2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit) | 978-1-6654-9871-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/EuCNC/6GSummit54941.2022.9815708

Uncertainty Quantification of 5G Positioning as a Location Data Analytics Function

Stefania Bartoletti*, Giacomo Bernini†, Ivan Palamà‡, Michael De Angelis†, Lorenzo Maria Monteforte‡, Takai Eddine Kennouche§, Kostas Tsagkaris¶, Giuseppe Bianchi‡, and Nicola Blefari Melazzi‡
**CNR-IEIIT and CNIT*, Bologna, Italy, email: stefania.bartoletti@ieiit.cnr.it
†*Nextworks*, Pisa, Italy
‡*University of Rome Tor Vergata and CNIT*, Rome, Italy
§*VIAVI Solutions*, Saint-Etienne, France
¶*Incelligent/WINGS*, Athens, Greece

***Abstract*—Mobile positioning is a fundamental service of 5G as it enables a number of applications that rely on location information and location-based analytics. In many applications, it is important to quantify the uncertainty associated with position estimation, for example, for confidence assessment on the location data and anomaly detection, as well as for location data fusion from heterogenous technologies. In this paper, we propose uncertainty quantification as a location data analytics function. First, we introduce an indicator of positioning uncertainty based on the residual measurement error, which does not require the ground truth knowledge. Then, we train and update an uncertainty map of a monitored environment by leveraging the position estimates and location-based measurements collected by multiple users. Such uncertainty map can be used to predict the positioning uncertainty level in any point of a monitored environment. Finally, we propose an implementation of such functionality as an analytics function within the 5G architecture. The functionality is then deployed in a virtualized environment and, using system-level simulations under different propagation conditions, we show how the uncertainty predicted through the proposed method is highly correlated with the true positioning error.**

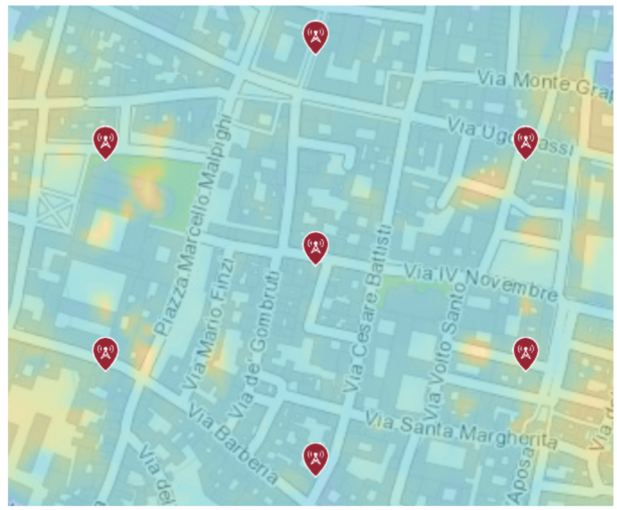

Fig. 1. Example uncertainty map obtained by simulating 5G positioning in a monitored area of Bologna, Italy.

I. Introduction

Mobile positioning is fundamental to a number of applications and services that rely on the location of users served by cellular networks. The 3GPP is indeed enhancing 5G networks and devices with localization capabilities starting from Release 16 [1]. The 5G standard, combined with the integration of heterogeneous technologies, enables accurate location-based analytics that empower a plethora of new 5G services and optimize network utilization. Location-based analytics rely on the positional information from multiple users in one or more monitored environments in order, for instance, to analyze flows or improve network performance through the extraction of complex features and mobility patterns. The deployment of location-based analytics in the 5G architecture calls for a revisit and enhancement of the 5G network functions to interface with location data through a flexible multilayer architecture that facilitates secure sharing of accurate location and context data for localization, and that combines information from different network functions for location-based analytics [2].

According to the 3GPP standardization, positioning in 5G is targeting sub-meter accuracy or better in nearly 95% of the time, thanks to the capability to operate in both sub-6 GHz and millimeter wave (mmWave) frequency bands and the use of massive antenna arrays [3]–[6]. This makes 5G a key enabler for many location-based services and for the extraction of analytics that require very high levels of accuracy, see, e.g., safety-critical vehicular applications. However, cluttered environments, such as urban ones, can yet negatively impact positioning accuracy due to obstructions caused by non-line-of-sight and multipath propagation [6].

