Horizon Europe Framework Programme

Towards an AI-native, user-centric air interface for 6G networks

---

D2.2 Digital Twin Training Sandbox

Contractual Delivery Date:	30-06-2024
Actual Delivery Date:	12-07-2024
Editor: *(name, organization)*	Houssem Sifaou and Osvaldo Simeone, King's College London
Deliverable nature:	R
Dissemination level:	PU
Version:	1.0
Keywords: Digital Twin, Bayesian Learning, Ray Tracing, Phase Errors, Calibration, Prediction-Powered Inference, Semi-Supervised Learning, Channel Knowledge Map	

ABSTRACT

One of the main objectives of WP2 is to develop reliable digital twin (DT) platforms for training and monitoring AI-AI methods. These platforms utilize virtual twins to simulate physical twins, enabling continuous cycles of simulation, prediction, analysis, and optimization. To ensure the reliability of DT systems, a Bayesian framework is proposed to manage model uncertainty arising from data limitations. This framework supports ensembling-based

This project has received funding from the European Union's Smart Networks and Services Joint Undertaking (SNS JU) under grant agreement No. 101096379.

methods for enhanced control and prediction. Moreover, a novel calibration scheme for ray tracing is introduced. This scheme employs a variational expectation maximization algorithm to correct phase errors, significantly improving prediction accuracy for tasks such as beamforming and user positioning. Additionally, a DT-aided semi-supervised learning approach is proposed. This method enhances AI model training by leveraging synthetic labels and mitigating biases through a tuned cross-prediction-powered inference scheme. These solutions enhance the management and optimization of AI models within DT platforms, ensuring their efficacy and reliability.

Disclaimer

This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101096379. This publication reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains.

Impressum

Full project title: Towards an AI-native, user-centric air interface for 6G networks

Short project title: CENTRIC

Number and title of the work package: WP2: AI-AI Hardware Platforms and Enablers

Number and title of task: T2.3: Training and Monitoring Environments

Document title: Digital Twin Training Sandbox

Editor: Osvaldo Simeone, King's College London

Work-package leader: Osvaldo Simeone, King's College London

Executive summary

This document aims to provide an overview of the developed algorithms within T2.3 of WP2, aiming at enhancing the reliability of digital twin (DT) platforms for training and monitoring AI-AI methods for 6G networks. The proposed solutions encompass a comprehensive approach: managing AI model uncertainty, calibrating data generated within the DT, and seamlessly integrating real data from the physical twin (PT) with synthetic data from the DT.

Bayesian Framework for DTs: a general Bayesian framework with the aim of quantifying and accounting for model uncertainty at the DT that is caused by limitations in the amount and quality of data available at the DT from the PT. In the proposed framework, the DT builds a Bayesian model of the communication system, which is leveraged to enable core DT functionalities such as control via multi-agent reinforcement learning (MARL), monitoring of the PT for anomaly detection, prediction, data-collection optimization, and counterfactual analysis. To exemplify the application of the proposed framework, a case-study system encompassing multiple sensing devices that report to a common receiver is investigated. Experimental results validate the effectiveness of the proposed Bayesian framework as compared to standard frequentist model-based solutions.

Calibrating Wireless Ray Tracing: a novel channel response-based scheme that estimates and compensates for the phase errors in the RT-generated channel responses. The proposed approach builds on the variational expectation maximization algorithm with a flexible choice of the prior phase-error distribution that bridges between a deterministic model with no phase errors and a stochastic model with uniform phase errors. The algorithm is computationally efficient, and is demonstrated, by leveraging the open-source differentiable RT software available within the Sionna library, to outperform existing methods in terms of the accuracy of RT predictions.

Reliable Semi-Supervised Learning via DTs: a novel semi-supervised learning scheme, referred to as tuned cross-prediction-powered inference (CPPI), is proposed. This scheme leverages synthetic labels generated by an ML model while accounting for inherent biases relative to true labels. Applied to beam alignment using channel knowledge maps in mmWave MIMO systems, simulation results demonstrate the superiority of this approach over conventional methods that rely solely on labeled data or standard pseudo-labeling strategies from semi-supervised learning.

