Empowering 6G Positioning and Tracking with Bayesian Neural Networks

Bernardo Camajori Tedeschini*†, Girim Kwon†, Monica Nicoli‡, Moe Z. Win†
*Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano, 20133 Milan, Italy
†Wireless Information and Network Sciences Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139 USA
‡Dipartimento di Ingegneria Gestionale (DIG), Politecnico di Milano, 20156 Milan, Italy
Email: bernardo.camajori@polimi.it, girimk@mit.edu, monica.nicoli@polimi.it, moewin@mit.edu

***Abstract*—In the rapidly evolving domain of forthcoming 6th generation (6G) networks, achieving precise dynamic positioning down to the centimeter becomes critical, particularly in complex urban scenarios as those envisioned for cooperative intelligent transport systems (C-ITSs). To face the challenges introduced by severe path loss and blockages in new 6G frequency bands, machine learning (ML) provides innovative strategies to extract locational intelligence from wide-band space-time radio signals. This paper proposes the integration of Bayesian neural networks (BNNs) into cellular multi-base station (BS) tracking systems, where uncertainties of BNNs account for finite training sets and measurement errors. Our approach utilizes a deep learning (DL)-based autoencoder (AE) structure that exploits the full channel impulse response (CIR) to infer location-centric attributes in both line-of-sight (LoS) and non-LoS (NLoS) conditions. Validations in a 3rd Generation Partnership Project (3GPP) compliant urban micro (UMi) setting, simulated with ray-tracing and traffic simulations, demonstrate the superior performances of BNN-based tracking with respect to both traditional geometric-based tracking methods and state-of-the-art DL models.**

***Index Terms*—Bayesian neural networks, cooperative tracking, positioning, channel impulse response, 6G.**

I. Introduction

The forthcoming 5th generation (5G) Advanced in 3rd generation partnership project (3GPP) Release 18 promises a significant leap in cellular positioning accuracy, aiming for centimeter-level precision through features such as massive multiple-input multiple-output (mMIMO), increased bandwidths, and millimeter waves (mmWave) technologies [1]–[3]. Challenges include higher path losses and frequent blockages, which limit conventional positioning solutions. 5G Advanced addresses these challenges by integrating machine learning (ML) into enhanced location services [4], [5]. In particular, exploiting deep learning (DL) techniques, such as autoencoder (AE) structures, permits to perform direct position estimation by matching channel impulse response (CIR) location fingerprints with pre-stored training data [6], [7]. This is critical in non-LoS (NLoS) conditions where reflections cause non-negligible measurement errors in geometric-based algorithms.

Despite ML's potential, a main limit is the lack of uncertainty quantification, especially for critical applications such as connected automated vehicles (CAV) positioning [8]. Bayesian neural networks (BNNs) offer a solution by not only providing point estimates but also quantifying the associated uncertainty, leading to more reliable and robust positioning [9]. The uncertainty derives from the limited spatial density of training data (*epistemic uncertainty*) and the measurement noise (*aleatoric uncertainty*). BNNs can be particularly resilient in static positioning, as they incorporate prior knowledge and are able to quantify the different types of uncertainty.

BNNs have been recently used for uncertainty estimation in static 5G localization [10], [11]. However, these works do not exploit the predicted uncertainty for refining the user equipment (UE) position. Regarding 5G mobile positioning, the majority of works rely on traditional Bayesian methods, such as extended Kalman filter (EKF) or message passing algorithm (MPA) [12], in conjunction with mmWave and MIMO enablers. In the field of DL, recurrent neural networks (RNNs) have been gathering much attention thanks to their ability to learn temporal dependencies [13]. Moreover, a recent study [14] investigated the usage of attention mechanism in temporal convolutional networks (TCNs) for NLoS outdoor tracking, achieving a state-of-the-art mean absolute error (MAE) of 1.8 m. Nevertheless, RNNs and TCNs face limitations in mobile positioning due to the need for highly accurate training data (i.e., ground truth for dynamic trajectories) and their inability to quantify prediction uncertainty, hindering their use in safety-critical contexts.

In this paper, we propose a novel BNN-based 6th generation (6G) tracking procedure that relies on an offline training phase and subsequent integration of BNN uncertainties in a Bayesian tracking scheme. Inspired by 3GPP's vision on future mobile systems and the superior results obtained in NLoS identification [7] and positioning [6], we integrate the BNN methodology into an AE structure that exploits the full CIR for positioning, i.e., the 2D angle-delay channel power matrix (ADCPM). We test the proposed methodology in a realistic cooperative intelligent transport system (C-ITS) scenario within an urban landscape. Our simulated network adheres to 5G standard-compliant sce-

The fundamental research described in this paper was supported in part by the Roberto Rocca Doctoral Fellowship granted by the Massachusetts Institute of Technology and Politecnico di Milano, in part by the project Centro Nazionale per la Mobilità Sostenibile (MOST), funded by the Italian Ministry of University and Research under the PNRR funding program, in part by the National Research Foundation of Korea under Grant 2021R1A6A3A14040142, in part by the National Science Foundation under Grant CNS-2148251, and in part by the Federal Agency and Industry Partners in the RINGS Program.

978-1-7281-9054-9/24/$31.00 ©2024 IEEE

narios [15] and provides realistic outdoor conditions through the use of accurate 3D ray tracing, along with microscopic vehicular traffic modeling for simulation of UE pathways [16].