Assessing the degree of uncertainty associated with the position estimate is important in several scenarios. For example, a location-based decision can be based on the level of uncertainty expected in the particular location. Another example is when one particular positioning service fails at delivering a required accuracy level; in such a case, more resources could be deployed to enhance its performance, or an alternative positioning service can be triggered if available. The ability to quantify the expected uncertainty associated with a particular deployment of resources is key for implementing complex interaction with other 5G functions, where the best possible level of uncertainty as a function of resources, timing, etc, must be found. In other cases, if the uncertainty in a considered position is very high in comparison to what is expected, then this can be an indication that there may be a spoofing or meaconing attack [7], [8]. Recently, within the definition of positioning and timing service of ETSI for vehicular scenarios, a lot of effort has been dedicated to the detection of failure occurring when the position and time entity are unable to estimate location with an error less than the maximum tolerable threshold. In addition, an important KPI is defined as the positioning integrity, which measures the trust that can be placed in the correctness and confidence of the estimated position [9].

The knowledge of the true positioning error would require to know the true position of the UE, which is unknown in practical applications. Therefore, the quantification of uncertainty in the absence of the ground truth is an important research topic for location-based services [10], [11]. For example, in [11], the authors proposed a maximum likelihood (ML)-based method for uncertainty prediction. The model is trained with a dataset that includes the ground truth position and leads to a predicted uncertainty that is highly correlated with the actual positioning error. In [10] the authors proposed a comprehensive analysis of estimators for the positioning error, which do not require the ground truth knowledge but rely on multiple estimates from each gNB; such estimators cannot be applied to the case where a single estimate is available.

The key contributions of this paper are: (i) we introduce the positioning uncertainty as an analytics function by exploiting the location information of multiple users; (ii) we propose an example of uncertainty quantification based on the residual error in 5G positioning which allows to build an uncertainty map; (iii) we propose ML-based regression for the prediction of uncertainty and use a goodness-of-fit test for the online updating of the model; (iii) we define a possible deployment of the uncertainty quantification within the 5G architecture as a Location Data Analytics Function (LDAF); (iv) we implement and test the proposed algorithms in a virtualized environment. Numerical results are obtained through system-level simulations that take into account the accuracy of 5G positioning according to the 3GPP standard. Then, the quantification of uncertainty is proposed through a ML-based regression model trained on a dataset of position estimates.

II. 5G Positioning

A. System Model

The UE position in 5G is estimated based on location-dependent measurements (e.g., time of arrivals or angles) performed between the UE to be located and one or multiple gNBs [1], [6]. Specifically, two signals have been defined for the purpose of UE positioning, namely downlink positioning reference signal (DL-PRS) and uplink sounding reference signal (UL-SRS). Nevertheless, it is possible to take advantage of other reference signals for positioning. Time-difference-of-arrival (TDOA), angle-of-arrival (AOA), and angle-of-departure (AOD) measurements are taken at a single or multiple reception points. The TDOA is measured with respect to a reference base station. The angle measurements are obtained by measuring the received signal power from different beams pointing in distinct directions.

We consider a network of N_g gNode-Bs (gNBs) that are operating in a monitored environment. The ith gNBs is at $\mathbf{p}_\text{gNB}^{(i)}$, while the true unknown position of a generic user is at $\mathbf{p}_\text{UE}$. A location-dependent measurement is performed for each gNBs, and corresponds to $m^{(i)}(\mathbf{p}_\text{UE})$. For example, $m^{(i)}(\mathbf{p}_\text{UE})$ can be a time measurement from DL-PRS between the ith gNBs and the user equipment (UE). The measurement model for $m^{(i)}(\mathbf{p}_\text{UE})$ can be described as

$$m^{(i)}(\mathbf{p}) = \check{m}^{(i)}(\mathbf{p}) + n^{(i)}(\mathbf{p}) \tag{1}$$

where $\check{m}^{(i)}(\mathbf{p})$ is a deterministic and known function and $n^{(i)}(\mathbf{p})$ is the measurement noise. In the case of DL-TDOA, $\check{m}^{(i)}(\mathbf{p}) = \|\mathbf{p}_\text{gNB}^{(i)} - \mathbf{p}_\text{UE}\|$, and the TDOA is calculated as $\check{m}^{(i)}(\mathbf{p}) - \check{m}^{(r)}(\mathbf{p})$, where the rth gNBs is considered as reference gNBs. The measurement noise $n^{(i)}(\mathbf{p})$ is generally dependent on the position, on the line-of-sight (LOS) conditions between the transmitter and receiver, and the signal-to-noise ratio (SNR) of the received signal.

Then, from the vector of measurements from the different gNBs, i.e. $\mathbf{m}(\mathbf{p}_\text{UE}) = [m^{(1)}(\mathbf{p}_\text{UE})\, m^{(2)}(\mathbf{p}_\text{UE}) \dots m^{(N_\text{g})}(\mathbf{p}_\text{UE})]$, we obtain an estimate of the UE position as $\hat{\mathbf{p}}_\text{UE}$. As an example, in the DL-TDOA based positioning, the $\hat{\mathbf{p}}_\text{UE}$ is obtained as the point that intersects the hyperbolas defined as the set of points whose distances from the reference gNB and the ith gNB is equal to $(\check{m}^{(i)}(\mathbf{p}) - \check{m}^{(r)}(\mathbf{p}))\, c$, where c is the speed-of-light. Then, the positioning error is defined as the Euclidean distance between the true and estimated UE position, i.e. $e(\mathbf{p}_\text{UE}, \hat{\mathbf{p}}_\text{UE}) = \|\mathbf{p}_\text{UE} - \hat{\mathbf{p}}_\text{UE}\|$.