List of authors

Company	Author	Contribution
KCL	Osvaldo Simeone	Editor
KCL	Houssem Sifaou	Contributor
KCL	Clement Ruah	Contributor
NVIDIA	Jakob Hoydis	Contributor

Table of Contents

List of figures

Abbreviations

AI	Artificial Intelligence
BS	Base Station
CKM	Channel Knowledge Map
COMA	Counterfactual Multi-Agent
CPPI	Cross-Prediction-Powered Inference
CSI	Channel State Information
DT	Digital Twin
ERM	Empirical Risk Minimization
FDTD	Finite-Difference Time-Domain
FPR	False Positive Rates
MARL	Multi-Agent Reinforcement Learning
MDP	Markov Decision Process
MIMO	Multiple-Input Multiple-Output
ML	Machine Learning
MSE	Mean Squared Error
PPI	Prediction-Powered Inference
PT	Physical Twin
RAN	Radio Access Network
ROC	Receiver Operating Characteristic
RT	Ray Tracing
SS	Semi-Supervised
TPR	True Positive Rates
UE	User Equipment
VEM	Variational Expectation Maximization

1 Introduction

The CENTRIC project is dedicated to pioneering a cutting-edge, AI-native air interface aimed at advancing sustainable solutions for 6G networks. A central objective of the project is to design and evaluate an advanced training and AI lifecycle monitoring environment utilizing Digital Twins (DTs). To achieve this goal, several critical challenges must be addressed, including managing AI model uncertainty arising from data limitations, calibrating synthetic data generated by the DT (such as wireless channels), and effectively leveraging synthetic data to improve AI model accuracy.

This report presents innovative solutions to these challenges. Specifically, it introduces a comprehensive Bayesian framework that enables the DT to quantify and manage model uncertainty due to data scarcity. Additionally, a novel calibration algorithm is developed to correct phase errors in ray tracing-generated channel responses, ensuring more accurate simulations of wireless environments. Furthermore, the report unveils a new semi-supervised learning scheme designed to optimally utilize synthetic labels while minimizing biases, thereby enhancing the precision of AI models.

1.1 A Bayesian Framework for Digital Twins

A key challenge in the deployment of DT systems is to ensure that virtual control optimization, monitoring, and analysis at the DT are safe and reliable, avoiding incorrect decisions caused by model exploitation [1]. To address this challenge, a general Bayesian framework is presented with the aim of quantifying and accounting for model uncertainty at the DT that is caused by limitations in the amount and quality of data available at the DT from the PT.

In the proposed framework, the DT builds a *Bayesian model* of the communication system dynamics based on data received from the PT. Unlike conventional *frequentist* parametric models, Bayesian models can quantify model uncertainty by maintaining a distribution over the model parameters [1], [2]. This enables ensembling-based control, prediction, and analysis methods, whereby policies, predictions, and recommendations are obtained by accounting for the agreements and disagreements among several models that are consistent with the available information.

1.2 Calibrating Wireless Ray Tracing for Digital Twins

DTs offer a natural framework for the deployment of software-based disaggregated wireless networks that follow the open radio access network (RAN) paradigm [3] [4]. Among the key advantages of DTs in this context is their capacity to synthesize data that can be leveraged to train, as well as to feed at run time, artificial intelligence (AI) models. Ray tracing (RT) is widely seen as an enabling technology for DTs of the RAN. By leveraging the predicted path components, a DT can simulate the propagation conditions for a given network deployment. The synthesized channels can then be used to train AI models that carry out tasks such as beamforming [4] user positioning [5], and channel charting [6].

In practical scenarios, it is not uncommon to observe small discrepancies, of the scale of a fraction of the carrier wavelength, between the ground-truth geometric properties and their

virtual counterpart. These discrepancies hinder the accuracy of the predicted phases of the simulated propagation paths, which are usually deemed to be unreliable [7]. Errors in the predicted phases can provide inaccurate interference patterns at the receiver when the temporal and/or spatial resolution of the system is not high enough to resolve each path individually. To solve this issue, a novel channel response-based calibration scheme for RT that estimates and mitigates per-path phase errors is proposed. Specifically, we introduce a novel phase error-aware calibration strategy for RT. The proposed approach builds on the variational expectation maximization (VEM) algorithm [2] with a tailored choice of the variational distribution on the phase errors. The algorithm is computationally efficient, and is demonstrated, by leveraging the open-source differentiable RT software available within the Sionna library, to outperform existing methods in terms of the accuracy of RT predictions.

