Estimating Multipath Component Delays with Transformer Models

Jonathan Ott[1], Maximilian Stahlke[1, 2], Tobias Feigl[1], and Christopher Mutschler[1]

***Abstract*— Multipath in radio propagation provides essential environmental information that is exploited for positioning or channel-simultaneous localization and mapping. This enables accurate and robust localization that requires less infrastructure than traditional methods. A key factor is the reliable and accurate extraction of multipath components (MPC). However, limited bandwidth and signal fading make it difficult to detect and determine the parameters of the individual signal components.**

In this article, we propose multipath delay estimation based on a Transformer (TF) neural network. In contrast to the state-of-the-art, we implicitly estimate the number of MPCs and achieve subsample accuracy without using computationally intensive super-resolution techniques. Our approach outperforms known methods in detection performance and accuracy at different bandwidths. Our ablation study shows exceptional results on simulated and real datasets and generalizes to unknown radio environments.

***Index Terms*— 5G, Attention, Multipath, Radio localization, Transformer, UWB, Wireless channel estimation**

I. Introduction

Indoor positioning is a key enabler for many applications such as emergency services [1], elderly care [2], and smart logistics [3]. Radio frequency (RF)-based systems are commonly utilized in this context. Conventional RF-based positioning systems exploit signal properties such as the received signal strength (RSS) [4], the time of arrival (ToA) [5], [6] or the angle of arrival (AoA) [7], which are directly related to the geometry between the transmitter and the receiver. However, these systems neglect the spatial information that is associated with interactions of the electromagnetic waves with obstacles in the environment. Reflection, diffraction, and scattering cause multiple propagation paths that arrive with different delays at the receiver. These signal components inherently contain information about the geometry of the environment. Modern approaches, such as multipath (MP)-assisted positioning [8], [9] and channel-simultaneous localization and mapping (CSLAM) [10], [11], exploit these signal components to increase the robustness as well as the accuracy of the position estimates, and reduce the amount of anchors. Limited bandwidth and dense MP introduce fading to the signal [12]. Accordingly, the extraction of spatial information from complex channel states is challenging.

Renowned analytical MP component (MPC) estimators are computationally intensive and at the same time limited by strong assumptions (thresholds or expected number of MPCs) [13]–[18] and deep learning (DL)-based methods [19] recently outperformed them in terms of accuracy and computational efficiency. However, the DL methods also use a threshold value that requires experimental adjustments. And to our knowledge, all methods are limited in their resolution due to discretization. Super-resolution techniques such as MUSIC increase resolution but are limited by the density of the steering vectors [13] and the DL method only estimates MPCs as accurately as the sampling resolution of the underlying channel impulse response (CIR). To address these weaknesses, in our previous work [20] we introduced a novel MPC delay estimation pipeline based on the transformer (TF) architecture [21]. Unlike analytical methods that require additional pre-processing mechanics, or DL that uses segmentation [19], our TF implicitly learns from the dataset to estimate the number of MPCs, i.e., the model order. In addition, our TF analyzes not only local patterns, but also dependencies between time steps (samples) in a (CIR) signal. And our TF exploits optimal (sub-sample) resolutions of the CIR signal when it uses spatially consistent measurements from a specific environment as it learns the relationships of MPCs directly from the signals. Through these properties, our TF [20] estimates MPC delays on complex-valued CIRs more accurately and robustly, reduces methodological and computational complexity, and outperforms existing MPC estimators.

In this article we complement our previous work [20] with additional experiments on input data, generalization, further comparison methods, and uncertainty estimation. We show how to adapt our TF on power delay profile (PDP) input signals to (at least) halve the effort of our previous (CIR) data processing without significant performance loss. We show that our TF generalizes to non-line-of-sight (NLOS) ultra-wideband (UWB) data. We extend our computational effort analysis to include training time and compare with MUSIC.

This work was supported by the Federal Ministry of Education and Research of Germany in the programme of “Souverän. Digital. Vernetzt.” joint project 6G-RIC (16KISK020K), the Bavarian Ministry for Economic Affairs, Regional Development, and Energy through the Center for Analytics - Data Applications (ADA-Center) within the framework of ”BAYERN DIGITAL II” (20-3410-2-9-8), and by the German Federal Ministry for Economic Affairs and Climate Action (BMWK) within the project ”5GILABB - 5G Industrial Lab Bischheim” (01MJ22017A).

[1]Fraunhofer IIS, Fraunhofer Institute for Integrated Circuits IIS, Division Positioning and Networks, Nuremberg, Germany (e-mails: firstname.lastname@iis.fraunhofer.de); [2]Department Artificial Intelligence in Biomedical Engineering (AIBE), Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Erlangen, Germany

Author Contributions: J.O. wrote and edited the article, reviewed related work, wrote the code, performed experiments and conceptualized the methodology. J.O., M.S. and T.F. interpreted and discussed the results. M.S., T.F. and C.M. reviewed and edited the article. C.M. supervised the work.

We are also the first to analyze the uncertainty of learning-based MPC delay estimation to provide a detailed insight.

The rest of the article is structured as follows. Sec. II reviews related work. Sec. III describes the problem. Sec. IV introduces our method. Sec. V describes our experiments. Sec. VI presents the results. Sec. VII discusses the limitations of our pipeline. Sec. VIII concludes.

II. Related Work

Positioning that benefits from accurate MPC-delays includes methods that leverage environmental knowledge to enhance the position estimates [9], [22] and CSLAM-based methods [10], [11]. Also channel charting algorithms [23] benefit from MP information, such as multi-point channel charting [24] or time-distance based approaches [25], [26]. This Section discusses analytical and data-driven approaches that extract MPCs from channel measurements.

The most prominent **analytical** techniques are MUSIC [17] and Esprit [15]. These subspace-based approaches exploit the orthogonality of the signal- and noise-subspaces. Under the assumption of non overlapping MPCs, these algorithms potentially extract MPCs at high accuracy. Maximum likelihood (ML)-based methods estimate in an iterative manner. These include, the EM [14] and SAGE [13] algorithms. RiMAX [16] extends their idea by incorporating diffuse MP. Kulmer *et al.* [18] introduces a optimization criterion that detects partly overlapping MPCs. However, both the subspace and ML-based approaches are computationally intensive due to complex or sequential operations and limited by strong assumptions, e.g., pre-known number of MPCs. Inherently, they only search for local occurrences of the (input) signal and are therefore unable to extract and describe significant correlations between (global) MPCs that represent the environment. These significantly lower positioning performance. They also rely on an a-priori estimation of the number of MPCs, i.e., model order. Typically, criteria such as the Akaike Information Criterion [27], the Bayesian Information Criterion [28] or Minimum Description Length [17] estimate the model order. However, it is well known that the correct number of MPCs (model order) is difficult to obtain at low signal-to-noise ratio (SNR) [29].

Recently, **data-driven** techniques compensate for the weaknesses of analytical methods. Yang *et al.* [30] estimated the model order using DL. However, to eliminate their need, Kram *et al.* [19] introduced a MPC delay estimation pipeline based on the U-Net architecture [31]. They formulate MPC estimation as a time series segmentation task that determines the probability of whether a MPC occurs in a predefined time interval, e.g., a sampling time step, or not. Kram *et al.* [19] also introduced a detection threshold that tunes a trade-off between the detection rate and wrong detections. They outperform the state-of-the-art w.r.t. computational cost, detection rate, and accuracy. However, their method's accuracy is limited by the temporal resolution of the CIR. So, their method cannot resolve multiple MPCs that occur in the same time interval. Wang *et al.* [32] show that U-Net has an inductive bias towards locality as it is based on convolutional layers. Thus, U-Net only recognizes temporally adjacent and limited patterns in the input signal. Hence, U-Net only searches for local occurrences of the waveform, and does not capture correlations between MPCs, which reduces the positioning accuracy.