B. Architectural Aspects

5G consists of the next-generation radio access network and the 5G core network. The enhanced 3GPP Location Service architecture defines the location-related functionalities for any UE. An UE in 5G can either be positioned by itself or by the network. Location services are initiated by the access and mobility management function (AMF), either on behalf of a particular UE or by a location services client, which can be any network element that interacts with the gateway mobile location center (GMLC) to access and process location data. Clients can be anywhere in the architecture, even within the UE. The location service request is then communicated to the location management function (LMF), i.e., the location server, which coordinates and calculates the UE position. The positioning assistance information and measurements are transferred between the UE and the TRP to and from the LMF.

III. Uncertainty Quantification

A. Residual Error for Uncertainty Assessment

Following an inverse problem approach, we propose to quantify the uncertainty considering the measurement model in (1) and following the data processing scheme illustrated in Fig. 2. Given the estimated position $\hat{\mathbf{p}}_\text{UE}$ and the measurement vector $\mathbf{m}(\mathbf{p}_\text{UE})$, for the ith gNB, we can consider the discrepancy function

$$m^{(i)}(\mathbf{p}_\text{UE}) - \check{m}^{(i)}(\hat{\mathbf{p}}_\text{UE}) \tag{2}$$

which is the difference between the ideal measurement that is expected at $\hat{\mathbf{p}}_\text{UE}$ and the true measurement $m^{(i)}(\mathbf{p}_\text{UE})$ and represents the residual error for the measurement model

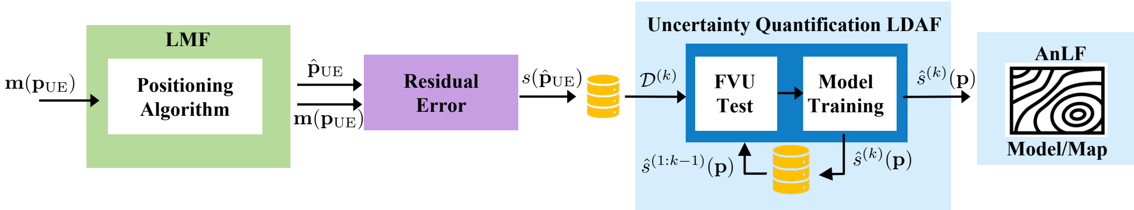

Fig. 2. Illustration of the data processing for uncertainty quantification.

(1) given the estimated position. Then, the standard error is considered as the uncertainty indicator and calculated as[1]

$$s(\hat{\mathbf{p}}_{\text{UE}}) = \left[\frac{1}{N_{\text{g}}-2}\sum_{i=1}^{N_{\text{g}}}(m^{(i)}(\mathbf{p}_{\text{UE}}) - \check{m}^{(i)}(\hat{\mathbf{p}}_{\text{UE}}))^2\right]^{1/2}. \quad (3)$$

In time-based positioning, a geometrical interpretation of such an indicator can be provided using the *law of cosines*. In such a case, the positioning error can be expressed as a function of the estimated and true position from the point of view of each gNB [10] as

$$\begin{aligned} e^2(\mathbf{p}_{\text{UE}}, \hat{\mathbf{p}}_{\text{UE}}) =& \left(\check{m}^{(i)}(\mathbf{p}_{\text{UE}})c\right)^2 + \left(\check{m}^{(i)}(\hat{\mathbf{p}}_{\text{UE}})c\right)^2 \\ & - 2\check{m}^{(i)}(\mathbf{p}_{\text{UE}})\check{m}^{(i)}(\hat{\mathbf{p}}_{\text{UE}})c^2\cos(\phi_i(\hat{\mathbf{p}}_{\text{UE}}, \mathbf{p}_{\text{UE}})) \end{aligned} \quad (4)$$