The article is structured as follows. Sec. II outlines the system architecture and the channel fingerprint. Sec. III introduces BNNs terminology and our proposed DL model. Sec. IV presents the integration of BNNs into the cellular positioning system. Sec. V details the simulation setup and findings, while Sec. VI draws the conclusions.

Notations: A random variable and its realization are denoted by x and x; a random vector and its realization are denoted by $\mathbf{x}$ and $\boldsymbol{x}$; a random matrix and its realization are denoted by $\mathbf{X}$ and $\boldsymbol{X}$, respectively. The function $p_{\mathsf{x}}(x)$, and simply $p(x)$ when there is no ambiguity, denotes the probability density function (PDF) of x. With the notation $\mathsf{x} \sim \mathcal{N}(\mu, \sigma^2)$ we indicate a Gaussian random variable x with mean μ and standard deviation σ, whose PDF is denoted by $\mathcal{N}(x; \mu, \sigma^2)$. With the notation $\mathsf{x} \sim \mathcal{U}(a, b)$ we indicate a Uniform random variable x with support $[a, b]$. We use $\mathbb{E}\{\cdot\}$ and $\mathbb{V}\{\cdot\}$ to denote the expectation and the variance of random variable, respectively. $\mathbb{R}$ and $\mathbb{C}$ stand for the set of real and complex numbers, respectively.

II. 6G Positioning System

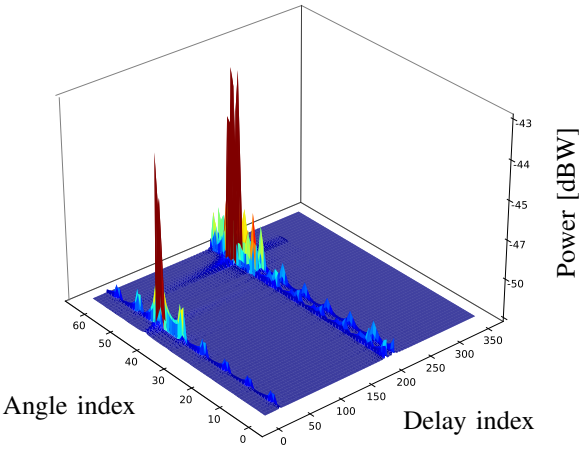
A. Space-Time Channel Model

We consider a mmWave orthogonal frequency division multiplexing (OFDM) system in which an uplink communication is established between a UE and a number of base stations (BSs) at a specific carrier wavelength λ_{c}. Each BS is equipped with a uniform planar array (UPA) consisting of $N_{\mathrm{v}} \times N_{\mathrm{h}}$ isotropic antenna elements, whereas the UE is outfitted with an omni-directional antenna. The channel comprises N_{p} distinct propagation paths, each one characterized by its time of flight (ToF) τ_p and angle of arrival (AoA), represented by zenith angle $\theta_p \in [0, \pi]$ and azimuth angle $\varphi_p \in [0, \pi]$ for path $p = 1, \ldots, N_{\mathrm{p}}$. Defining with T_{s} the sampling interval and N_{c} the number of sub-carriers, we obtain a symbol duration of $T_{\mathrm{c}} = N_{\mathrm{c}} T_{\mathrm{s}}$. The frequency for the k-th sub-carrier is $f_k = \frac{k}{T_{\mathrm{c}}}$, $k = 0, \ldots, N_{\mathrm{c}} - 1$, and we assume that the cyclic-prefix duration of $T_{\mathrm{g}} = N_{\mathrm{g}} T_{\mathrm{s}}$ exceeds the channel's maximum delay τ_{MAX}, where N_{g} indicates the number of sampling intervals comprising a guard interval.

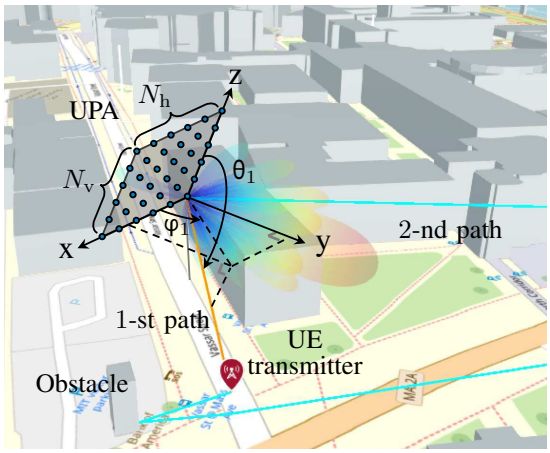
Assuming a sampling rate of $1/T_{\mathrm{s}}$ and treating each path as independent and wide-sense stationary, the channel frequency response (CFR) for the k-th sub-carrier over a UE-BS link is expressed as follows [17], [18]:

$$\mathbf{h}_k = \sum_{p=1}^{N_{\mathrm{p}}} \alpha_p e^{-j2\pi\tau_p f_k} \mathbf{e}(\theta_p, \varphi_p), \tag{1}$$

where $\mathbf{e}(\theta_p, \varphi_p) \in \mathbb{C}^{N_{\mathrm{h}} N_{\mathrm{v}}}$ is the array response vector [19], and $\alpha_p = \mathsf{a}_p e^{-j2\pi(\frac{\mathsf{d}_p}{\lambda_{\mathrm{c}}} - \nu_p \tau_p)}$ is the complex gain of p-th path which includes the Doppler frequency shift ν_p and has average power $\sigma_p^2 = \mathbb{E}\{|\mathsf{a}_p|^2\}$ and $\mathsf{d}_p = c\tau_p$ is the traveled distance (where c is the speed of light in air). The overall space-frequency channel response matrix (SFCRM) is $\mathbf{H} = [\mathbf{h}_0\ \mathbf{h}_1\ \ldots\ \mathbf{h}_{N_{\mathrm{c}}-1}] \in \mathbb{C}^{N_{\mathrm{h}} N_{\mathrm{v}} \times N_{\mathrm{c}}}$.