1.3 Reliable Semi-Supervised Learning via Digital Twins

Next-generation wireless systems are expected to rely extensively on machine learning (ML) and data-driven decision-making [8]. Optimizing effective ML algorithms hinges on the availability of high-quality labeled data. However, obtaining labeled data is a challenging task in numerous wireless scenarios. Semi-supervised learning via pseudo-labeling provides a promising alternative by leveraging synthetic labels produced by ML models for unlabeled data [9]. However, predictions generated by ML models may be of insufficient quality. Therefore, making reliable use of synthetic labels requires an additional effort to reduce the bias caused by the discrepancy between synthetic and real labels.

Inspired by the recently developed prediction-powered inference (PPI) framework, we investigate how to leverage the synthetic labels produced by an ML model, while accounting for the inherent bias with respect to true labels. To this end, we first review PPI and its recent extensions, namely tuned PPI and cross-prediction-powered inference (CPPI). Then, we introduce a novel variant of PPI, referred to as tuned CPPI, that provides CPPI with an additional degree of freedom in adapting to the quality of the ML-based labels. We showcase an application of PPI-based techniques in wireless systems, namely beam alignment based on channel knowledge maps in millimeter-wave systems. Simulation results show the advantages of PPI-based techniques over conventional approaches that rely solely on labeled data or that apply standard pseudo-labeling strategies from semi-supervised learning. Furthermore, the proposed tuned CPPI method is observed to guarantee the best performance among all benchmark schemes.

2 A Bayesian Framework for Digital Twin

A critical obstacle in deploying DT systems is guaranteeing safe and dependable decision-making at the virtual control level. This is due to the potential for model exploitation arising from uncertainties within the data used by the DT [1]. To overcome this challenge, we introduce a novel framework based on Bayesian statistics. This framework quantifies and incorporates model uncertainties originating from data limitations within the DT, specifically data received from the PT (Figure 1).

The proposed framework equips the DT with the ability to construct a Bayesian model of the communication system dynamics using data from the PT. Unlike traditional frequentist models, Bayesian models inherently quantify model uncertainty by maintaining a probabilistic distribution over the model parameters [2, 3]. This enables the utilization of ensemble-based control, prediction, and analysis techniques. In essence, these techniques generate policies, predictions, and recommendations that consider the agreements and disagreements between multiple models built with the available information. High levels of disagreement among these models serve as a quantifiable indicator of significant model uncertainty.