where $\phi_i(\hat{\mathbf{p}}_{\text{UE}}, \mathbf{p}_{\text{UE}})$ is the angle formed between the true position $\hat{\mathbf{p}}_{\text{UE}}$, the gNB position $\mathbf{p}_{\text{gNB}}^{(i)}$, and the estimated position $\hat{\mathbf{p}}_{\text{UE}}$. As $\check{m}^{(i)}(\mathbf{p}_{\text{UE}})$ is unknown, each squared residual $(m^{(i)}(\mathbf{p}_{\text{UE}}) - \check{m}^{(i)}(\hat{\mathbf{p}}_{\text{UE}}))^2$ in (3) can be considered as an approximation of $e^2(\mathbf{p}_{\text{UE}}, \hat{\mathbf{p}}_{\text{UE}})$ when $\phi_i(\hat{\mathbf{p}}_{\text{UE}}, \mathbf{p}_{\text{UE}}) \simeq 0$ (i.e., we are assuming that the true position is on the same line that intersects $\hat{\mathbf{p}}_{\text{UE}}$ and $\mathbf{p}_{\text{gNB}}^{(i)}$) and $\check{m}^{(i)}(\mathbf{p}_{\text{UE}}) \simeq m^{(i)}(\mathbf{p}_{\text{UE}})$ (i.e., we have a small measurement error). Then, we average among the measurements available from each gNB, i.e. varying i to obtain an estimate of $e(\mathbf{p}_{\text{UE}}, \hat{\mathbf{p}}_{\text{UE}})$ through (3).

B. Uncertainty as an Analytic Function

If a dataset of estimated positions and associated standard errors from several UEs is collected in a monitored area, the residual standard error $s(\hat{\mathbf{p}})$ in (3) can be learned as a function of $\hat{\mathbf{p}}$ by applying a regression analysis. This can allow us to predict the uncertainty in any possible estimated position. To this aim, any ML algorithm for regression can be used, e.g. random forest, or Gaussian process. As the standard error changes with the measurement noise model, i.e. under different propagation conditions, the learned function $s(\hat{\mathbf{p}})$ could be updated online to reflect the new measurement model.

We start with a dataset $\mathcal{D}^{(0)} = \{\hat{\mathbf{p}}_{\text{UE}}^{(i)}, s(\hat{\mathbf{p}}_{\text{UE}}^{(i)})\}_{i=1}^{N^{(0)}}$ of position estimates from $N^{(0)}$ UEs. Then, we generate a regression model based on such a dataset and we obtain $\hat{s}(\mathbf{p})$ from which we can construct a map by predicting the uncertainty values over different positions. When a new dataset $\mathcal{D}^{(k)} = \{\hat{\mathbf{p}}_{\text{UE}}^{(i)}, s(\hat{\mathbf{p}}_{\text{UE}}^{(i)})\}_{i=1}^{N^{(k)}}$ at time index k is obtained, we can calculate the fraction of variance unexplained (FVU) of the latest model or of a set of available models. The FVU is the fraction of variance of the observed data that can be explained by a regression model and is calculated as

$$F_{\text{vu}} = \frac{\sum_{i=1}^{N^{(k)}}\left(s(\hat{\mathbf{p}}_{\text{UE}}^{(i)}) - \hat{s}(\hat{\mathbf{p}}_{\text{UE}}^{(i)})\right)^2}{\sum_{i=1}^{N^{(k)}}\left(s(\hat{\mathbf{p}}_{\text{UE}}^{(i)}) - \overline{s}^{(k)}\right)^2} \quad (5)$$

where $\overline{s}^{(k)} = \frac{1}{N^{(k)}}\sum_{i=1}^{N^{(k)}} s(\hat{\mathbf{p}}_{\text{UE}}^{(i)})$. Specifically, small values for F_{vu} indicate that the regression model considered is able to explain most of the variance of the observed data. Then, a threshold ξ_{fvu} can be set such that if $F_{\text{vu}} \geq \xi_{\text{fvu}}$, a new model is generated. Alternatively, if any model is available for which $F_{\text{vu}} < \xi_{\text{fvu}}$, the model with lowest value of F_{vu} is considered.

[1]The denominator $N_{\text{g}} - 2$ represents the degrees of fredom, which are equal to the number of measurements minus the number of parameters to be estimated.

IV. Location Data Analytics

3GPP has introduced a general framework for data analytics in 5G infrastructures starting from the Releases 15 and 16. In particular the 3GPP TR 23.791 [12] defined a dedicated function within the 5G core, called Network Data Analytics Function (NWDAF) for the collection of data from the other network functions (and external data sources) and the delivery of data analytics services.

The NWDAF can apply different levels of analytics granularity (e.g. global, per network slice), and can be deployed as a centralised core-based instance alongside distributed edge instances. It collects metrics or data analytics information locally elaborated from heterogeneous sources, such as 5G network functions, application functions, the management system as well as from external data repositories. All this data is processed by means of aggregation mechanisms, prediction algorithms, etc., with the aim of generating further analytics data to be then offered to other network functions or if needed stored in dedicated data repositories. Starting from Release 16, the 3GPP TS 23.288 specification [13] is standardizing the NWDAF interfaces and the procedures enabling the consumption of data analytics services.