(a)



(b)

Fig. 1. (a) ADCPM fingerprint with $N_{\mathrm{h}} N_{\mathrm{v}} = 64$ angle indexes and $N_{\mathrm{g}} = 352$ delay indexes. (b) Corresponding deterministic macro paths from the UE to the BS.

B. Location Fingerprinting

Location estimation benefits significantly from transforming the channel response into the angle-delay domain, i.e., into the ADCPM domain. This conversion encapsulates in a convenient representation all the location-related parameters like ToF, AoA, and received signal strength (RSS) for each propagation path, including both line-of-sight (LoS) and NLoS components, which change depending on the surrounding environment, thus acting as location-specific signatures or fingerprints. The larger the number of antenna elements and bandwidth, the more precise and stable the estimation, owing to improved spatial and frequency resolution. To extract features from the angle-delay domain, we adopt the same methodology described in [6] and obtain the ADCPM $\boldsymbol{P} \in \mathbb{R}^{N_{\mathrm{h}} N_{\mathrm{v}} \times N_{\mathrm{g}}}$. For an example of ADCPM and related propagation paths, we refer to Fig. 1. In our experimental setup, we utilize the ADCPM $\boldsymbol{P}$ as the input $\mathbf{x}$ for the DL model, as a measure for performing positioning and subsequent tracking.

III. BNN Methodology

A. Problem Formulation

Consider a ML supervised regression setting where the target UE position t, here considered as a scalar, is modelled as:

$$\mathsf{t} = f(\mathbf{x}) + \boldsymbol{\varepsilon}(\mathbf{x}), \tag{2}$$

where $f(\mathbf{x})$ is a non-linear function that relates the ADCPM measurement $\mathbf{x}$ to the location and $\boldsymbol{\varepsilon}(\mathbf{x}) \sim \mathcal{N}\left(0, \sigma_{\boldsymbol{\varepsilon}}(\mathbf{x})^2\right)$. The aim is to train a neural network (NN) $y(\mathbf{x}, \boldsymbol{\theta})$ with parameters $\boldsymbol{\theta}$ to approximate $f(\mathbf{x})$ using an input training dataset $\mathcal{D} = \{(t_n, \boldsymbol{x}_n) \mid t_n \in \mathcal{D}_t, \boldsymbol{x}_n \in \mathcal{D}_{\boldsymbol{x}}\}_{n=1}^{N}$ with N training points. Assuming independence between target variables t_n and the Gaussian random noise $\boldsymbol{\varepsilon}$, we can write the likelihood function of $\boldsymbol{\theta}$ as:

$$p_{\mathsf{D_t}|\mathsf{D_x},\boldsymbol{\theta}}(\mathcal{D}_t|\mathcal{D}_{\boldsymbol{x}}, \boldsymbol{\theta}) = \prod_{n=1}^{N} \mathcal{N}\left(t_n; y(\boldsymbol{x}_n, \boldsymbol{\theta}), \sigma_{\boldsymbol{\varepsilon}}(\boldsymbol{x}_n)^2\right). \tag{3}$$

In traditional ML frameworks, a discriminative probabilistic approach is utilized [20], where the NN parameters (deterministic variables) are computed through maximum likelihood estimation (MLE). Conversely, Bayesian frameworks employ a stochastic network characterized by random parameters with a prior distribution $p_{\boldsymbol{\theta}}(\boldsymbol{\theta})$, which reflects the model's uncertainty due to the limited size of the training dataset. Bayesian NN models adopt a generative approach by calculating the so-called *posterior predictive distribution* [21]:

$$p_{\mathsf{t}|\mathbf{x},\mathsf{D}}(t|\boldsymbol{x}, \mathcal{D}) = \int_{\boldsymbol{\theta'}} p_{\mathsf{t}|\mathbf{x},\boldsymbol{\theta}}(t|\boldsymbol{x}, \boldsymbol{\theta'}) p_{\boldsymbol{\theta}|\mathsf{D}}(\boldsymbol{\theta'}|\mathcal{D}) d\boldsymbol{\theta'}, \tag{4}$$

where $p_{\boldsymbol{\theta}|\mathsf{D}}(\boldsymbol{\theta}|\mathcal{D})$ is the computational intractable posterior distribution. As a solution, most BNN methodologies strive to approximate $p_{\boldsymbol{\theta}|\mathsf{D}}(\boldsymbol{\theta}|\mathcal{D})$ through a sampling process and then compute (4) using Monte Carlo (MC) sampling as follows:

$$p_{\mathsf{t}|\mathbf{x},\mathsf{D}}(t|\boldsymbol{x}, \mathcal{D}) \backsimeq \frac{1}{L} \sum_{\ell=1}^{L} p(t|\boldsymbol{x}, \boldsymbol{\theta}_\ell), \tag{5}$$

where L is the number of samples $\boldsymbol{\theta}_\ell$ drawn from $p_{\boldsymbol{\theta}|\mathsf{D}}(\boldsymbol{\theta}|\mathcal{D})$. From (5), we obtain the mean prediction, i.e., *predictive mean*, as:

$$\mathbb{E}\{\mathsf{t}|\mathbf{x}, \mathsf{D}\} \backsimeq \frac{1}{L} \sum_{\ell=1}^{L} \int_{t'} t' p(t'|\boldsymbol{x}, \boldsymbol{\theta}_\ell) dt' \backsimeq \frac{1}{L} \sum_{\ell=1}^{L} y(\boldsymbol{x}, \boldsymbol{\theta}_\ell). \tag{6}$$