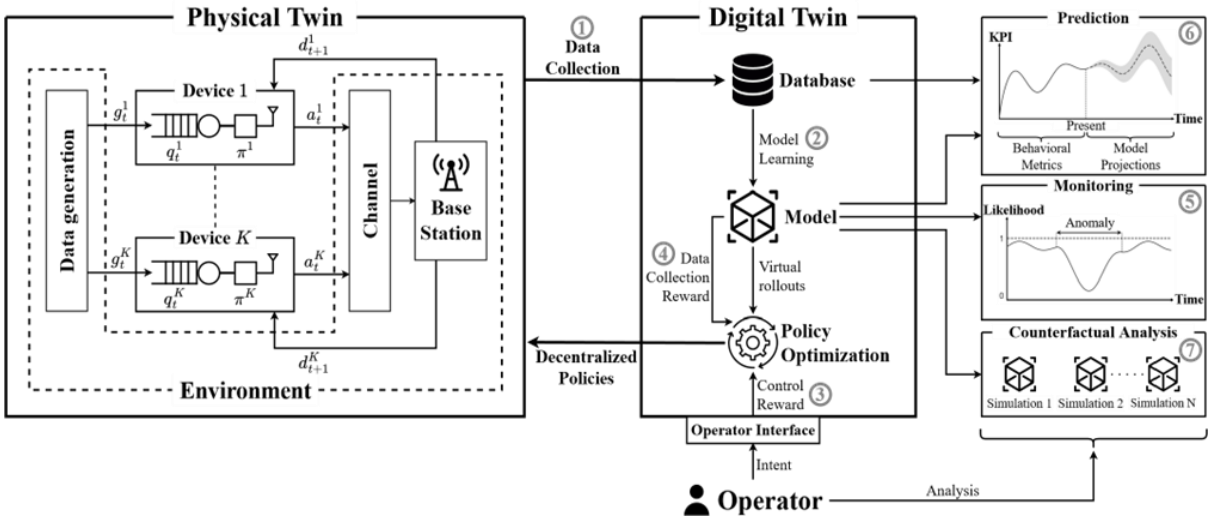

Figure 1: A DT platform controlling, monitoring, and analyzing the operation of a communication system operates along the phases of data collection, model learning, policy optimization, and data collection policy optimization.

The Bayesian model at the DT can naturally incorporate domain knowledge about the communication systems, including traffic and channel models, while enabling data-driven exploration of the system dynamics. With the available Bayesian model, the DT can carry out the core functionalities of control, monitoring, prediction, data-collection optimization, and counterfactual analysis, while providing uncertainty-aware outputs. We specifically investigate and detail control via model-based Bayesian multi-agent reinforcement learning (MARL), monitoring for anomaly detection, prediction of unobserved dynamics with

uncertainty quantification, and data collection optimization via directed model-based exploration.

2.1 Digital Twin and Physical Twin Systems

The system under study encompasses a multi-agent PT, which describes a telecommunications network, and a single DT located in the cloud, for a large PT system, or at the edge, for a local PT system. The DT collects data from the PT, either periodically or in an adaptive manner, and the data is used to optimize a model of the PT dynamics. The DT collects data from the PT over dedicated periods of time (phase 1 in Figure 1). Based on the data obtained in each data collection period, the DT constructs a model of the transition dynamics of the PT (phase 2). The model is used by the DT to recommend control policies to the PT (phase 3), as well as to carry out monitoring functionalities such as anomaly detection (phase 4), prediction (phase 6), and counterfactual analysis (phase 7).

Physical Twin: The PT system of interest consists of K agents that operate over a discrete time index t, The time index runs over the relevant time units for the system of interest, which are typically time slots or frames. At each time t, each agent k takes an *action* a_t^k from a discrete set of possible actions by following a *policy* that leverages information collected by the agent regarding the current *state* s_t of the overall system. The state evolves according to some *ground-truth transition probability* $T(s_{t+1}|s_t, a_t)$. Specifically, the probability distribution of the next state $s_{t+1} \sim T(s_{t+1}|s_t, a_t)$ is modelled as a *Markov decision process* (MDP), and only depends on the current state s_t and joint action $a_t = (a_t^1, \dots, a_t^K)$ of all agents.

At each time t, each agent k observes a function o_t^k of the overall state s_t. This captures the fact that an agent k typically has access only to *local* information about the state of the system. It is assumed that agents cannot communicate with each other, and thus the overall information available at agent k at time t amounts to its *action-observation history* h_t^k. Accordingly, the behavior of agent k is defined by a policy $\pi^k(a_t^k|h_T^K)$.

Digital Twin: The DT maintains a *model* of the PT ground-truth dynamics $T(s_{t+1}|s_t, a_t)$. To this end, the DT assumes a family of parametric models $T_\theta(s_{t+1}|s_t, a_t)$ that are determined by a parameter vector θ. In the model learning phase, the parameter vector θ is optimized based on data collected from the PT. To account for information available at the DT about the structure of the PT, we partition the state s_t into M distinct subsets of state variables, such that each subset s_t^i of state variables is a geographically and/or semantically distinct unit. Given the state subset and actions $\{a_t^k\}$ of all agents, we introduce a graph with M current state-nodes, one for each subset s_t^i ; K action-nodes, one for each action a_t^k ; and M future state-nodes, one for each subset.

2.2 Model Learning at the DT

Data collection: At the beginning of each data collection phase, the DT may provide the PT with a data-collection policy for each agent. These policies may be designed by the DT based on information about the PT prior to the data collection phase. Alternatively, the agents may follow fixed exploration policies. All agents in the PT execute their own policies during T time steps and the DT recovers the dataset $D_T^{\pi_d}$ of experienced transitions.