A. LDAF architecture and functional split

The use cases currently identified by 3GPP for the NWDAF, as well as the related services (data types, semantics), do not cover the UE positioning information for further contextualized, location-enriched analytics that would generate an added value from the raw location information, including enabling the realization of new location-based services. We propose to fill this gap in contextualized and location-enriched analytics, by trying to enrich the basic NWDAF functionalities with specific localization information and analytics.

In practice, we model functions that mainly provide data analytics on top of UE positioning data (like the uncertainty quantification described above) as Localization Data Analytics Functions (LDAFs), to cover the part of data analytics exclusively associated with localization-awareness within the 5G core architecture [2]. Therefore, the LDAF considers as main input UE positioning information produced by the LMF, but other related data can also be pulled if available and necessary. For location based analytics, such external data could include video surveillance streams, mapping information, and telemetry data from urban infrastructures and the like, that can enhance the positioning accuracy and analytics output by correlating and anchoring the network data with the physical environment. From a 5G core architecture point of view, we define the LDAF as a specialized NWDAF, thus following the service-based paradigm and its standard interfaces and procedures (based on pub/sub and HTTP-based mechanisms). However, the LDAF produces specific new analytics information, and can then be associated to new analytics identifiers and data structures to be added as part of the input/output data of analytics services described in [13].

Beyond the alignment with the NWDAF basic functionalities, and since the adoption of ML techniques is becoming more and more relevant in the context of 5G networks, the proposed LDAF is split into two logical functions, to clearly separate the pure analytics logic from the ML model training and management. This follows the evolved NWDAF approach proposed in [13], and allows to separate ML model and pipeline management functionalities (e.g., for training and creating new versions of the same model) from the actual analytics functions (i.e., pre-trained) ready to be deployed and operated for specific analytics purposes (e.g., the uncertainty quantification). As shown in Fig. 3, and inline with services defined in [13], the LDAF Model Training Logical Function (MTLF) takes care of ML models training, exposing dedicated services and APIs for external functions to discover and query available trained models (ML Model Info) and to provision them (ML Model Provision). On the other hand, the LDAF Analytics Logical Function (AnLF) performs inference, elaborates statistics or predictions and exposes dedicated services for analytics information retrieval (Analytics Info), e.g., in the form of specific location analytics data structures and identifiers. The main source of input of the LDAF MTLF within the 5G core architecture is data coming from he LMF, which can provide UE positioning information, as well as additional data collected from other network functions and external sources, following the data delivery and storage mechanisms defined in 3GPP TS 23.288 [13] and implemented by the combination of Data Collection Coordination Function (DCCF) and Analytics Data Repository Function (ADRF).

B. Implementation of Uncertainty Quantification as LDAF

The 5G network architecture brings a new disruptive approach that considers a high-degree of virtualization of the network functions and services by design. In particular, the 5G core architecture and its network functions are defined to be natively deployed and operated on top of virtualized infrastructures. In line with this design principle, the proposed LDAF implements the uncertainty quantification functionality described in Section III and is realized as a set of virtual

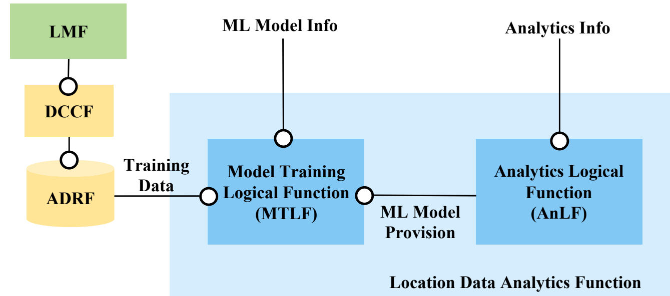

Fig. 3. LDAF functional decomposition

functions that allow to deploy AnLF and MTLF as containerized services in a Kubernetes[2] [14] based environment. Fig. 4 shows how the uncertainty quantification LDAF has been implemented.

First, the LDAF MTLF makes use of Kubeflow[3], a tool for making deployments of ML workflows on Kubernetes simple, portable and scalable. In particular, Kubeflow Pipelines allow to model machine learning workflows, including all of the components in the workflow and how they combine in the form of a graph. A pipeline component in Kubeflow is an implementation of a pipeline task, a self-contained set of user code, representing a step in the workflow. Each component is packaged as a Docker [14] container that performs a single step in the pipeline (e.g., data pre-processing, data transformation, model training and so on).[4] In the uncertainty quantification LDAF, a Kubeflow Pipeline has been implemented for ML model training, and creates new versions of the model when new relevant training data is available in the ADRF. For this reason, the LDAF MTLF also includes a ML model registry service (implemented through MinIO[5]), where all available versions of the ML model are stored. When the Kubeflow Pipeline is executed, the various components are automatically executed as a virtualized service in a Kubernetes POD to perform the model training, and as final steps it deploys the generated model in the Kubernetes cluster to make it available as a running AnLF (in a dedicated Kubernetes namespace). The new generated ML model is also stored in the registry service. In addition, the LDAF MTLF is equipped with a ML model evaluation functionality, which computes fraction of variance unexplained (FVU) scores for all the available models in the registry service whenever a new training dataset is available. This allows to perform automated lifecycle management of the LDAF AnLF, assuring that the best performing uncertainty quantification ML model is always used in the AnLF.