To address these weaknesses, we formulate the problem as an autoregressive task that learns to estimate MPC delays for each CIR. Due to the low temporal resolution of a CIR, a time step may incorrectly describe MPCs that are actually different in time. As the temporal resolution of our TF is not limited, it may capture these deficiencies. And, in contrast to the state-of-the-art, our TF captures MPCs dependencies in the entire input signal, i.e., dataset. So, our TF learns environment-specific interactions between MPCs, is not limited by a single CIR's temporal resolution, and increases the performance of the MPC estimation and positioning. In addition, our method saves time as it does not require handcrafted tuning parameters.

III. Problem Description

We characterize the complex-valued baseband signal $r(t)$ as the outcome of transmitting a pulse $s(t)$ through the channel $h(t)$ [33] such that

$$r(t) = s(t) * h(t) + w(t), \tag{1}$$

where the CIR $h(t)$ describes the radio channel

$$h(t) = \sum_{m=1}^{M} \alpha_m \delta(t - \tau_m) + \nu(t), \tag{2}$$

as a linear combination of M deterministic signal components with complex amplitude α_m, delay τ_m, and diffuse nondeterministic MP $\nu(t)$, which are characterized by a stochastic process [33]. To take non-spatial signal components into account, noise $w(t)$ is added. Limited signal bandwidth and complex MP-rich environments cause distortions and overlaps of MPCs in the CIR measurement. Hence, detecting the MPCs, that significantly vary in number depending on the environment, and correctly estimating τ_m are challenging tasks. Moreover, MPC parameter estimation algorithms must handle low resolutions due to a low sampling rate of CIR measurements. These challenges are even more complicated in realistic applications when only PDP measurements are available. In contrast to CIR measurements, PDPs

$$p(t) = |r(t)|^2, \tag{3}$$

describe the signal in terms of power and path delays, but neglect phase information. However, processing PDP is less computationally expensive compared to complex-valued CIR.

IV. Multipath Delay Estimation using Transformer

This Section introduces our novel TF architecture (Sec. IV-A), data pre- and post-processing (Sec. IV-B), and training and inference phases (Sec. IV-D).

A. Transformer Architecture

Our Transformer (TF) architecture [21] is structured as Encoder-Decoder (see Fig. 1 left: Encoder, right: Decoder). Functional entities consist of a series of N_b blocks with identical sublayers. The Encoder maps an input sequence to a representation, which is provided to each block of the autoregressive Decoder as a context vector. The key feature of the TF is its exclusive use of multi-head attention layers to express correlations between the indices or positions in a given sequence. The attention layers compare the entire sequence of inputs (i.e., CIR) in terms of their correlations. Based on that, weights are calculated that determine how much each position contributes to the current representation calculation step. The idea is that the mechanism “attends” to different positions at each additional layer and attention-head. A major benefit in using attention layers lies in the capturing of long term dependencies between elements in the sequence [34]. Note that these mechanisms are key to learning sub-samples and accurately extracting MPCs. Each Encoder block comprises multi-head self-attention stage with N_h heads, followed by a position-wise feed-forward network (FFN). The FFN implements two consecutive convolutional layers with a kernel size of 1×1. The position-wise FFN applies the same weights to each position of a given sequence. The visible layers of each FFN have a dimension of d_m and the hidden layers have a dimension of d_{ff}. In this context d_m refers to the size of the input embeddings. Considering the Decoder block, an Encoder-Decoder multi-head attention layer with N_h heads is incorporated between the self-attention layer and the FFN. So, the Decoder attends to the context vector provided by the Encoder. Both, the attention layer and the FFNs, are bypassed by a residual connection. We employ LayerNorm and Dropout for regularization. Inspired by Devlin *et al.* [35] and Radford *et al.* [36], we replace the ReLU with the GeLU activation function throughout the architecture, as GeLU approximates complex functions more efficiently than ReLU due to its nonconvexity and nonmonotonicity [37].

B. Pre- and Post-Processing

Fig. 1 shows our extensions together with the TF. We preprocess the T_r samples of long sequences of complex raw channel measurements $\boldsymbol{r}$ and feed them to TF. ① The channel measurements values are normalized such that the largest magnitude of each channel measurement corresponds to 1. We consider two variants of the input: First the complex channel measurement that includes information about the magnitude and phase of the CIR signal, second the PDP which drops the signal phase information to halve the input size. ② We interpret the real and imaginary part of the complex channel measurement as two channels $[Re(\boldsymbol{r}), Im(\boldsymbol{r})]$ and the PDP as a single channel $|\boldsymbol{r}|^2$. ③ Subsequently, we apply a convolutional layer with kernel size 1×1 (denoted in Fig. 1 as Conv). This maps the low dimensional input to a high dimensional latent-space with d_m dimensions. ④ Similarly, the Decoder input is mapped to a latent-space with a dimensionality of d_m. Note that the multi-head attention layer splits the input vectors into multiple slices along the embedding dimensions. Given the high dimensional embedding, we apply a positional encoding [21]. This induces a notion of sequence to the data that the TF can not inherently express.

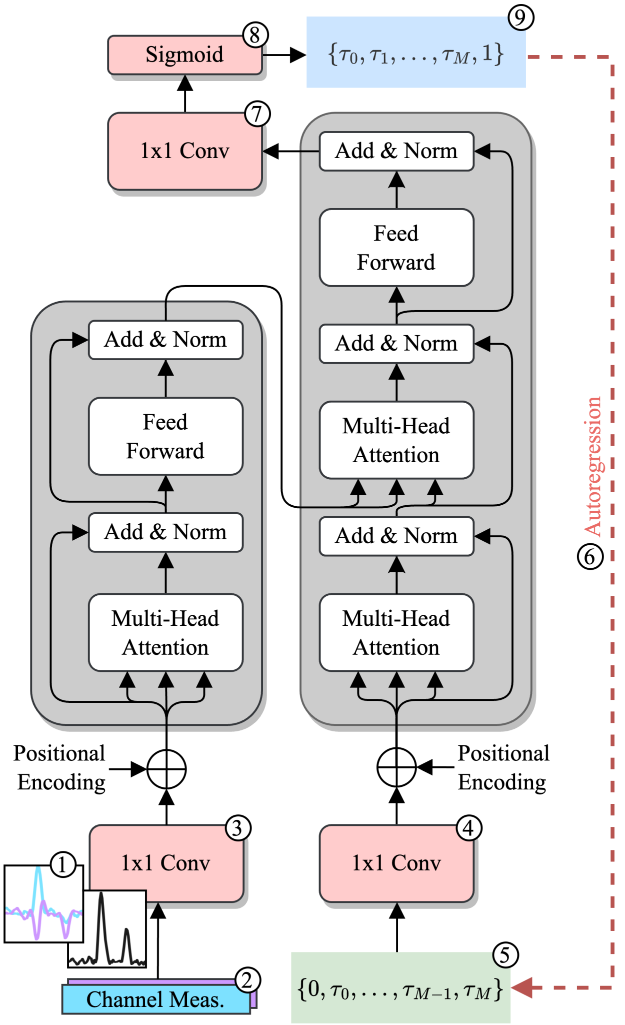

Fig. 1. Overview of the proposed method. Grey blocks denote the original TF [21]. Red blocks mark our adaptations. We visualize the channel measurement by its real (cyan) and imaginary parts (purple) and PDP thereof (black). We show the Encoder input (green) and corresponding references (blue). The red dotted arrow indicates the autoregressive property of the Decoder during inference.

Although working in a sequential manner when generating estimates, TF is trained in parallel. Thus, we provide all MPC delays to the Decoder at once ⑤ and apply a mask to the attention scores to prevent invalid information flow. ⑥ During inference, we feed back the generated estimates to the Decoder in an autoregressive way. ⑦ The Decoder output has a dimension of d_m. To obtain a 1D sequence of estimates, we apply a 1×1 convolutional layer to map the representation back to the ”MPC delay space”. ⑧ Note that we normalize the delay estimates w.r.t. the delay window. Hence, a sigmoid function constrains the value range between 0 and 1. Consequently, we scale the values by $T_r \cdot t_s$, where t_s is the sampling period, to obtain the actual delays.