Bayesian Learning: Based on the dataset $D_T^{\pi_d}$, the DT seeks to optimize a parametric model to approximate the ground-truth unknown transition distribution $T(s_{t+1}|s_t, a_t)$. To this end, we propose that the DT adopts *Bayesian learning* to obtain a well-calibrated model. Bayesian learning aims at evaluating the posterior distribution $P(\theta|D_T^{\pi_d})$ of the unknown model parameters θ. Depending on the size of the state and action spaces, computing the exact posterior may not be feasible, and one should resort to function approximation.

2.3 Policy Optimization

During policy optimization (phase 3 in Figure 1), the DT aims at optimizing the *decentralized policy* $\pi = \{\pi^k(a_t^k|h_t^k)\}_{k\in K}$ of the K agents so as to maximize some user-specified performance criterion. This criterion is defined by a *reward* function, which determines the *total discounted return* when the PT applies the policy π. The optimal control problem consists of the maximization of the average long-term reward [10]. This amounts to a Decentralized MDP (Dec-MDP) [11]. The DT cannot directly interact with the PT during the policy optimization phase and must solely rely on the observed data $D_T^{\pi_d}$. Given that all policies are issued by the central DT platform, policy optimization can naturally rely on CTDE methods characterized by centralized training at the DT and decentralized execution at the PT. This class of approaches bypasses non-stationarity issues that affect decentralized learning schemes [12].

2.3.1 Control Policy Optimization

We adopt the COunterfactual Multi-Agent (COMA) algorithm in [13]. The key distinction between the approach adopted here and the conventional COMA implementation is the fact that the model $T_\theta(s_{t+1}|s_t, a_t)$ assumed here is stochastic in the sense that the model parameter vector θ is distributed according to the (approximate) posterior distribution $P(\theta|D_T^{\pi_d})$.

The proposed approach addresses the problem via model-generated virtual rollouts at the DT. In a manner similar to [14], we account for the epistemic uncertainty encoded by the posterior $P(\theta|D_T^{\pi_d})$ by periodically sampling a parameter vector $\theta \sim P(\theta|D_T^{\pi_d})$ during policy optimization so as to produce the next state $s_{t+1} \sim T_\theta(s_{t+1}|s_t, a_t)$ in the virtual rollouts.

In a manner similar to standard actor-critic algorithms [10], the DT maintains a centralized critic $Q_w(s_t|a_t)$, with parameter vector w, as well as the decentralized policies $\pi_v =$

$\{\pi_v^k(a_t^k|h_t^k)\}_{k\in K'}$ with common parameter vector v. During *policy evaluation*, the critic $Q_w(s_t|a_t)$ aims at approximating the *Q-value* $Q^{\pi_v}(s|a) = \mathbb{E}_{\pi^v}[G_t|s_t = s, a_t = a]$, i.e., the average future return under policy π_v starting from a given global state s and joint action a. Then, during *policy improvement*, the policies of all agents $k \in \mathrm{K}$ are updated to maximize the expected return. This is done by using the centralized critic $Q_w(s_t|a_t)$ to reward actions that enhance the performance at the system level. As we will detail next, during the policy optimization phase, we alternate between policy evaluation and policy improvement steps until convergence of the decentralized policy π_v. Upon convergence, only the learned policies need to be transmitted by the DT to their respective agents.

2.3.2 Data Collection Optimization

The data-collection policy $\pi_d(a_t|s_t)$ can be optimized by the DT based on the available data and on the DT's assessment about operating regimes characterized by more significant model uncertainty. For this purpose, the DT uses the available posterior parameter distribution $P(\theta|D_T^{\pi_d})$ to identify transitions (s_t, a_t, a_{t+1}) that yield high epistemic uncertainty, i.e., where models T_θ drawn from distribution $P(\theta|D_T^{\pi_d})$ disagree more significantly [2]. The resulting disagreement metric is used to engineer a data collection reward r_d. With such reward function, the data collection policy π_d is optimized at the DT.