The LDAF AnLF is represented by the ML model that the Kubeflow pipeline service automatically deploys as result of its execution. This is achieved by the integration of Kubeflow with Seldon[6], a tool that allows to package the AnLFs as Docker containers and be executed and exposed as REST microservices in Kubernetes to retrieve on-demand the uncertainty quantification analytics results through the Analytics Info REST interface.

[2]https://kubernetes.io/
[3]https://kubeflow.org/
[4]https://docker.com/
[5]https://min.io/
[6]https://www.seldon.io/

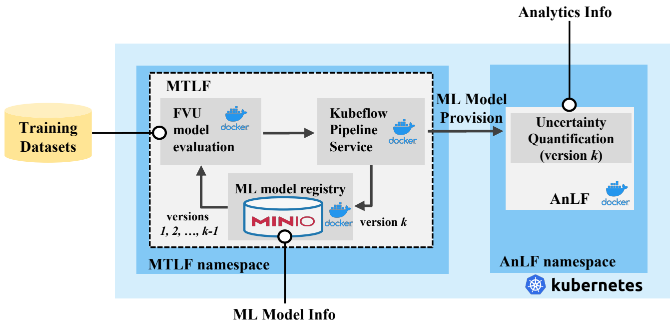

Fig. 4. LDAF implementation

V. Case Study

A. Simulation Settings

As simulated scenario we considered a portion of the Bologna city, in Italy. In particular, a monitored squared area of $500\,\mathrm{m}\times 500\,\mathrm{m}$ is considered, with seven gNBs located as illustrated in Fig. 1, with an inter-site distance of 200m, following the example of the 3GPP TR 38.901 [15]. The collection of each dataset has been performed by considering 2000 uniformly distributed positions for the UE in the monitored area. 5G New Radio positioning is simulated based on the DL-TDOA algorithm in the 5G Toolbox Matlab environment.

LOS/NLOS conditions between each gNB and UE have been calculated using the Openstreetmap software. Given the LOS/NLOS condition, the corresponding tapped delay line proposed in 3GPP has been considered for simulating channel propagation (TDL-E for LOS and TDL-A for NLOS) [15]. Such channel models are parameterized with respect to the delay spread. We have used such parameter to simulate a change of channel conditions, and in particular, we have considered four different measurement models that correspond to four different values of delay spread, i.e. 10ns (Model 1), 30ns (Model 2), 100ns (Model 3), and 300ns (Model 4). Based on the estimated position and the TOA measurements at each gNB, the residual error is calculated. The regression algorithm used to build the uncertainty map is the Extra Trees Classifier. Such algorithm was chosen because it has comparable performance to the random forest regression model, which has been shown to be suitable in the context of position estimation in [11], while requiring less time resources.

The LDAF MTLF, implemented as reported in Figure 4, has been deployed in a single node Kubernetes cluster environment realized through a Microk8s[7] (v1.21.9) available on a Virtual Machine (16 vCPU, 32GB RAM, 100GB HDD) instantiated on an OpenStack[8] cloud computing infrastructure. Kubeflow was deployed in the Kubernetes cluster using the official Microk8s add-on while MinIO was instantiated using the official Helm chart release. As last steps, for the availability of the components deployed on the cluster, Kubeflow Pipeline and MinIO have been externally exposed using a NodePort Kubernetes service type.

The model evaluation is performed by calculating the FVU every time a new dataset of 1000 users is collected. The AnLF model is then automatically deployed and updated in a dedicated Kubernetes namespace only when the FVU overcomes the decided threshold $\xi_{\mathrm{FVU}} = 0.5$ which means that model explains most of the variability in the measured residual errors. To appreciate the importance of model updating, we simulate the use of different measurement models for generating the training and testing datasets. Then, the results obtained when the model is updated are compared to those obtained without updating the model.

As performance metrics, we use: (i) the Pearson correlation coefficient between the true positioning error and the uncertainty level; (ii) the FVU in (6); and (iii) the root mean square of the prediction relative error, i.e. $\hat{s}(\mathbf{p}_{\mathrm{UE}}) - s(\mathbf{p}_{\mathrm{UE}})$.