C. Training and Inference

To supervise the training, we employ a sequence that includes M normalized MPC real delays as Decoder input ⑤ and references ⑨. The Decoder input in ⑤ is augmented with

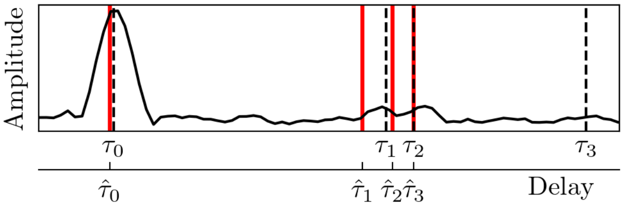

Fig. 2. Normalized magnitude of an input signal $\boldsymbol{r(t)}$. The signal covers a direct path and 3 specular reflections. Dashed lines show true MPC delays $\boldsymbol{\tau_m}$. Red lines show estimated MPC delays $\boldsymbol{\hat{\tau}_\mu}$.

a leading 0 and the labels with a trailing 1 in ⑨ to indicate the start and end points, see Fig. 1. The attention layers of the Decoder in ④ to ⑦ use masking to prevent information flow that contradicts the autoregressive property [21].

For inference, we initially feed a single 0 to the Decoder that indicates the start of the sequence. Thereupon, the M' delay estimates are created one after the other, in ascending order of the delay, and include all previous estimates. Therefore, TF generates the next estimate based on the information from the previous ones, thus it learns dependencies between MPCs, especially in environments with spatially coherent channel measurements. When an estimate is close to 1 (the Sigmoid function never reaches 1) we stop the decoding procedure. The first value above a pre-examined constant threshold of 0.95 excludes estimates in the last $5\ \%$ of the time window. As most different MPCs arrive early (below the threshold), no important information is lost. Note that this threshold is not a hyperparameter but a pre-examined implementation detail. Vaswani et al. [21] claimed that the results of a TF improve as the number of trainable parameters increases. Therefore, we maximize the number of trainable parameters so that TF can be trained in a reasonable time with limited computational resources (see Sec. IV-D for details). All experiments use the hyperparameters: $N_b = 4$, $N_h = 4$, $d_m = 64$, and $d_{ff} = 128$, and learning rate of the training process $= 0.01$.

D. MPC Extraction and Mapping

In general, to determine whether an MPC is classified as detected, we fit M' MPC delay estimates to M reference MPC delays using the hungarian method [38]. We then apply a maximum allowable error threshold to MPC pairs (estimate and reference) that are too far apart, to remove them from the dataset. Fig. 2 shows correctly detected true positives (TPs) multipath components (MPCs) $(\tau_0, \hat{\tau}_0)$, $(\tau_1, \hat{\tau}_2)$, $(\tau_2, \hat{\tau}_3)$. $\hat{\tau}_1$ is a false positive (FP) and τ_3 is a false negative (FN).

V. Experimental Setup

This Section installs benchmark datasets (Sec. V-A), baseline methods (Sec. V-B), and evaluation metrics (Sec. V-C).

A. Datasets

Our evaluations employ two synthetic datasets that mimic 5G and UWB systems and a real-world dataset of an UWB system. Note that we explicitly employ synthetic datasets to control effects in data and to apply the TF architecture in the first place, as TF is known to only work when there is a lot of training data, which is difficult to acquire in the real world [39]. Sec. V-A.1 describes our synthetic 5G dataset that mimics an empty hall with fixed objects and includes stochastic alterations. Sec. V-A.1 describes our synthetic UWB dataset that mimics a random environment with varying noise characteristics and signal obstructions. And Sec. V-A.1 describes our real-world dataset of a static environment.

1) Synthetic 5G dataset: The key idea of this large synthetic dataset is to investigate the ability of MPC estimators to exploit characteristic patterns of dynamic environments. To do this, we generated the dataset with our simulation setup, that is based on the stochastic channel model QuaDRiGa [40]. We model the channel as a set of reflectors and clusters. A single reflection point describes the reflectors. Clusters combine multiple subpaths caused by reflections in close proximity. These paths overlap in the delay domain as they arrive at the receiver in a short temporal span. The simulation distinguishes between random cluster (RC) and complementary semi-deterministic cluster (SDC) [41]. We determine RC locations purely stochastic. We use SDCs to model reflections at fixed positions. The data represent a dynamic environment and include characteristic static signal interactions such as wall reflections. The setup mimics an empty hall of size $45\ \mathrm{m} \times 30\ \mathrm{m}$. All channel measurements contain a direct path MPC, i.e., all signals have line-of-sight (LOS). We mounted six evenly spaced receivers on the walls of the hall. They provide six channel measurements for each signal burst. The recorded CIRs are aligned with the shortest delay of each burst. As in real-world deployments, the true time of flight (TOF) is unavailable. Note that we recorded the signals simultaneously at the receivers, but we employ them as independent channel measurements. The dataset incorporates 7 SDCs that represent the walls, floor, ceiling, and a rack as well as 23 RCs. We generate $958{,}320$ channel measurements each sampled at $122.88\ \mathrm{MHz}$ with a delay window of $1.4\ \mathrm{\mu s}$ that results in a length of 167 samples. The signal is generated with a carrier frequency of $4\ \mathrm{GHz}$ and $100\ \mathrm{MHz}$ bandwidth. We split this dataset explicitly into $70\ \%$ training, $10\ \%$ validation, and $20\ \%$ test data. This split ensures generalizability as the subsets cover different independent channel measurements, e.g., different people walking on different trajectories.

2) Synthetic UWB dataset: The key idea of this large synthetic dataset is to distinguish between deterministic and diffuse signal components, see Eq. 2. To achieve this, we employed Kram *et al.* [19] simulation setup, that generates realistic UWB data. In contrast to our environment-specific 5G dataset, the generated reflection points of the UWB data are spatially randomly distributed. Consequently, there is no spatial correlation between the data points. This experiment therefore provides information about the generalizability of the methods. The setup mimics an industrial environment of size $15\ \mathrm{m} \times 15\ \mathrm{m}$. We placed the transmitter, receiver, and all reflection points in it. To mimic a typical complicated industrial environment, only $50\ \%$ of the signals are LOS and the remaining signals are NLOS. And, to cover real-world signal characteristics, we parameterized the signal-to-noise ratio ($SNR \in \{20, 30, 40, 50\}$ dB) of all signals, i.e., the power ratio of uniformly distributed non-spatial noise, and the signal-to-interference ratio ($SIR \in \{20, 30, 40, 50\}$ dB), i.e., the performance ratio between deterministic and diffuse signal components. We generated $1,000,000$ channel measurements, each with $T_r = 128$ samples at a sampling rate of $f_s = 1$ GHz. We generated the signals with a bandwidth of 500 MHz and a carrier frequency of 4 GHz. We split this dataset in the same way as the 5G dataset to ensure generalizability.

3) Real-World UWB dataset: The key idea of this small real-world dataset is to explore the exploitability of environment-specific features and to verify the performance of the methods on real data. In general, complex (NLOS) real-world propagation scenarios consist of many untraceable signal paths. Obtaining reference MPCs is of course impossible there. Therefore, we selected the public deterministic real-world dataset from Kulmer *et al.* [42]. The scenario includes a single receiver and single transmitter, which are synchronized. The dataset contains UWB channel measurements, corresponding reference positions, and a map of the environment. The channel measurements cover an area of $1.25 \text{ m} \times 1.7 \text{ m}$ in the center of a furnished laboratory room. The path delays are less variable than those of the synthetic datasets. Hence it can be considered as less complex. We considered specular reflections on flat surfaces to be deterministic MPC, i.e., reflections on the walls of the room. Consequently, we treat all other signal components as diffuse MP signals. We manually annotate four MPC delays (i.e., wall reflections), through the geometrical relationships of the specular reflections. The dataset incorporates 420 channel measurements, each sampled at 6.95 GHz with a delay window of ≈ 216 ns, resulting in a length of 501 samples. The signal is generated with a carrier frequency of 4 GHz and a bandwidth of 416.7 MHz. We subsampled the CIRs by a factor of 3, resulting in CIRs with a sampling frequency of approximately $f_s = 2.3$ GHz and a delay window length of 167 samples. This ensures a fair comparison and reduces the computational effort. We explicitly split this dataset into 200 training and 50 validation and 170 test data points.