For computing the variance of the prediction, i.e., *predictive variance*, we also need to estimate the variance of the noise in (2). This is usually performed through an additional NN output $y_{\mathrm{al}}(\mathbf{x}, \boldsymbol{\theta})$ according to the model [22]:

$$\sigma_{\boldsymbol{\varepsilon}}(\mathbf{x})^2 = y_{\mathrm{al}}(\mathbf{x}, \boldsymbol{\theta}) + \xi_{\mathrm{al}}, \tag{7}$$

where $\xi_{\mathrm{al}} \sim \mathcal{N}(0, \sigma^2_{\xi_{\mathrm{al}}})$. The predictive variance is finally computed as:

$$\begin{aligned}\mathbb{V}\{\mathsf{t}|\mathbf{x}, \mathsf{D}\} &\backsimeq \frac{1}{L} \sum_{\ell=1}^{L} \int_{t'} (t' - \mathbb{E}\{\mathsf{t}|\mathbf{x}, \mathsf{D}\})^2 p(t'|\boldsymbol{x}, \boldsymbol{\theta}_\ell) dt' \\ &\backsimeq \frac{1}{L} \sum_{\ell=1}^{L} y(\boldsymbol{x}, \boldsymbol{\theta}_\ell)^2 - \left(\frac{1}{L} \sum_{\ell=1}^{L} y(\boldsymbol{x}, \boldsymbol{\theta}_\ell)\right)^2 \\ &+ \frac{1}{L} \sum_{\ell=1}^{L} y_{\mathrm{al}}(\boldsymbol{x}, \boldsymbol{\theta}_\ell),\end{aligned} \tag{8}$$

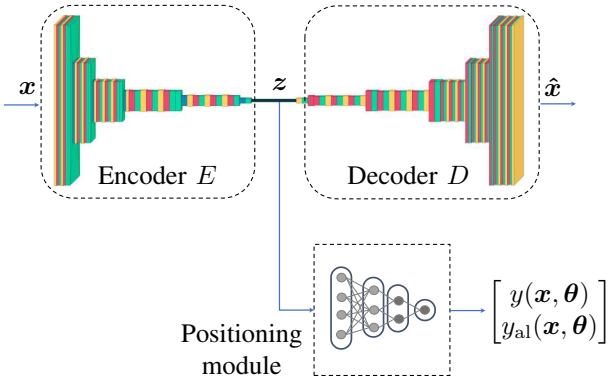


Fig. 2. Proposed DL model composed of an autoencoder (AE) and a positioning module, which outputs the target estimate $y(\boldsymbol{x}, \boldsymbol{\theta})$ and related aleatoric uncertainty $y_{\mathrm{al}}(\boldsymbol{x}, \boldsymbol{\theta})$.

where the first two terms are the *epistemic uncertainty* prediction, while the last term is the *aleatoric uncertainty* prediction.

B. DL Model Based on Autoencoder

For position estimation, we propose the integration of an AE within each BS to extract essential features from the sparse ADCPM samples $\boldsymbol{x}$, as illustrated in Fig. 2. The encoder function $E(\boldsymbol{x})$ translates these samples into latent features $\boldsymbol{z}$, capturing the channel's intrinsic location-specific information. In turn, the decoder function $D(\boldsymbol{z})$ reconstructs the original input, yielding $\hat{\boldsymbol{x}}$. The AE is designed to minimize the reconstruction error $\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2$ [23], facilitating the model's ability to replicate the input $\boldsymbol{x}$ through the compressed representation in $\boldsymbol{z}$. Moreover, the DL model integrates a multi-layer perceptron (MLP) positioning module that leverages the compact latent features to estimate the 3D position of the target.

For prediction uncertainty, we employ the stochastic gradient Langevin dynamics (SGLD) BNN algorithm [24], which guarantees to directly sample from the real posterior $p_{\boldsymbol{\theta}|\mathsf{D}}(\boldsymbol{\theta}|\mathcal{D})$. To speed up convergence, the AE is treated as a standard NN, whereas the positioning module is trained as a full BNN with SGLD optimizer. The loss function for single input $\boldsymbol{x}$ is:

$$E_{\mathrm{rr}}(\boldsymbol{\theta}|\boldsymbol{x}) = \lambda_{\mathrm{pos}}\, A(\boldsymbol{\theta}|\boldsymbol{x}) + \lambda_{\mathrm{rec}} \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2, \tag{9}$$

where λ_{pos} determines the significance of position estimation, λ_{rec} modulates sample reconstruction and $A(\boldsymbol{\theta}|\boldsymbol{x})$ is the corresponding matrix version of:

$$A(\boldsymbol{\theta}|\boldsymbol{x}) = \frac{1}{2} \log\left(y_{\mathrm{al}}(\boldsymbol{x}, \boldsymbol{\theta})\right) + \frac{\|t - y(\boldsymbol{x}, \boldsymbol{\theta})\|_2^2}{2\, y_{\mathrm{al}}(\boldsymbol{x}, \boldsymbol{\theta})}. \tag{10}$$

Equation (10) is derived from $A(\boldsymbol{\theta}|\boldsymbol{x}) = -\log p(t|\boldsymbol{x}, \boldsymbol{\theta})$ and permits the training of the BNN with the SGLD optimizer.