B. Simulation Results

Figure 5 shows the correlation coefficient between the true positioning error $e(\mathbf{p}_{\mathrm{UE}}, \hat{\mathbf{p}}_{\mathrm{UE}})$ and the measured uncertainty level $s(\mathbf{p}_{\mathrm{UE}})$, as well as between the true positioning error and the predicted uncertainty level $\hat{s}(\mathbf{p}_{\mathrm{UE}})$. Results are obtained varying the measurement model for generating the dataset used for training; the same measurement model is used for generating the dataset used for testing. Note that values above 0.7 represent high correlation, while values between 0.5 and 0.7 are considered medium correlation. The p-value for all the simulation was below 10^{-5}. The figure shows that the proposed uncertainty indicator is highly correlated with the true positioning error for all the models considered. Also, the predicted value is highly correlated with the true error for models between 1 and 3, and moderately correlated for a delay spread of 300 ns (i.e., Model 4).

Figure 6 shows the FVU varying the measurement model using for generating both the training and testing datasets. Results show that the prediction obtained with Models 1-3 have low FVU even when the model is changed among the same group of models, i.e. the residual statistics are comparable. Differently, if we test the regression model trained with a dataset generated from Model 4 with dataset generated with any other model, we have an FVU much higher. Similarly, the regression model trained with datasets from Models 1-3 is not suitable to predict the uncertainty when the measurement model is Model 4, i.e. the FVU is higher. Therefore, it is important to update the regression model when the channel propagation conditions change. Note that when testing a dataset from Model 4, an FVU larger than 0.5 is experienced even when the trained dataset is generated with the same model. This means that as the delay spread gets larger, there is more variability that cannot be explained by the prediction model.

Figure 7 shows the root mean square of the prediction relative error, which measures the goodness of the regression model. In the example, a first regression model is trained with a dataset obtained with Model 4 (i.e., delay spread of 300 ns). Then, it is tested with two datasets generated with different measurement models. After the second dataset is collected, the first model is evaluated through the FVU test and might be updated. When the model is updated, the relative error is reduced with all the models. With Model 4, the reduction is smaller as this is the same model used for generating the training dataset. Furthemore, in agreement with the results shown in Fig. 6 and Fig. 5, the error obtained with Model 4 is higher than for the other models.

[7]https://microk8s.io/
[8]https://openstack.org/

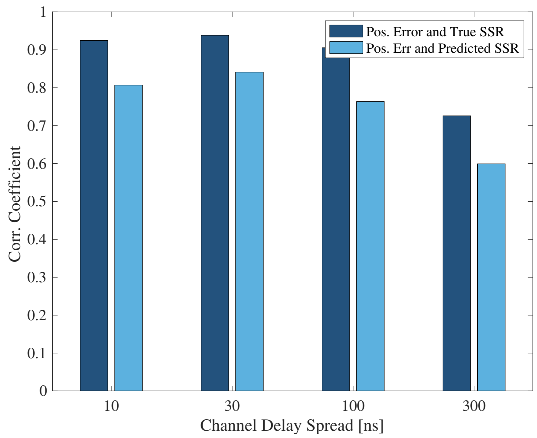

Fig. 5. The correlation coefficient between the true error and the measured residual error is compared with the one between the true error and the predicted residual error varying the data measurement models, i.e., varying the channel delay spread as 10 ns (Model 1), 30 ns (Model 2), 100 ns (Model 2), and 300 ns (Model 4).

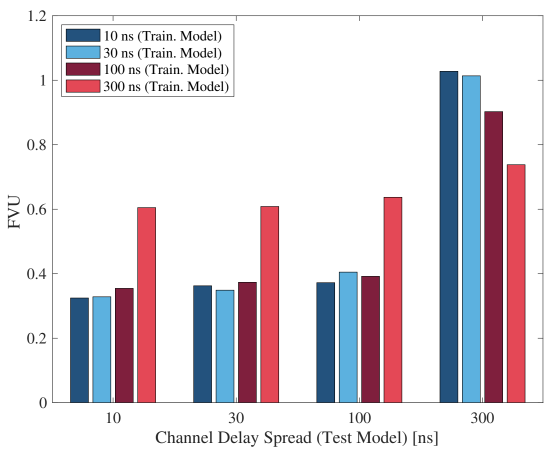

Fig. 6. Fraction of Variance Unexplained (FVU) obtained by varying the measurement model of both the training set for the regression model and test set for its validation.