B. Baseline Methods

We compare our TF with two conventional MPC-delay estimators [17], [18] and the latest DL method [19].

1) MUSIC: We implemented the renowned subspace-based spectral-MUSIC algorithm [43]. It offers a pseudo-spectrum that exhibits peaks at contained signal components. To estimate the number of MPCs we apply the minimum description length (MDL) criterion. We employ a peak-finding algorithm to determine the final delay estimates. A threshold parameterizes the peak-finding process. We adapt the threshold experimentally for each dataset to obtain an optimal trade-off between sensitivity and precision.

2) Kulmer et al.: We also implemented the renowned iterative algorithm (search and subtract) of Kulmer *et al.* [18]. As MUSIC, it relies on the a priori estimated number of MPCs. We parameterized it manually to prevent errors caused by upstream source number estimators. In line with Kram *et al.* [19], we set a higher than necessary fixed number of estimates to increase the detection rate. As the number of MPCs differs significantly for each dataset, for best results we adjust it individually. Again, we adapt all parameters for each dataset experimentally.

3) U-Net: We also implemented the state-of-the-art DL algorithm of Kram *et al.* [19]. It exploits U-Net convolutional neural network (CNN) [31] to provide probabilities of the presence of MPCs per input sample. We applied a threshold to adjust a trade-off between detection sensitivity and precision. Again, we adapt it for each dataset experimentally.

C. Metrics

To comprehensively evaluate the performance of the MPC estimators, we employ the following three metrics:

1) Sensitivity and Precision: To evaluate the detection performance we define the sensitivity such that

$$\text{Sensitivity} = TP/(TP + FN) \tag{4}$$

and the precision such that

$$\text{Precision} = TP/(TP + FP). \tag{5}$$

Sensitivity describes how many reference MPCs were detected and precision describes the certainty of whether an estimate is assigned to a correct reference MPC.

2) Absolute Distance Error (ADE): To measure the estimation error between an estimated $\hat{\tau}_\mu$ and a reference delay τ_m, we use the absolute distance error (ADE) of Kram *et al.* [44]:

$$d_{\mu,m} = c|\hat{\tau}_\mu - \tau_m|. \tag{6}$$

We scale the temporal distance by the speed of light c to obtain distances in m. We set the error threshold of the upper limit to $ADE = 3 \text{ m}$ to correspond to an error of $\approx 10 \text{ ns}$ in the time domain. We use the ADE metric in two ways: First, we consider the average ADE of the assigned delay pairs per channel measurement, denoted as mean absolute distance error (MADE). Second, to benchmark the MPC estimation w.r.t. path delays, we consider ADE as a function of path delays.

VI. Evaluation

This Section discusses the results of all methods along the two synthetic datasets 5G (Sec. VI-A) and UWB (Sec. VI-B) as well as the real UWB dataset (Sec. VI-C) and the computational effort (Sec. VI-D). We employ the metrics MADE, sensitivity, precision, and the ADE distributions w.r.t. path delay. Table I summarizes all results. KU represents Kulmers' algorithm [18], TF processes complex channel measurements and TF-PDP processes PDP measurements. Note that we compare our dataset-specific findings in Sec. VII.

A. Results: Synthetic 5G Dataset

First, we discuss the overall performance of the methods w.r.t. their detection performance (see Fig. 3, left) and their estimation accuracy (see Fig. 4, left). Then, we discuss their performance w.r.t. path delays (see Fig. 5).

Fig. 3 shows that TF and TF-PDP result in a combination of high precision and sensitivity (> 70 %). TF significantly (> 20 %) outperforms the baselines KU and U-Net that yield a low precision and sensitivity (< 50 % on average). We assume that TF achieves high precision as it implicitly estimates the number of MPCs as it learns to estimate delays. Its autoregression ensures that the number of MPCs depends

TABLE I

RESULTS (BEST IN BOLD) PER METHOD AND TEST DATASET. FOR EACH, WE LIST THE MEAN AND 90TH PERCENTILE OF CUMULATIVE ERROR (P90) OF THE MADE METRIC (LEFT COLUMN, LOWER IS BETTER), AS WELL AS THE MEAN AND 10TH PERCENTILE (P10) OF SENSITIVITY (MIDDLE COLUMN) AND PRECISION DISTRIBUTIONS (RIGHT COLUMN, HIGHER IS BETTER).

	Synthetic 5G						**Synthetic UWB**						**Real-World UWB**					
	MADE [m]		Sensitivity [%]		Precision [%]		MADE [m]		Sensitivity [%]		Precision [%]		MADE [m]		Sensitivity [%]		Precision [%]	
Method	Mean	P90	Mean	P10	Mean	P10	Mean	P90	Mean	P10	Mean	P10	Mean	P90	Mean	P10	Mean	P10
MUSIC [17]	**0.73**	**1.12**	19.6	11.1	**78.2**	**57.1**	**0.51**	**1.36**	30.9	9.0	**72.0**	27.2	0.29	0.73	69.1	50.0	98.7	**100.0**
KU [18]	1.20	1.62	25.9	10.0	22.0	9.4	0.98	1.99	23.0	0.0	10.1	0.0	0.43	0.56	**100.0**	**100.0**	26.6	26.6
U-Net [19]	1.17	1.55	41.4	12.9	20.3	7.7	0.86	1.74	34.9	0.0	32.3	0.0	0.34	0.62	86.3	60.0	73.7	50.0
TF	0.94	1.21	**72.0**	**50.0**	73.2	51.9	0.75	1.33	**57.8**	**23.8**	58.5	28.5	0.11	0.22	**100.0**	**100.0**	**100.0**	**100.0**
TF-PDP	0.99	1.31	68.0	44.4	70.0	44.8	0.86	1.38	54.0	23.1	58.8	**30.0**	**0.07**	**0.13**	**100.0**	**100.0**	**100.0**	**100.0**

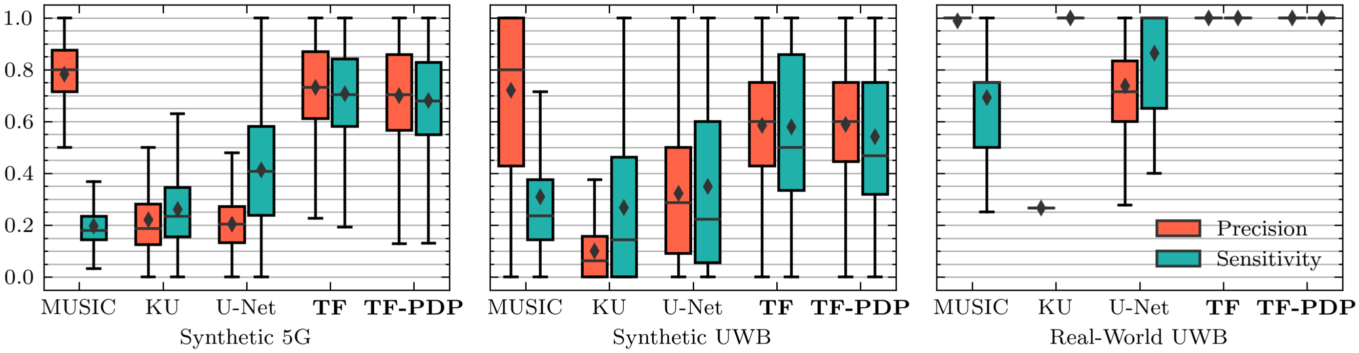

Fig. 3. MPC detection performance results: distribution of precision (orange) and sensitivity (teal) metrics of all methods on our datasets. A black diamond represents the mean and a black horizontal line represents the median. We do not show outliers.