IV. Tracking with BNN Methodology

A. Tracking Problem

We address the problem of Bayesian tracking for a mobile target, characterized by a non-linear state evolution [25]:

$$\mathbf{t}_n = f_n^{(\mathrm{t})}(\mathbf{t}_{n-1}) + \boldsymbol{\varepsilon}_{n-1}^{(\mathrm{t})}, \tag{11}$$

where $f_n^{(\mathrm{t})}(\mathbf{t}_{n-1})$ denotes the non-linear state transition function at time $n-1$, and $\boldsymbol{\varepsilon}_{n-1}^{(\mathrm{t})}$ represents a non-independent and identical distributed (IID) noise component. The tracking system's measurements are modeled as:

$$\mathbf{x}_n = f_n^{(\mathrm{x})}(\mathbf{t}_n) + \boldsymbol{\varepsilon}_n^{(\mathrm{x})}, \tag{12}$$

where $f_n^{(\mathrm{x})}(\mathbf{t}_n)$ links the state to the measurement via a non-linear relationship, and $\boldsymbol{\varepsilon}_n^{(\mathrm{x})}$ is another non-IID noise component. Additionally, we denote the cumulative set of measurements up to time n as $\mathbf{x}_{1:n} = \{\mathbf{x}_i, i = 1, \ldots, n\}$. The Bayesian tracking proceeds by alternating two phases at each time step. In the *prediction phase*, the state is updated using the Chapman–Kolmogorov equation [25]:

$$p(\boldsymbol{t}_n|\boldsymbol{x}_{1:n-1}) = \int p(\boldsymbol{t}_n|\boldsymbol{t}_{n-1}) p(\boldsymbol{t}_{n-1}|\boldsymbol{x}_{1:n-1}) d\boldsymbol{t}_{n-1}, \tag{13}$$

where $p(\boldsymbol{t}_{n-1}|\boldsymbol{x}_{1:n-1})$ and $p(\boldsymbol{t}_n|\boldsymbol{x}_{1:n-1})$ are the posterior PDF at time $n-1$ and the prior PDF at time n, respectively. The subsequent *update phase* then incorporates measurements to refine the posterior at time n:

$$p(\boldsymbol{t}_n|\boldsymbol{x}_{1:n}) \propto p(\boldsymbol{x}_n|\boldsymbol{t}_n) p(\boldsymbol{t}_n|\boldsymbol{x}_{1:n-1}). \tag{14}$$

To incorporate complex measurements such as the ADCPM, an efficient representation or computation of the non-linear function $f_n^{(\mathrm{x})}(\mathbf{t}_n)$ becomes crucial. NN can be employed to approximate such function from training data, as proposed in the next section.

B. Incorporating BNN into Tracking Systems

The integration of BNN into tracking systems involves a collection of BSs, denoted by $\mathcal{S}_{\mathrm{BS}}$. During the offline training phase, each base station j exploits a local dataset $\mathcal{D}^{(j)} = \left\{\boldsymbol{t}_n^{(j)}, \boldsymbol{x}_n^{(j)}\right\}_{n=1}^{N^{(j)}}$ to train its respective BNN. In the online tracking phase, a subset of base stations $\mathcal{S}_{\mathrm{BS},n} \subseteq \mathcal{S}_{\mathrm{BS}}$ detects the target at timestep n and performs tracking according to Algorithm 1. We propose to maintain the prediction phase unchanged while seamlessly integrate BNN into the update phase. This strategy eases the incorporation into existing algorithms, allowing for the replacement or augmentation of the update component with BNN, and permits more precise and accurate training procedures. Upon concluding the prediction phase and acquiring the prior PDF $p(\boldsymbol{t}_n|\boldsymbol{x}_{1:n-1})$, each base station j in $\mathcal{S}_{\mathrm{BS},n}$ provides the posterior predictive distribution $p\big(\boldsymbol{t}_n|\boldsymbol{x}_n^{(j)}, \mathcal{D}^{(j)}\big)$, denoted as $p\big(\boldsymbol{t}_n|\boldsymbol{x}_n^{(j)}\big)$. Then, since at each timestep n, the BNN lacks prior knowledge of the target's previous position at $n-1$, formally represented as $\mathbf{t}_n \sim \mathcal{U}\big(\boldsymbol{t}_{\min}^{(j)}, \boldsymbol{t}_{\max}^{(j)}\big)$, where $\boldsymbol{t}_{\min}^{(j)}$ and $\boldsymbol{t}_{\max}^{(j)}$ are the limits of the coverage area of the j-th BS, we can write that $p\big(\boldsymbol{x}_n^{(j)}|\boldsymbol{t}_n\big) \propto p\big(\boldsymbol{t}_n|\boldsymbol{x}_n^{(j)}, \mathcal{D}\big)$. This formulation facilitates the fusion of multiple base station predictions with the prior PDF on the target state, resulting in an updated posterior.

