consists in the following process:

- Start simulations with a uniform distribution of users' positions in the network.
- Each time a loss event is detected, the spatial distribution of the users' positions is biased by increasing the probability of being generated around the area where the loss occurred.

This process is classically applied for the performance evaluation of rare events in communication networks [9][10], and the biasing is followed by the unbiasing of results so that the probability of rare events is computed accurately. However, in our case, we do not aim at computing the QoS degradation event, but to learn how to detect it, so that the final unbiasing is not necessary.

Data issued from simulation with IS was used in the training phase of the "Basic-NN" model and regular simulation data was used in the testing phase. The corresponding evaluation is given in line 2 of Table II and presents 0% of false negatives (i.e., detection of 100% of outage situations). However, the global accuracy is reduced to 98% compared to the evaluation without IS which is explained by the increase of the relative false positives (FP) to 2%, i.e., the NN predicts degradation in some outage-free situations.

IV. POSITIONS-BASED PREDICTION

We now move to the case of proactive schemes where the trajectory information, and thus the positions of AGVs, is available to the QoS prediction module. In this case, we do not have detailed SINR values, as we suppose that the future interference is not known. We always use the importance sampling data for the training set.

A. Neural Network Classifier

As a first step, the Basic-NN model was trained by the users' positions to predict the QoS class. Line 3 of Table II shows that the simple architecture of the Basic-NN is no more adapted to the positions as input features, since the global accuracy drops to 83% and the prediction degradation impacts both latency classes (12% FN and 16% FP) which suggests the need for a deeper model to fit with users positions. After hyper parameters tuning, another NN structure is proposed and is composed of 2 hidden layers with 64 neurons. In terms of performance evaluation, line 4 of Table II presents a global accuracy of 97.4% representing a significant enhancement compared to the basic model and the misdetection and false alarm probabilities are respectively 8.3% and 2.5%. No further enhancement was obtained by parameter tuning.

Before moving to a different set of classifiers adapted to the new type of input features, we move back to the SINR-based classifier and make use of the enhanced NN structure for predicting QoS. Line 5 of Table II shows that the prediction results are the same as for the basic-NN model when considering the SINR as input, consolidating the result that Basic-NN is the best option when SINR is available.

B. K-Nearest Neighbors Classifier

Even if the results of the NN classifier with positions (line 4) are enhanced compared to the basic NN structure (line 3), they are still unacceptable for the URLLC use case. We thus investigate the usage of other classifiers and our choice comes naturally to the K-Nearest Neighbors (KNN) classifier. Indeed, the main reason behind exceeding the latency budget is the high level of interference in some specific regions of the industrial environment, and when the positions are known, clustering based on the spatial dimension seems a natural choice. In the classification phase, the latency class of a given user is thus assigned to the most common class among the K nearest neighbors of the user in terms of Euclidian distance. Line 6 of Table II presents the performance of KNN algorithm when K is equal to 1 and generally the results are comparable to the neural networks classifier (line 4). More specifically, we observe a slight degradation of the global accuracy and false alarm probability and an enhancement of the FN. Increasing the parameter K to 3 does not bring any improvement (line 7).

C. M-KNN proposed Classifier

As our aim is to enhance detection of rare events, we propose a generalized version of KNN, called M-KNN, that biases results towards the less represented class in the training set. The first step of M-KNN operates as the traditional KNN where the classes of the K nearest neighbors are determined. We then assign the tested point to the "degraded QoS" class if at least M among these K neighbors belong to this latter class. This second step allows to minimize the probability of misdetection of the rare event, when M is smaller than K/2. This method was applied for K=3 and M=1, meaning that we predict a degraded QoS if at least one of the 3 nearest neighbors in the training set has a degraded QoS. The results of this enhanced clustering method is given by line 8 of Table II where we can notice a large improvement of the QoS degradation prediction (FP=0), with a slight increase of the false alarms (from 3.4% to 6.7%).

V. IMPACT ON THE NETWORK MANAGEMENT

QoS prediction is a first step towards a management framework that anticipates network degradation and reconfigures locally the network or the application for preventing, proactively, the degradation of critical applications. While the mechanisms for network and application reconfiguration are out of the scope of this paper, we describe here the architecture that enables such adaptation and quantify the cost of mis-predictions in terms of number of unnecessary reconfigurations.