The reward function r_d should capture the extent to which the ensemble of models $T_\theta(s_{t+1}|s_t, a_t)$ disagree on the prediction of the next state s_{t+1} given the previous-step state-action pair (s_t, a_t) [15]. One way to gauge this disagreement is to use the mutual information $I(s_{t+1}; \theta|s_t, a_t)$ evaluated under the posterior $P(\theta|D_T^{\pi_d})$.

2.4 Monitoring Functionalities

Anomaly Detection: the goal is to detect significant changes in the dynamics of the PT. To formulate this problem, assume that, during the operation of the system following policy optimization, the DT has access to the information about the state-action sequence experienced by the PT within some *monitoring time window* T^M under the optimized policy π. The DT tests if the collected data is consistent with the data reported by the PT during the most recent model learning phase (phase 1 and 2 in Figure 1), or rather if it provides evidence of changed conditions or anomalous behavior.

While frequentist learning is known to perform poorly for detection of out-of-distribution, or abnormal, samples, Bayesian learning has the key advantage of being capable of quantifying epistemic uncertainty via *disagreement-based test metrics*.

Prediction: One of the key motivations behind the model-based approach adopted by the DT paradigm is the possibility of predicting future states of the PT system by simulating the operation of the system via the model. While frequentist models would generally provide

unreliable measures of prediction uncertainty, Bayesian models can not only provide useful point predictions but also well-calibrated error bars.

To describe the problem, we define a *prediction time lag* T^H, corresponding to the number of time steps in the future we wish to predict, and a *target metric* y_p, which is a function of future trajectories within the prediction time window duration T^H, starting from a known state s_1. We also assume that the agents follow a known policy π. As an example, the metric of interest y_p may be the average number of packet losses for a subset of devices connected to the same base station over the next T^H time steps. Under these conditions, the DT can roll out the model defined by transitions T_θ and policy π to estimate statistics of the target metric y_p. With a Bayesian model, such statistics are further averaged over the posterior distribution, providing a reliable measure of prediction uncertainty. Accordingly, prediction using a Bayesian model requires a number of samples that is larger as compared to its frequentist counterpart by a factor given by the number of models sampled from the posterior.

Counterfactual Analysis: The predictive methodology described in the previous subsection is also a useful tool for counterfactual analysis of the PT behavior [16]. In such analysis, one wishes to assess the impact that changes in the system, as described by the ground-truth dynamics T, would have on some target metrics of interest. To this end, one could roll out different models T_θ or policies π implementing the given changes of interest, and then evaluate measures such as the average treatment effect [17].

2.5 Application to a Multi-Access System

As illustrated in Figure 1, the PT system under study comprises K sensing devices that obtain data with correlated data arrivals both in time and across devices and communicate with a common base station (BS) over a channel with an unknown distribution. Time is slotted, and each device may transmit in a slot if its buffer is not empty. We consider $K = 4$ sensing devices equipped with a buffer of capacity $Q_{max}^k = 1$ packet, with all buffers being initially empty. This scenario is of interest for devices that transmit updates, discarding previous packets from the queue as outdated. To account for spatial correlation, we partition the devices into clusters, where each cluster C_i contains devices with correlated packet arrivals.

The data generation distribution within each cluster does not depend on previously generated data and is such that both devices cannot simultaneously generate a packet, with a new packet being generated at either device with probability 0.4. This captures a situation in which devices monitor distinct parts of a process, e.g., the location of a target in distinct spatial regions. The channel allows for the successful transmission of a single packet with probability 1; while, for two simultaneous transmissions, one packet is received with probability 0.8 and both packets are received with probability 0.2. More than two simultaneous transmissions cause the loss of all packets.

We evaluate the performance of policy optimization in the ground-truth environment by using the following metrics: (i) the throughput, i.e., the average number of packets successfully sent at each time step; and (ii) the average probability of buffer overflow across all devices.