Algorithm 1 Tracking procedure

Input: Posterior $p(\boldsymbol{t}_{n-1}|\boldsymbol{x}_{1:n-1})$ at time $n-1$ ▷ Run at BS j at timestep n

Output: Posterior $p(\boldsymbol{t}_n|\boldsymbol{x}_{1:n})$ at time n

1: Compute prediction phase in (13)
2: Measure sample $\boldsymbol{x}_n^{(j)}$
3: Compute $p\Big(\boldsymbol{x}_n^{(j)}|\boldsymbol{t}_n\Big) \propto p\Big(\boldsymbol{t}_n|\boldsymbol{x}_n^{(j)}, \mathcal{D}\Big)$
4: **for** $j' \in \mathcal{S}_{\mathrm{BS},n} \backslash \{j\}$ **do**
5: Send $p\Big(\boldsymbol{x}_n^{(j)}|\boldsymbol{t}_n\Big)$ to j'
6: Receive $p\Big(\boldsymbol{x}_n^{(j')}|\boldsymbol{t}_n\Big)$ from j'
7: **end for**
8: Update $p(\boldsymbol{t}_n|\boldsymbol{x}_{1:n}) \propto \prod_{j \in \mathcal{S}_{\mathrm{BS},n}} p\Big(\boldsymbol{x}_n^{(j)}|\boldsymbol{t}_n\Big) p(\boldsymbol{t}_n|\boldsymbol{x}_{1:n-1})$

Practically, the posterior predictive distribution is described by two parameters, i.e., the predictive mean (6) and the predictive variance (8). Therefore, we propose to approximate the likelihood function obtained by each BS with a multivariate normal distribution as:

$$\begin{aligned} p\Big(\boldsymbol{x}_n^{(j)}|\boldsymbol{t}_n\Big) &\backsimeq \mathcal{N}\Big(\boldsymbol{x}_n^{(j)}; \mathbb{E}\Big\{\mathbf{t}_n|\mathbf{x}_n^{(j)}, \mathsf{D}\Big\}, \mathbb{V}\Big\{\mathbf{t}_n|\mathbf{x}_n^{(j)}, \mathsf{D}\Big\}\Big) \\ &= \mathcal{N}\Big(\boldsymbol{x}_n^{(j)}; \boldsymbol{\mu}_n^{(j)}, \boldsymbol{\Sigma}_n^{(j)}\Big). \end{aligned} \tag{15}$$

This approximation makes it very easy and effective computing (14) by combining the likelihood functions of the BSs as $p(\boldsymbol{x}_n|\boldsymbol{t}_n) = \prod_{j \in \mathcal{S}_{\mathrm{BS},n}} p\Big(\boldsymbol{x}_n^{(j)}|\boldsymbol{t}_n\Big) \propto \mathcal{N}\Big(\boldsymbol{x}_n; \boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n\Big)$ where [26]:

$$\boldsymbol{\mu}_n = \boldsymbol{\Sigma}_n \left(\sum_{j \in \mathcal{S}_{\mathrm{BS},n}} {\boldsymbol{\Sigma}_n^{(j)}}^{-1} \boldsymbol{\mu}_n^{(j)} \right), \tag{16}$$

$$\boldsymbol{\Sigma}_n = \left(\sum_{j \in \mathcal{S}_{\mathrm{BS},n}} {\boldsymbol{\Sigma}_n^{(j)}}^{-1} \right)^{-1}. \tag{17}$$

V. Numerical Results

A. Simulation Setup

For the simulations, we employ a ray-tracing method [27], utilizing the Wireless InSite 3D prediction tool [28], in conjunction with a 5G new radio (NR) MATLAB clustered delay line (CDL) channel model at carrier frequency $f_{\mathrm{c}} = 28$ GHz and bandwidth $B = 400$ MHz. The experiment includes a 3GPP urban micro (UMi) setting [15] within a 1000 × 1000 m area near the MIT campus in Cambridge, MA, USA, shown in Fig. 3. The setting includes 19 sites, each with an inter-site distance (ISD) of 200 m, forming a hexagonal layout. Each site comprises 3 BSs, elevated 25 m and angled 120 degrees apart. The BSs employ a UPA configuration, derived from [29], with $N_{\mathrm{v}} = N_{\mathrm{h}} = 8$ antenna elements and have a 15-degree mechanical downtilt. UE trajectories are generated by the SUMO software [16], which simulates realistic vehicular traffic. During the 600-second simulation, 100 vehicle trajectories were created, collecting data points every second. This resulted in 2593 and 702 positions for training and testing, respectively,

Fig. 3. 3D map representation of the ray-tracing scenario in the area of Massachusetts Institute of Technology (MIT), Cambridge, USA.

and approximately $9.3 \cdot 10^4$ and $2.5 \cdot 10^4$ training and testing ADCPM samples, respectively.

The AE segment of the DL positioning model is built using the Segnet architecture [30], handling the sparsity of the ADCPM input and facilitating robust feature extraction crucial for precise positioning. The positioning module is composed of an MLP with a number of neurons in each layer of: $[16, 32, 64, 128, 256, 512, 256, 128, 64, 32, 16, 9]$, and GeLu activation functions. Moreover, softplus activation functions are placed at the output of the positioning module in the diagonal entries to ensure non-negative variances. Additionally, regularization terms are added to the diagonals of the covariance matrix predictions to guarantee non-singularity and symmetry. The models were trained for 600 epochs with a batch size of $M = 256$ and $L = 40$. The learning rate was set to 10^{-5}, to ensure stable convergence. The hyper-parameters λ_{rec} and λ_{pos} were empirically set using a grid method in the range $[0.1, 1]$ with a step size of 0.1, resulting in values of 0.1 and 0.9, respectively. The prior distribution of the parameters was chosen to be spherical Gaussian, as in standard SGLD initialization, with regularizer $\lambda_{\text{prior}} = 0.1$.