A. QoS prediction architecture

Figure 3 illustrates the QoS prediction architecture adopted in this paper. This architecture is inspired from 3GPP rel-16 architecture [8]. We propose to use the Network Data Analytics Function (NWDAF), that is the network function and technical feature of the predictive QoS of the 5G system architecture [8]. NDWAF provides analytics to 5G Core Network Functions and Operation, Administration and Maintenance (OAM) functions. Different procedures are used to support the network data analytics such as the slice and network load analytics. Among the different services provided by the NWDAF, the predictive QoS solution in 3GPP Rel-16 is called QoS Sustainability Analytics where a consumer (a network function) may subscribe to NWDAF analytics service regarding the likelihood of QoS changes for a given period in the future or in the past in a certain area. The outputs depend on the analytics target period, which is specified in the request/ subscription message, and consists of statistics if the target period refers to the past or predictions if the target period refers to the future. In this case, the QoS prediction function that is part of the NWDF receives network and QoS measurements from the dedicated measurement function at the NWDAF, and traffic/trajectory planning from the application controller.

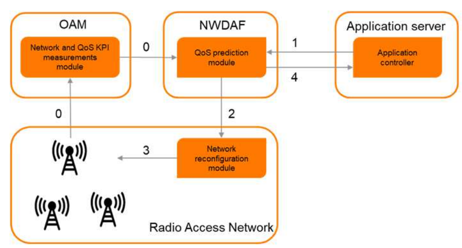

Figure 3: QoS prediction architecture.

B. Example reconfiguration actions

Figure 3 illustrates the steps for the system reconfiguration based on QoS prediction. The OAM module responsible for QoS measurements continuously monitors the network and QoS Key Performance Indicators (KPI) (step 0). The application server sends periodically to the QoS prediction module the trajectory planning and the traffic predictions (step 1). The QoS prediction module combines these two sources of information (steps 0 and 1) and makes use of the AI methods described in this paper to predict QoS. If a degraded QoS is predicted for some devices, the NWDAF sends this information to the network reconfiguration module (step 2), that applies the necessary reconfiguration actions (step 3). The impact of these actions is translated to the NWDAF via a change in the network KPIs. The QoS prediction module predicts again the QoS. If the predicted QoS is still degraded despite the network reconfiguration, it sends a notification to the application controller that needs to adapt the application (step 4).

Note that the QoS changes may be triggered due to various reasons, as for example the handover towards a cell with lower radio signal strength, or the traffic overload. The network reconfiguration procedures may involve additional spectrum reservation, enabling multi-connectivity, interference management, activation of small base stations in high spectrum, etc. When the QoS is predicted to degrade, despite the network reconfiguration, the industrial application has to reconfigure. Note that industrial applications have typically two operation modes:

- Normal mode: where the end to end communication service is delivered according to agreed QoS.
- Safe mode: where the end to end communication service cannot be delivered according to agreed QoS. The safe operation mode is intended for maintenance and troubleshooting to bring application back to Normal mode.

The predictive QoS introduces an additional mode that is "adapted mode", where the End to End communication service may not be delivered according to agreed QoS, but the application temporarily continues with adapted behavior such as reduced speed [4].

C. Quantifying the excessive reconfiguration overhead

While the network and application reconfiguration itself is out of the scope of this paper, we are able to estimate the impact of our proposed QoS prediction method on the initiation of reconfiguration tasks. We apply our prediction framework on the system simulator described in Section II, and observe QoS compared to the predicted QoS when using SINR values (NN parameters of Table II, line 2) or predicted positions (M-KNN parameters of Table II, line 8). We made use of these prediction methods as they ensure 100% detection of the degradation (no False Negatives), with a cost on some false positives (2% and 6.7%, respectively). A false positive may trigger an unnecessary reconfiguration action if it occurs for a user that is attached to a cell which does not present any other user with degraded QoS.