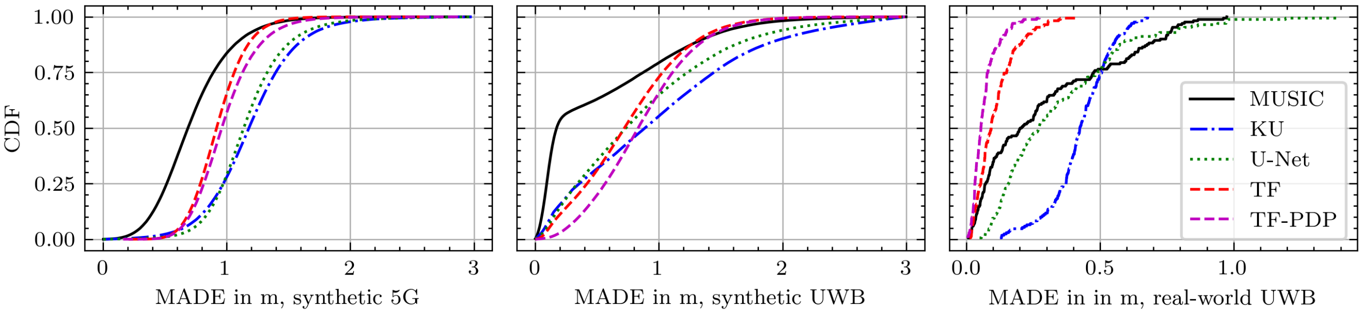

Fig. 4. MPC estimation performance results: Cumulative density function of the MADE per CIR w.r.t. the method and datasets.

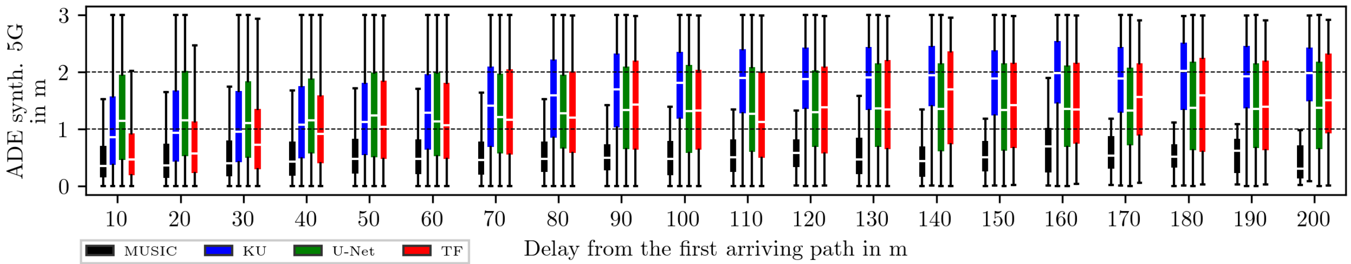

Fig. 5. Performance w.r.t. path delays: ADE distributions w.r.t. path delays for the synthetic 5G dataset. We bin the ADE values w.r.t. delays with a bin size of 10 m. We do not consider late delay bins > 200 m as the number of MPCs is not significant. The delays depend on the first arriving path of the respective burst (see Sec. V-A.1).

on the input signal and on previous delay estimates. This may result in few FPs and increases precision. In contrast to KU and U-Net, TF's attention mechanism captures dependencies across all input signals (dataset). In this way, TF captures global long-term dependency patterns characteristic of each environment, thus providing more accurate and reliable MPC estimates. MUSIC is an exception. Compared to all other methods, MUSIC achieves the highest accuracy (78 % on average), but also the lowest sensitivity (20 % on average). Hence, MUSIC does not identify (map) the correct MPCs in most (80 %) cases. MUSIC therefore provides significantly less spatial information (MPCs) and therefore provides less accurate positions than all other methods.

Fig. 4 (left) supports these findings. MUSIC (black line), TF and TF-PDP (dashed lines) provide the most accurate delays and outperform KU and U-Net (blue and green lines). However, MUSIC, KU and U-Net do not recognize most MPCs, so their sensitivity (blue-green boxplots in Fig. 3) is significantly lower than that of TF and TF-PDP. We think that TF offers significantly higher sensitivity as it takes into account long time and environmental dependencies when all other methods only consider local signal patterns. So TF recognizes global patterns in the input signal as it captures signal patterns of the environment (of a dataset).

Let us now discuss the results of the ADE distributions w.r.t. path delays in a signal. Fig. 5 shows the estimation accuracy (of all methods) of the delays along a signal. For simplicity, we only show results of TF (that are on par with TF-PDP). As the dataset does not provide true TOF, we normalize the delays to the first incoming path per signal / position. KU, U-Net, and TF show a significant increase in ADE with late arriving MPCs (compare the delays at 10 m and 200 m). KU, U-Net, and TF achieve inaccurate results for late MPCs. Instead, our TF estimates early MPCs (see delay at 10 m and 20 m) much more accurately (95% of ADE below 2 m at a distance of less than 10 m). MUSIC is again an exception. MUSIC (black boxplot) achieves the lowest ADE (below 1 m in most cases) over the entire distance (see delay at 10 m and 200 m). However, its sensitivity is the lowest (blue-green boxplots in Fig. 3), so it misses most MPCs, thus loses valuable spatial information and drops positioning accuracy drastically.

B. Results: Synthethic UWB dataset

Again, we first discuss the overall performance of the methods w.r.t. their detection performance (see Fig. 3, middle) and their estimation accuracy (see Fig. 4, middle). Then, we discuss their performance w.r.t. path delays (see Fig. 6).

The overall performance of all methods along all metrics on the synthetic UWB dataset is similar to that on the synthetic 5G dataset. However, Fig. 3 clearly shows that in general the variance of all methods along both metrics is higher than for the 5G dataset. We think this is due to the high variance of the SNR and SIR as well as the 50 % NLOS data points (see Sec. V-A.2). MPC estimation on UWB data is therefore significantly more challenging. Again, MUSIC is an exception as it returns the highest precision (72 %) but the lowest sensitivity (21 %). Instead, TF and TF-PDP result in slightly lower but balanced precision and sensitivity (average of 58 %), and guarantee high detection reliability and accuracy. Interestingly, TF-PDP provides higher precision (30 %) than MUSIC (27 %) for the 90 % percentile. As the results of KU and U-Net on the 5G dataset, they again perform worst.

Fig. 4 shows interesting results. The estimation accuracy of MUSIC lowers significantly from the 50 % percentile onwards (MADE > 0.2 m). We think this is because MUSIC accurately estimates MPCs on the 50 % LOS signals of the dataset, but on the remaining signals with indirect paths, MUSIC fails due to its low sensitivity or low ADE. In contrast, all other methods return larger but consistent errors regardless of the signals. Again, TF and TFT-PDP estimate MPCs more accurately than KU and U-Net. This time MUSIC yields the highest precision (82 %) but the MPC estimation accuracy drops drastically at over 55 % of the data points and at over 90 % of the data points our TF and TF-PDP outperform the rest (MADE < 1.33 m). TF-PDP is therefore over 41 cm more accurate than the next best, U-Net (MADE > 1.74 m).