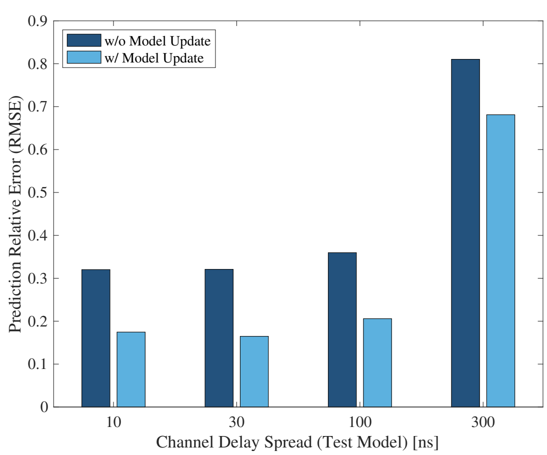

Fig. 7. Root mean square of the relative error between the predicted and true SSR value. Darker bars are obtained when the model is not updated. Lighter bars are obtained when the model is updated based on the FVU.

VI. Final Remark

In this paper we proposed the positioning uncertainty quantification as an analytics function in the context of 5G positioning. We have proposed a simple yet effective indicator for the positioning uncertainty level and a ML-based approach for predicting and updating the uncertainty level and build an uncertainty map in a monitored environment. We have implemented the proposed approach in a virtualized environment as a location data analytics function that is compliant with the 5G architecture. Results show that the proposed indicator is highly correlated with the true positioning error.

Acknowledgements

This work was supported by the European Union's Horizon 2020 research and innovation programme under Grant no. 871249.

References

[1] *Study on NR positioning support*, 3rd Generation Partnership Project 3GPP™ ETSI TR 138 955 V16.0.0, Dec. 2019, release 16.

[2] S. Bartoletti, L. Chiaraviglio, S. Fortes, T. E. Kennouche, G. Solmaz, G. Bernini, D. Giustiniano, J. Widmer, R. Barco, G. Siracusano, A. Conti, and N. Blefari-Melazzi, "Location-based analytics in 5g and beyond," *IEEE Communications Magazine*, vol. 59, no. 7, pp. 38–43, 2021.

[3] 3GPP, "5G System (5GS) Location Services (LCS); Stage 2," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 23.273, 7 2020, version 16.4.0.

[4] ——, "Universal Terrestrial Radio Access (UTRA) and Evolved UTRA (E-UTRA); User Equipment (UE) performance requirements for Radio Access Technology (RAT) Independent Positioning Enhancements," 3GPP, Technical Specification (TS) 37.171, 10 2020, version 16.1.0.

[5] H. Sarieddeen, N. Saeed, T. Y. Al-Naffouri, and M. Alouini, "Next generation Terahertz communications: A rendezvous of sensing, imaging, and localization," *IEEE Communications Magazine*, vol. 58, no. 5, pp. 69–75, 2020.

[6] S. Dwivedi, R. Shreevastav, F. Munier, J. Nygren, I. Siomina, Y. Lyazidi, D. Shrestha, G. Lindmark, P. Ernström, E. Stare, S. M. Razavi, S. Muruganathan, G. Masini, Busin, and F. Gunnarsson, "Positioning in 5G networks," *IEEE Communications Magazine*, vol. 59, no. 11, pp. 38–44, 2021.

[7] R. Morales-Ferre, P. Richter, E. Falletti, A. de la Fuente, and E. S. Lohan, "A survey on coping with intentional interference in satellite navigation for manned and unmanned aircraft," *IEEE Communications Surveys Tutorials*, vol. 22, no. 1, pp. 249–291, 2020.

[8] S. Bartoletti, G. Bianchi, D. Orlando, I. Palamà, and N. Blefari-Melazzi, "Location security under reference signals' spoofing attacks: Threat model and bounds," in *The 16th International Conference on Availability, Reliability and Security*, ser. ARES 2021, 2021.

[9] "Intelligent transport systems (ITS); facilities layer function; Part 2: Position and time management (PoTi); release 2," *ETSI EN 302 890-2*, Mar. 2020.

[10] J. Nounagnon and T. Pratt, "Improving position MSE estimation for biased estimators without true position knowledge," in *2016 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2016, pp. 100–105.

[11] Y. Zhao and D. Shrestha, "Uncertainty in position estimation using machine learning," in *2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2021, pp. 1–7.

[12] *Study of enablers for Network Automation for 5G*, 3rd Generation Partnership Project 3GPP™ ETSI TR 23 791 V16.2.0, Jun. 2019, release 16.

[13] 3GPP, "Architecture enhancements for 5G System (5GS) to support network data analytics services," 3GPP, Technical Specification (TS) 23.288, 11 2021, version 17.3.0.

[14] D. Bernstein, "Containers and cloud: From LXC to docker to kubernetes," *IEEE Cloud Computing*, vol. 1, no. 3, pp. 81–84, 2014.

[15] *Study on channel model for frequencies from 0.5 to 100 GHz*, 3rd Generation Partnership Project 3GPP™ ETSI TR 138 900 V16.1.0 (2019-12), Dec. 2019, release 16.