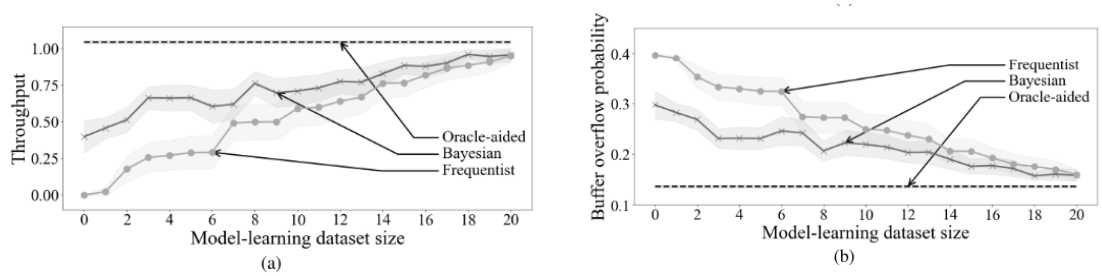

Figure 2: Throughput and buffer overflow probability as a function of the size of the dataset available in the model learning phase for the proposed Bayesian model-based approach, as well as the oracle-aided model-free and frequentist model-based benchmarks.

From Figure 2, we observe that, in regimes with high data availability during the model learning phase, i.e., with large T, both Bayesian and frequentist model-based methods yield policies with similar performance to the oracle-aided benchmark. In the low-data regime, however, Bayesian learning achieves superior performance as compared to its frequentist counterpart with, for instance, a 20% increase in throughput at $T = 10$.

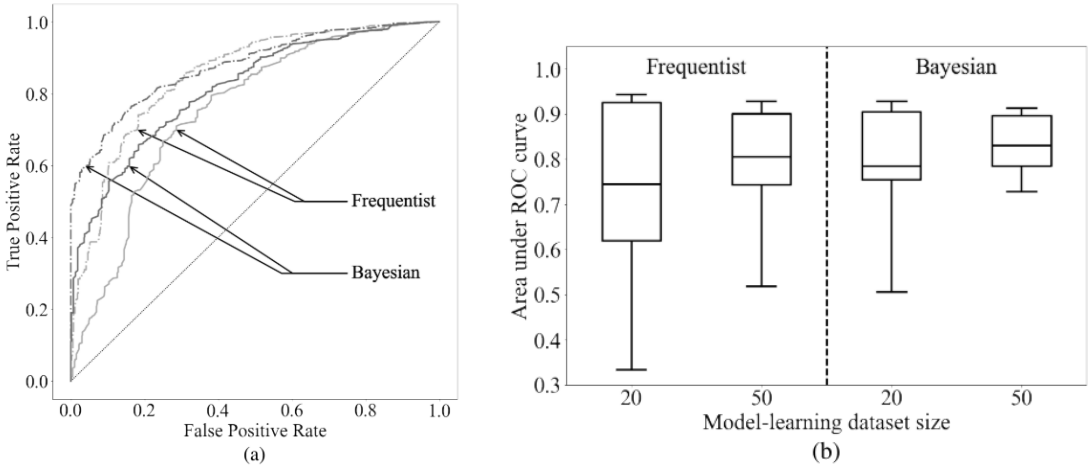

Figure 3: Mean ROC curves (a) and area under ROC curves (AUC) (b) for the Bayesian and frequentist anomaly detection tests. Solid lines in (a) represent model learning dataset sizes of T=20 steps, while dashed lines correspond to dataset sizes of T=50 steps. Mean AUCs in (b) are represented by a horizontal bar, while boxes denote the 25% and 75% quantiles and whiskers denote the 10% and 90% quantiles. Results are obtained from 50 independent data collection and model learning cycles.

We now consider the performance of anomaly detection by assuming that an anomalous event occurs when device 2 is disconnected, resulting in an anomalous packet-generation distribution for which a packet is generated at device 1 only with probability 0.4, and no packet is generated either at device 1 or 2 with probability 0.6. We then report the false positive rates (FPR) and the true positive rates (TPR) of the anomaly detection tests in Figure 3 by varying the detection threshold. The experiment is repeated 50 times over independent data collection and model learning phases, while the optimized policy used to report experiences remains the same.