Fig. 6 shows the ADE distributions of the path delay in a signal. This time the path delay in a signal represents the true TOF. Interestingly, this time the ADE of all methods increase with later arriving MPCs. TF achieves exceptionally low ADE with over 75 % of values below 1 m in the early time window up to a path delay of 10 m, which are comparable to MUSIC for TOF < 7 m. In contrast to the ADE distributions on the 5G dataset, the median values (white lines in Fig. 6) on the UWB dataset decrease significantly, especially for TOF < 12 m. ADE of all methods for TOF > 12 m increases, with MUSIC being particularly noticeable, whose ADE increases dramatically compared to the 5G dataset. The ADEs of TF and TF-PDP remain comparably low, even at very late path delays. We think that this is because TF regresses the delays directly, so that the accuracy of the estimate is not (categorically) limited and can be (numerically) higher, whereas all other methods can only consider MPCs in local, short time intervals and are therefore less accurate. The MUSIC algorithm yields the highest accuracy, but is again severely limited in its sensitivity. For signal path lengths below 12 m, TF and TF-PDP yield similar results to MUSIC and outperform KU and U-Net. While KU's ADE increases significantly, the ADEs of DL-based approaches TF, TF-PDP and U-Net become similar. MUSIC consistently results in the lowest median values $<$ 0.5 m) up to TOF < 22 m and also increases significantly beyond that. U-Net provides lower ADE than TF and TF-PDP for path lengths larger than 12 m, but the precision of MUSIC and U-Net is lower, rendering them impractical for positioning. Even on this dataset with 50 % N/LOS, TF and TF-PDP perform significantly better than all other methods.

C. Results: Real-World UWB dataset

Again, we first discuss the overall performance of the methods w.r.t. their detection performance (see Fig. 3, right) and their estimation accuracy (see Fig. 4, right). Then, we discuss their performance w.r.t. path delays (see Fig. 7). Fig. 8 shows an exemplary real channel measurement, annotated with path delays, that we also see in Fig. 7. It visualizes two MPCs that arrive close behind the LOS component at 5 m and significantly interfere with each other. Two more MPCs arrive later at 10 m and are more clearly separated from each other. However, the signal disrupts the diffuse MP transmission and causes destructive interference. So, no peaks are visible. Fig. 8 helps to reason why TF and TF-PDP perform best.

Fig. 3 shows an exceptional success rate (100 %) of TF and TF-PDP in detecting (precision and sensitivity) MPCs. We think that TF and TF-PDP perform best because the real data combines all possible effects that can only be exploited by TF's attention mechanism, resulting in higher success rates. Similar to the other datasets, MUSIC also yields high precision (100 %) but lower significantly average sensitivity (69 %). Interestingly, this time MUSIC provides significantly higher sensitivities on this real dataset than the other two. We believe this is because the MPCs in this dataset are clearly separable and recognizable. Unexpectedly, KU and U-Net also show high sensitivity (up to 100 %) but significantly lower precision (up to 27 %) as they may incorrectly estimate too many MPCs.

Even the MPC estimation performance (MADE) is best compared to the other datasets for all methods, see Fig. 4. This time the MPC estimation performance (MADE) of TF and

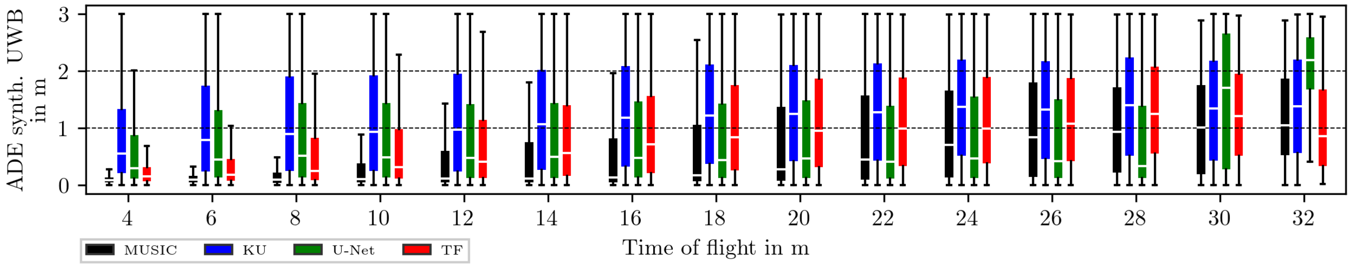

Fig. 6. Performance w.r.t. path delays: ADE distributions w.r.t. path delays for the synthetic UWB dataset. We bin the ADE values w.r.t. delays with a bin size of 2 m. We do not consider late delay bins > 32 m as the number of MPCs is not significant. The delays correspond to the true TOF.

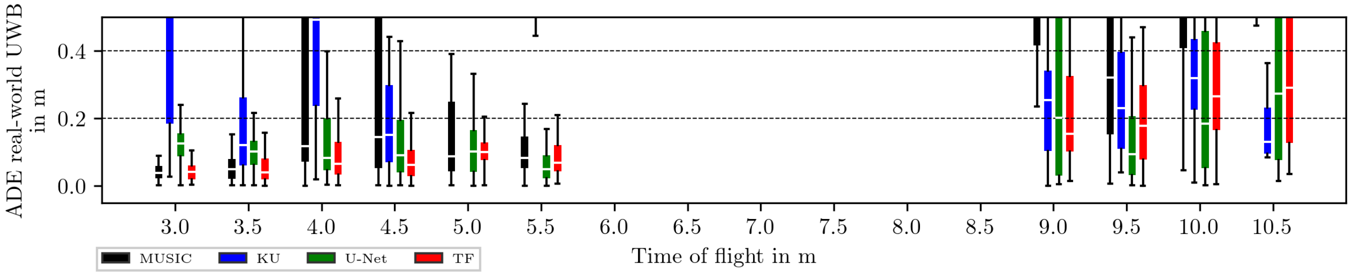

Fig. 7. Performance w.r.t. path delays: ADE distributions w.r.t. path delays for the real-world UWB dataset. We bin the ADE values w.r.t. delays with a bin size of 0.5 m. For readability, we crop high ADE values from the image. The delays correspond to the true TOF.

TF-PDP is in line with their detection performance (100 %). For all methods, they provide the lowest MADE (7 cm and 11 cm) even for the 90 % percentile. Interestingly, MUSIC falls significantly behind this time. It yields similar results as U-Net up to the 75 % percentile (MADE < 0.5 m). But KU and U-Net (MADE < 0.6 m) outperform MUSIC (MADE > 0.8 m) beyond the 80 % percentile. We believe this is because, unlike on simulated data on real data, MUSIC's assumptions are too limiting and lower its performance.

The results of the ADE distributions of the path delay for the real UWB dataset are particularly remarkable, see Fig. 7. Compared to the synthetic UWB dataset (ADE > 0.5 m), all methods result in much lower ADE for early (ADE < 0.25 m) and late (ADE < 0.45 m) MPCs on real data. We attribute this to the simplicity of the environment (and high sampling rate). TF, TF-PDP, and U-Net provide the smallest errors with the lowest variance (ADE < 0.2m) for early MPCs (TOF < 6 m) immediately after the LOS peak (< 0.2 m). As with the synthetic 5G and UWB datasets, we assume that TF estimates exceptionally accurate early (overlapping) MPCs due to the unique combination of global signal patterns in the dataset and the direct regression of the delays. Instead, MUSIC and KU yield significantly worse results with high variance at TOF < 6m. We think this is because they cannot estimate the immediate overlapping MPCs after the LOS peak. Note that due to the propagation environment, we do not measure MPCs in the range 6 m $<$ TOF $<$ 8.5 m, so there is a gap. Interestingly, this time for all methods MUSIC yields significantly worse results (ADE > 0.5m) for late delay estimates (TOF $>$ 9 m). Here, TF, KU, and U-Net provide much more consistent estimates (ADE < 0.4m). We think that this increase is caused by the attenuation of the signal paths due to the free space path loss that makes them less robust against diffuse MP.