B. Mobile positioning in urban environment

This experiment evaluates the performance of the proposed BNN-based tracking approach with respect to a geometric-based EKF and a state-of-the-art TCN model [14]. To ensure a fair assessment, both the BNN method and the EKF utilize identical motion models, i.e., a random walk with a positional standard deviation of 2 m. The EKF employs traditional geometric localization techniques using LoS time difference of flight (TDoF) measurements, derived from cross-correlation with the sounding reference signal (SRS) according to 3GPP standards, and LoS AoA measurements via the multiple signal classification (MUSIC) algorithm [31]. Additionally, in light of the substantial signal obstruction caused by urban structures, UEs are outfitted with global navigation satellite systems (GNSS) receivers that provide state measurements with a Gaussian noise standard deviation of 2 m. Conversely, the BNN-based tracking relies on both LoS and NLoS ADCPM measurements. For the TCN model, it adopts a 1D CIR representation of the closest BS, converting the 2D ADCPM into a singular vector, as recommended in [14].

The tracking results are presented in Fig. 4 through the absolute location error per timestep and the number of LoS BSs. Additionally, Fig. 5 presents the CDF of the positioning errors across all methodologies. From Fig. 4, we can note that the EKF struggles to maintain a location accuracy below 2 meters when fewer than three LoS BSs are available. Indeed, despite the dense UMi setting, we observe 1.6 LoS BSs on average. In contrast, the TCN model exhibits marginally superior performance, accurately tracking UE positions even with a single BS measurement, due to its fingerprinting approach. Nonetheless, the BNN-based tracking demonstrates consistent sub-meter precision even in the absence of LoS BSs. The superior performance with respect to traditional methods is derived from its ability to coherently merge multiple NLoS position estimates. On the contrary, the BNN-based model outperforms the TCN model thanks to the exploitation of the 2D ADCPM within an AE structure that discerns spatial correlations in the input data. Moreover, the fusion of BSs measurements is weighted by the uncertainty of the model, resulting in more consistent performances. The CDF of the absolute errors further confirms the BNN-based tracking's dominance, achieving a median error of 46 cm and staying under 1 meter in 87% of the cases.

VI. Conclusion

In this work, we tackled the challenge of 6G tracking in urban areas characterized by significant signal obstructions. We proposed an integration of BNNs into a Bayesian tracking system, where full CIRs, i.e., 2D ADCPM, are processed as measurements through an AE-based DL model. Realistic simulations, within a C-ITS environment and 3GPP-compliant UMi ray-tracing scenario, show superior performances compared to geometric-based tracking filters and advanced TCN models, achieving a median absolute positioning error of just 46 cm.

References

[1] 3GPP, "Study on artificial intelligence (AI)/machine learning (ML) for NR air interface," 3rd Generation Partnership Project (3GPP), TR 38.843, 2022, version 18.0.0.

[2] "Summary of RAN rel-18 workshop," 2021, document RWS-210659, 3GPP RAN Chair, 3GPP Technical.

[3] J. Zheng, J. Zhang *et al.*, "Asynchronous cell-free massive MIMO with rate-splitting," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 5, p. 1366–1382, 2023.

[4] X. Lin, "An overview of 5G advanced evolution in 3GPP release 18," *IEEE Commun. Stand. Mag.*, vol. 6, no. 3, pp. 77–83, 2022.

[5] L. Italiano, B. Camajori Tedeschini *et al.*, "A tutorial on 5G positioning," *ArXiv*, 2023.

[6] B. Camajori Tedeschini and M. Nicoli, "Cooperative deep-learning positioning in mmWave 5G-advanced networks," *IEEE J. Sel. Areas Commun.*, vol. 41, pp. 1–18, 2023.

[7] B. Camajori Tedeschini, M. Nicoli *et al.*, "On the latent space of mmWave MIMO channels for NLOS identification in 5G-advanced systems," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 6, pp. 1655–1669, 2023.