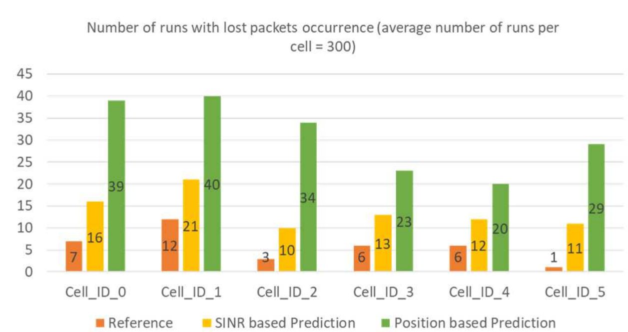

Figure 4: Quantification of the impact of prediction errors on the network reconfiguration planning

Figure 4 illustrates the number of reconfiguration actions that are effectively needed, compared with those based on predictions. There are 300 different simulated positions. We observe that the SINR-based predictions trigger in average 3.7 times more reconfiguration actions, while the position-based prediction applies 9.4 more reconfigurations.

VI. Conclusion

This paper has investigated AI-based QoS prediction for 5G networks in industrial environments. The targeted services are latency-critical and are usually rare in well dimensioned networks. This makes the prediction task hard as the training set is largely imbalanced. We propose an importance sampling technique for re-balancing the training set. We then test prediction methods on two sets of inputs: SINR and device positions. We found out that a Neural Network predictor is adequate for SINR inputs, while a nearest neighbor classifier is better for positions inputs. As the service is critical, we showed how to tune the parameters of the predictors to detect the QoS degradation, with the cost of some false positives. We then propose an architecture for implementing our proposed framework and assessed the impact of the false positives on the system management overhead.

References

[1] Elayoubi, S. E., Jemaa, S. B., Altman, Z., & Galindo-Serrano, A. (2019). 5G RAN slicing for verticals: Enablers and challenges. IEEE Communications Magazine, 57(1), 28-34.

[2] Brown, P., & Elayoubi, S. E. (2020, July). Semi-distributed Contention-based Resource Allocation for Ultra Reliable Low Latency Communications. In IEEE INFOCOM 2020-IEEE Conference on Computer Communications (pp. 1172-1180). IEEE.

[3] Lin, X., Yu, D., & Wiemann, H. (2021). A primer on bandwidth parts in 5g new radio. In 5G and Beyond (pp. 357-370). Springer, Cham.

[4] 5G-ACIA Internal Technical Report "Predictive Data Analytics for Industrial Automation", 16 June 2020

[5] Samba, A., Busnel, Y., Blanc, A., Dooze, P., & Simon, G. (2017, May). Instantaneous throughput prediction in cellular networks: Which information is needed?. In 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM) (pp. 624-627). IEEE

[6] Lin, Y. T., Oliveira, E. M. R., Jemaa, S. B., & Elayoubi, S. E. (2017, May). Machine learning for predicting QoE of video streaming in mobile networks. In 2017 IEEE International Conference on Communications (ICC) (pp. 1-6). IEEE.

[7] Tsilimantos, D., Karagkioules, T., & Valentin, S. (2018, June). Classifying flows and buffer state for YouTube's HTTP adaptive streaming service in mobile networks. In Proceedings of the 9th ACM Multimedia Systems Conference (pp. 138-149).

[8] 3GPP, "TR 38.901 V16.1.0 Study on channel model for frequencies from 0.5 to 100 GHz," December 2019

[9] Smith, P. J.; Shafi, M.; Gao, H. (1997). "Quick simulation: A review of importance sampling techniques in communication systems". IEEE Journal on Selected Areas in Communications. 15 (4): 597–613

[10] S.-E. Elayoubi, B. Fourestié, S. Ben Jemaa, Fast and reliable performance evaluation in mobile wireless networks using dynamic simulations, Computers & Electrical Engineering, Volume 32, Issues 1–3, 2006, https://doi.org/10.1016/j.compeleceng.2006.01.006.

[11] M. Mhedhbi, M. Morcos, A. Galindo-Serrano and S. E. Elayoubi, "Performance Evaluation of 5G Radio Configurations for Industry 4.0," *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2019, pp. 1-6, doi: 10.1109/WiMOB.2019.8923609.

[12] TS 23.288 -Architecture enhancements for 5G System (5GS) to support network data analytics services (Release 16)