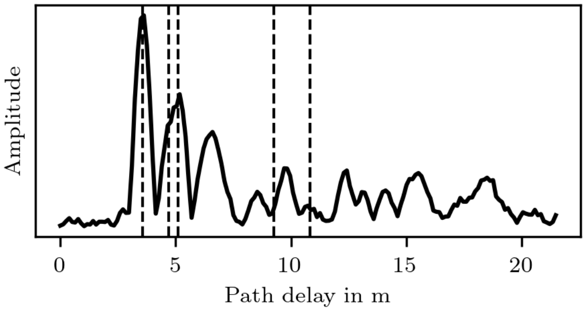

Fig. 8. Example channel measurement of the real UWB dataset. Black solid line shows the bandwidth-limited CIR. Vertical dashed lines show the MPC delays.

D. Training and Inference time

In this ablation study, we examine the inference and training times of all methods. As localization is done in real time, fast inference of the MPC estimators is crucial. And to conserve computing resources, short/low training effort is crucial.

For a fair **inference time** comparison, we run each method on a single AMD Ryzen 9 7900X CPU at 3.0 GHz and each input time series consists of 176 samples. Fig. 9 shows the average inference time of each method versus the number of estimates. TF and KU generate the estimates iteratively. Therefore, the inference time increases with the number of estimates. The inference times of U-Net and MUSIC are independent of the number of estimates. KU is slowest, e.g., on average 9 times slower than TF. U-Net offers the fastest inference of all methods (1 ms). Instead, MUSIC is significantly slower than TF (up to 25 estimates). We think that the PDP data in memory management (mini-batches) is first converted into a high-dimensional embedding (similar to the CIRs) and thus requires comparable computing capacities (see Sec. IV-B). Fig. 9 shows an average inference time of TF below 10 ms for 10 or fewer estimates. So in an environment not exceeding 10 MPCs, TF processes channel measurements at 100 Hz. Although attention

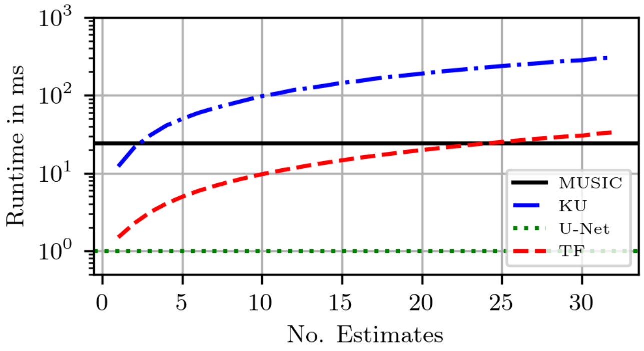

Fig. 9. Average inference time versus number of delay estimates. Note that MUSIC and U-Net do not depend on the number of estimates.

increases quadratically with longer input sequences, inference time increases linearly with more estimates. This is because the length of the output sequence is comparatively short. We assume that the inference time of the TF can be reduced if explicitly optimized for this.

We cannot directly compare the **training effort** of all methods, as we had to train them on different hardware platforms. We trained MUSIC and KU on the same AMD Ryzen 9 7900X CPU at 3.0 GHz, U-Net on a GTX 3070 graphics card (20 TeraFLOPS), and TF and TF-PDP on 4 Tesla V100 graphics cards (989 TeraFLOPS). MUSIC and KU achieve optimal MPC estimates after 49.5 and 31.3 hours. U-Net reaches its best performance after a few epochs ($<$ 1 hour). TF achieves high performance after 5,000 epochs, that continuously improves as training progresses. Due to time and effort constraints, we stopped its training after 23,000 epochs (27 days). In essence, DL-based methods require most time to train but provide fastest inference.

VII. Discussion

This Section discusses our findings along input variants, sensitivity, generalization, and environment-specific features.

Our experiments on different input data (CIR and PDP) show no significant performance difference between TF on CIR and TF on PDP input data on all three datasets (sensitivity: -4 %, precision: -3 %, MADE: -5 cm). This clarifies that PDP signals contain similarly valuable information as CIR signals. However, PDP signals halve the data management effort and are therefore the first choice.

All methods accurately estimate delays of the MPCs at high signal power. But delays of late arriving, attenuated or overlapping MPCs are often estimated inaccurately. Ignoring them, leads to high precision and low MADE, but at the same time low sensitivity. In consequence, essential spatial information is lost and the positioning accuracy worsens. The best example of this is MUSIC. It estimates MPCs such as the direct path very accurately, but fails to detect most (remaining) MPCs in a channel measurement. Instead, our TF detects most MPCs (synthetic 5G: 72.0 % and UWB: 57.8 %, real-world UWB: 100 %) with delay accuracy as MUSIC.

Our experiments show that TF extracts delays of MPCs accurately and robustly. However, DL-based approaches such as TF and U-Net are not limited to this task. For example, TF could estimate additional parameters such as AoA in multiple input multiple output (MIMO) setups.

Our experiments show that TF generalize to unknown environments and exploits environment-specific features best. The synthetic UWB dataset does not represent a specific environment. Each data point is assigned a stochastic (unknown) environmental interaction. Therefore, the real-world UWB dataset only contains information that is independent of the environment. Here, although TF cannot exploit spatial and temporal (environmental) dependencies in the entire signal, TF still outperforms all other methods because, unlike all other methods, TF potentially captures the (physical) channel properties of the system. Based on this valuable additional knowledge about the system characteristics, TF estimates significantly more accurately and reliably than all other methods. As TF is the only method to capture dependencies across the entire input signal (and dataset), it also captures environment-specific patterns within it. Based on this valuable additional spatial and temporal knowledge, TF estimates significantly more accurately and reliably than all other methods. For example, the real UWB dataset represents a static environment. Alike, the synthetic 5G dataset represents static elements in a dynamic environment. TF yields the best results of all methods here as TF directly learns the hidden environment-specific patterns in the signal (dataset). It is possible that TF also learns additional environmentally independent information when it is trained on input signals from different environments. In essence, environment independence reduces deployment effort and exploiting environment-specific patterns improves accuracy and reliability. This renders TF a realistic and practical method beyond the state of the art.

VIII. Conclusion

In this article, we showed that our MPC delay estimation based on a TF neural network achieves more accurate, precise, and sensitive results than the state-of-the-art. We demonstrated this for three radio systems with different bandwidth limitations and sampling rates. Key features of our method are the implicit estimation of the number of MPCs and the ability to resolve dependencies throughout the input signal (and beyond). The latter provide TF with additional spatial and temporal information about the environment that is hidden from other methods, and result in outstanding positioning performance. Our experiments show that the spatially consistent channels of environments with certain characteristic interactions, e.g., wall reflections, improve the overall performance of TF. As TF (regresses) estimates the MPC delays directly and attends all signal characteristics, it can achieve sub-sample accuracy. Our experiments also show that TF estimates early arriving MPCs with exceptional accuracy and particularly low uncertainty. Although our TF's inference time increases with more estimates, it is shorter than that of most other methods. In addition, less complex PDP input data does not affect TF, as TF still extracts important information from the signal. However, PDP significantly reduces the effort involved in data processing and storage as it halves the input. In future work, we will increase the amount of environmental information extracted from the channel and comprehensively evaluate TF in real-world scenarios to better understand its limitations.

IX. Acknowledgement

We thank Ernst Eberlein for his help with data simulation.

[1] A. F. G. G. Ferreira, D. M. A. Fernandes, A. P. Catarino, and J. L. Monteiro, "Localization and positioning systems for emergency responders: A survey," *IEEE Comms. Surveys & Tutorials*, vol. 19, no. 4, pp. 2836–2870, 2017.

[2] K. Witrisal, P. Meissner, E. Leitinger, Y. Shen, C. Gustafson, F. Tufvesson, K. Haneda, D. Dardari, A. F. Molisch, A. Conti *et al.*, "High-accuracy localization for assisted living: 5g systems will turn multipath channels from foe to friend," *IEEE Signal Processing Magazine*, vol. 33, no. 2, pp. 59–70, 2016.

[3] M. Elsanhoury, P. Mäkelä, J. Koljonen, P. Välisuo, A. Shamsuzzoha, T. Mantere, M. Elmusrati, and H. Kuusniemi, "Precision positioning for smart logistics using ultra-wideband technology-based indoor navigation: A review," *IEEE Access*, vol. 10, pp. 44 413–44 445, 2022.