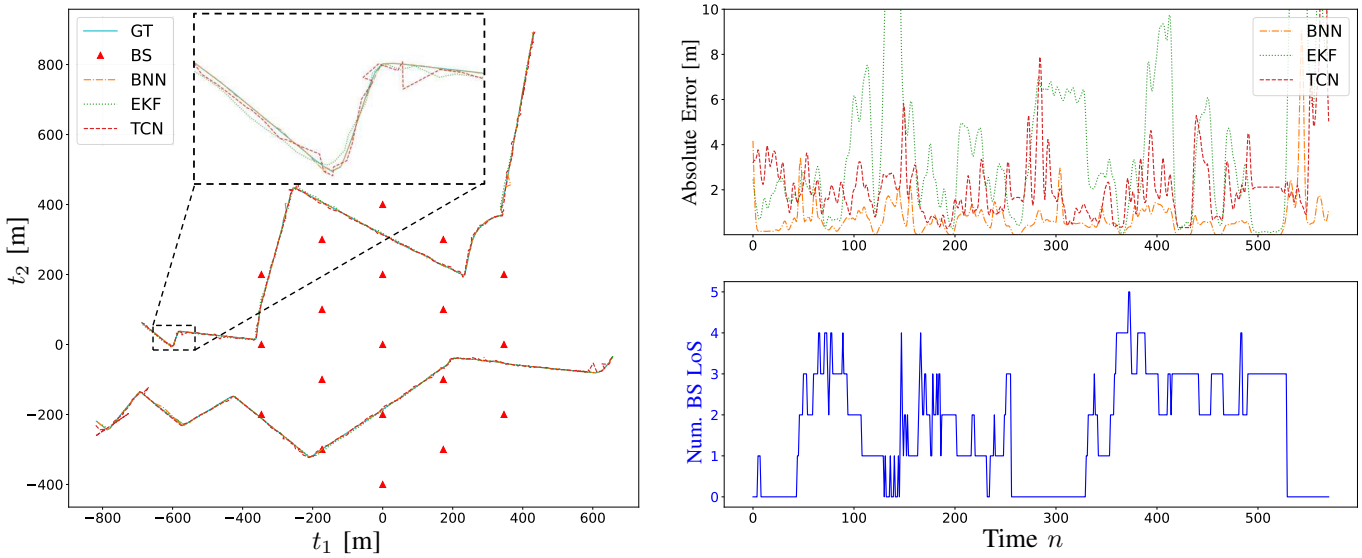


Fig. 4. Ground truth and predicted UE trajectories within the UMi urban scenario of Fig. 3 (left). Tracking performances in terms of absolute location error and number of BSs in LoS per timestep (right).

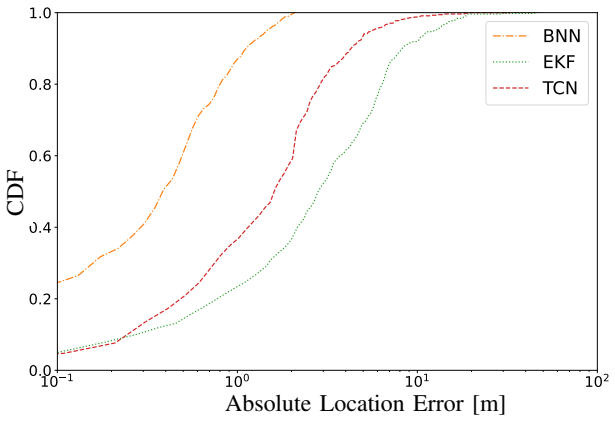


Fig. 5. Positioning performances in terms of CDF of the distance error for the proposed BNN-based tracking, EKF and TCN.

[8] B. Camajori Tedeschini, M. Brambilla *et al.*, "Cooperative lidar sensing for pedestrian detection: Data association based on message passing neural networks," *IEEE Trans. Signal Process.*, vol. 71, pp. 3028–3042, 2023.

[9] L. V. Jospin, H. Laga *et al.*, "Hands-on Bayesian neural networks—a tutorial for deep learning users," *IEEE Comput. Intell. Mag.*, vol. 17, no. 2, pp. 29–48, 2022.

[10] M. A. M. Sadr, J. Gante *et al.*, "Uncertainty estimation via monte carlo dropout in CNN-based mmWave MIMO localization," *IEEE Signal Process. Lett.*, vol. 29, pp. 269–273, 2022.

[11] M. Zecchin, S. Park *et al.*, "Robust Bayesian learning for reliable wireless AI: Framework and applications," *IEEE Trans. Cogn. Commun. and Netw.*, vol. 9, no. 4, pp. 897–912, 2023.

[12] H. Kim, H. Wymeersch *et al.*, "5G mmWave vehicular tracking," in *2018 52nd Asilomar Conf. Signals, Syst., Comput.* IEEE, 2018, pp. 541–547.

[13] Y. Ruan, L. Chen *et al.*, "Hi-loc: Hybrid indoor localization via enhanced 5G NR CSI," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–15, 2022.

[14] J. Gante, G. Falcão *et al.*, "Deep learning architectures for accurate millimeter wave positioning in 5G," *Neural Process. Lett.*, vol. 51, no. 1, pp. 487–514, 2020.

[15] 3GPP, "Study on NR positioning support," 3rd Generation Partnership Project (3GPP), TR 38.855, 2019, version 16.0.0.

[16] P. Alvarez Lopez, M. Behrisch *et al.*, "Microscopic traffic simulation using SUMO," in *2018 21st Int. Conf. Intell. Transp. Syst. (ITSC)*. IEEE, 2018, p. 2575–2582.

[17] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge University Press, 2005, oCLC: ocm57751753.

[18] X. Sun, X. Gao *et al.*, "Single-site localization based on a new type of fingerprint for massive MIMO-OFDM systems," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6134–6145, 2018.

[19] H. L. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*, 1st ed. Wiley, 2002.

[20] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006, vol. 4, no. 4.

[21] M. Abdar, F. Pourpanah *et al.*, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Inf. Fusion*, vol. 76, no. C, p. 243–297, 2021.

[22] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?" *ArXiv*, p. 5580–5590, 2017.

[23] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, 2015.

[24] M. Welling and Y. W. Teh, "Bayesian learning via stochastic gradient Langevin dynamics," in *Proc. 28th Int. Conf. Int. Conf. Mach. Learn.*, ser. ICML'11. Omnipress, 2011, p. 681–688.

[25] M. Arulampalam, S. Maskell *et al.*, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, 2002.

[26] K. B. Petersen, M. S. Pedersen *et al.*, "The matrix cookbook," Nov. 2012, version 20121115.

[27] Hsueh-Jyh Li, Cheng-Chung Chen *et al.*, "Applicability of ray-tracing technique for the prediction of outdoor channel characteristics," *IEEE Trans. Veh. Technol.*, vol. 49, no. 6, pp. 2336–2349, 2000.

[28] R. Inc., "Wireless insite 3D wireless prediction software," 2023.

[29] "Guidelines for evaluation of radio interface technologies for IMT-2020," Report International Telecommunication Union (ITU)-R M, https://www.itu.int/md/R15-SG05-C-0057, 2017.

[30] V. Badrinarayanan, A. Kendall *et al.*, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, 2017.

[31] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, 1986.