[4] T. Gigl, G. J. Janssen, V. Dizdarevic, K. Witrisal, and Z. Irahhauten, "Analysis of a uwb indoor positioning system based on received signal strength," in *4th Workshop on Positioning, Navigation and Communication*. IEEE, 2007, pp. 97–101.

[5] D. Dardari, A. Conti, U. Ferner, A. Giorgetti, and M. Z. Win, "Ranging with ultrawide bandwidth signals in multipath environments," *Proc. of the IEEE*, vol. 97, no. 2, pp. 404–426, 2009.

[6] D. Dardari, P. Closas, and P. M. Djurić, "Indoor tracking: Theory, methods, and technologies," *IEEE Trans. on Vehicular Technology*, vol. 64, no. 4, pp. 1263–1278, 2015.

[7] J. Xiong and K. Jamieson, "Arraytrack: A fine-grained indoor location system." Usenix, 2013.

[8] S. Aditya, A. F. Molisch, and H. M. Behairy, "A survey on the impact of multipath on wideband time-of-arrival based localization," *Proc. of the IEEE*, vol. 106, no. 7, pp. 1183–1203, 2018.

[9] P. Setlur, G. E. Smith, F. Ahmad, and M. G. Amin, "Target localization with a single sensor via multipath exploitation," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 1996–2014, 2012.

[10] C. Gentner, T. Jost, W. Wang, S. Zhang, A. Dammann, and U.-C. Fiebig, "Multipath assisted positioning with simultaneous localization and mapping," *IEEE Trans. on Wireless Comms.*, vol. 15, no. 9, pp. 6104–6117, 2016.

[11] E. Leitinger, F. Meyer, F. Hlawatsch, K. Witrisal, F. Tufvesson, and M. Z. Win, "A belief propagation algorithm for multipath-based slam," *IEEE Trans. on wireless Comms.*, vol. 18, no. 12, pp. 5613–5629, 2019.

[12] A. F. Molisch, "Ultra-wide-band propagation channels," *Proc. of the IEEE*, vol. 97, no. 2, pp. 353–371, 2009.

[13] J. A. Fessler and A. O. Hero, "Space-alternating generalized expectation-maximization algorithm," *IEEE Trans. on signal processing*, vol. 42, no. 10, pp. 2664–2677, 1994.

[14] M. Feder and E. Weinstein, "Parameter estimation of superimposed signals using the em algorithm," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 36, no. 4, pp. 477–489, 1988.

[15] A. Paulraj, R. Roy, and T. Kailath, "Estimation of signal parameters via rotational invariance techniques- esprit," in *Nineteeth Asilomar Conf. on Circuits, Systems and Computers, 1985.*, 1985, pp. 83–89.

[16] A. Richter, "Estimation of radio channel parameters: Models and algorithms." ISLE Blacksburg, VA, USA, 2005.

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

[18] J. Kulmer, S. Grebien, E. Leitinger, and K. Witrisal, "Delay estimation in presence of dense multipath," *IEEE wireless Comms. letters*, vol. 8, no. 5, pp. 1481–1484, 2019.

[19] S. Kram, C. Kraus, M. Stahlke, T. Feigl, J. Thielecke, and C. Mutschler, "Delay estimation in dense multipath environments using time series segmentation," in *2022 IEEE Wireless Comms. and Networking Conf. (WCNC)*. IEEE, 2022, pp. 1671–1676.

[20] J. Ott, M. Stahlke, S. Kram, T. Feigl, and C. Mutschler, "Multipath delay estimation in complex environments using transformer," in *2023 13th Intl. Conf. on Indoor Positioning and Indoor Navigation (IPIN)*, 2023, pp. 1–6.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[22] Y. Shen and M. Z. Win, "On the use of multipath geometry for wideband cooperative localization," in *IEEE Global Telecommunications Conf. (GLOBECOM)*. IEEE, 2009, pp. 1–6.

[23] C. Studer, S. Medjkouh, E. Gonultaş, T. Goldstein, and O. Tirkkonen, "Channel charting: Locating users within the radio environment using channel state information," *IEEE Access*, vol. 6, pp. 47 682–47 698, 2018.

[24] J. Deng, S. Medjkouh, N. Malm, O. Tirkkonen, and C. Studer, "Multipoint channel charting for wireless networks," in *2018 52nd Asilomar Conf. on Signals, Systems, and Computers*. IEEE, 2018, pp. 286–290.

[25] M. Stahlke, G. Yammine, T. Feigl, B. M. Eskofier, and C. Mutschler, "Indoor localization with robust global channel charting: A time-distance-based approach," *IEEE Trans. on Machine Learning in Comms. and Networking*, 2023.

[26] P. Stephan, F. Euchner, and S. t. Brink, "Angle-delay profile-based and timestamp-aided dissimilarity metrics for channel charting," *arXiv:2308.09539*, 2023.

[27] H. Akaike, "A new look at the statistical model identification," *IEEE transactions on automatic control*, vol. 19, no. 6, pp. 716–723, 1974.

[28] G. Schwarz, "Estimating the dimension of a model," *The annals of statistics*, pp. 461–464, 1978.

[29] D. Spielman, A. Paulraj, and T. Kailath, "Performance analysis of the music algorithm," in *ICASSP'86. IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing*, vol. 11. IEEE, 1986, pp. 1909–1912.

[30] Y. Yang, F. Gao, C. Qian, and G. Liao, "Model-aided deep neural network for source number detection," *IEEE Signal Processing Letters*, vol. 27, pp. 91–95, 2019.

[31] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer Intl. Publishing, 2015, pp. 234–241.

[32] Z. Wang and L. Wu, "Theoretical analysis of the inductive biases in deep convolutional networks," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[33] E. Leitinger, P. Meissner, C. Rüdisser, G. Dumphart, and K. Witrisal, "Evaluation of position-related information in multipath components for indoor positioning," *IEEE J. on Selected Areas in communications*, vol. 33, no. 11, pp. 2313–2328, 2015.

[34] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv:1409.0473*, 2014.

[35] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018.

[36] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," 2018.

[37] D. Hendrycks and K. Gimpel, "Gaussian error linear units (gelus)," *arXiv:1606.08415*, 2016.

[38] J. Munkres, "Algorithms for the assignment and transportation problems," *J. of the society for industrial and applied mathematics*, vol. 5, no. 1, pp. 32–38, 1957.

[39] N. Park and S. Kim, "How do vision transformers work?" *arXiv:2202.06709*, 2022.

[40] S. Jaeckel, "Quasi-deterministic channel modeling and experimental validation in cooperative and massive mimo deployment topologies," Ph.D. dissertation, Dissertation, Ilmenau, TU Ilmenau, 2017, 2017.

[41] M. Alawieh, E. Eberlein, S. Jäckel, N. Franke, B. Ghimire, T. Feigl, G. Yammine, and C. Mutschler, "Complementary semi-deterministic clusters for realistic statistical channel models for positioning," *arXiv:2207.07837*, 2022.

[42] J. Kulmer, S. Hinteregger, B. Grosswindhager, M. Rath, M. S. Bakr, E. Leitinger, and K. Witrisal, "Using decawave uwb transceivers for high-accuracy multipath-assisted indoor positioning," in *IEEE Intl. Conf. on Comms. Workshops (ICC Workshops)*. IEEE, 2017, pp. 1239–1245.

[43] X. Li and K. Pahlavan, "Super-resolution toa estimation with diversity for indoor geolocation," *IEEE Trans. on wireless Comms.*, vol. 3, no. 1, pp. 224–234, 2004.

[44] S. Kram, M. Stahlke, T. Feigl, J. Seitz, and J. Thielecke, "Uwb channel impulse responses for positioning in complex environments: A detailed feature analysis," *Sensors*, vol. 19, no. 24, p. 5547, 2019.