For both model-learning dataset sizes of $T = 20$ and $T = 50$ steps in Figure 3b, Bayesian anomaly detection achieves, on average, a higher area under the receiver operating characteristic (ROC) curve; with a 5% average area increase and a 22% larger area at the 25% quantile for $T = 20$ compared to its frequentist counterpart. From Figure 5a, the proposed Bayesian framework is also observed to uniformly outperform the frequentist ROC curve for T = 20 steps, while providing higher performance at lower FPR for $T = 50$ steps. For instance, at a TPR of 0.75 in Figure 5a, the Bayesian anomaly detector has a FPR of 0.30 for a model learning dataset size of $T = 20$ and a FPR of 0.15 for a dataset size of T = 50; whereas the frequentist benchmark has a FPR of 0.34 for $T = 20$ and 0.21 for $T = 50$. These results suggest that measuring epistemic uncertainty, instead of likelihood, can yield more effective and robust monitoring solutions.

3 Calibrating Wireless Ray Tracing for Digital Twins

A key question regarding the digital twinning of radio access networks (RANs) is whether the built DT can simulate the propagation conditions of the deployment of interest. At high-fidelity levels, such simulation could be used to generate accurate site-specific channel responses at any deployment location without the need to collect real-world data via channel estimation. In this regard, RT is widely seen as an enabling technology for DTs of the RAN [3]. Given a three-dimensional model specifying the shapes, positions, and electromagnetic properties of the objects in the propagation environment, the RT simulates multiple propagation paths between any given transmitter (Tx) and receiver (Rx) through specular and diffuse reflection, refraction, and diffraction [18], [7].

3.1 Wireless Ray Tracing Calibration amidst Geometric Discrepancies

3.1.1 Wireless Ray Tracing Calibration

Compared to purely data-driven design, RT holds the promise of more accurate and explainable predictions of channel conditions, which hinge on prior physics-based knowledge of electromagnetic dynamics. Nonetheless, the fidelity of the virtual simulation depends on the precision of the geometric and electromagnetic properties, including permittivity, conductivity, and permeability, fed to the DT as inputs. While specialized reports [30] can provide approximate values for the electromagnetic properties of generic materials, a more precise and bespoke selection is generally necessary to ensure sufficiently reliable simulations. As illustrated in Figure 4, this optimization requires the estimation of the material parameters of each object in the scene based on measurements obtained from the ground-truth deployment scenario, in a process known as calibration [19].

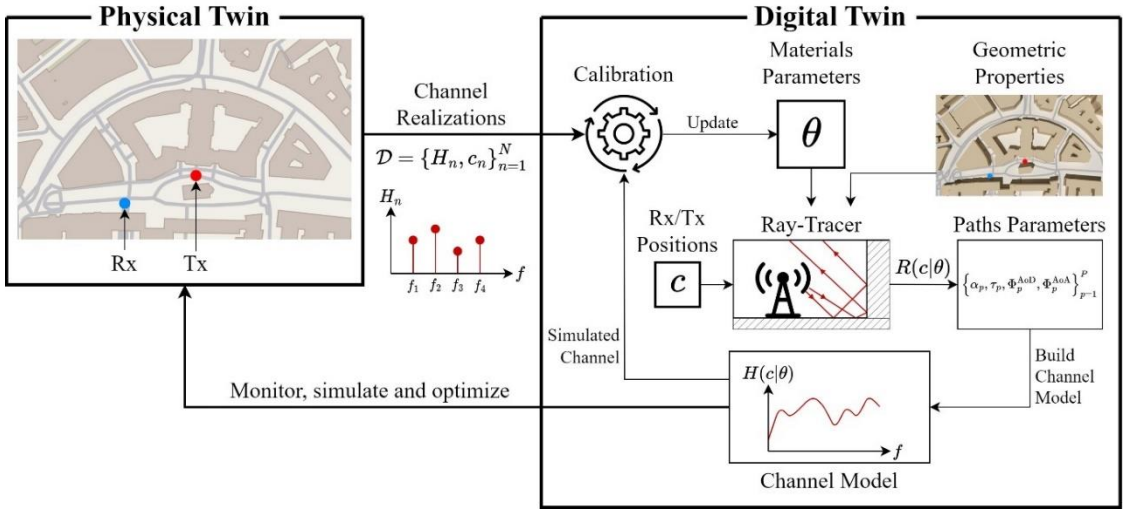

Figure 4: Taking as input the geometric properties of the scene, the electromagnetic material parameters θ, and the coordinates c of transmitter (Tx) and receiver (Rx), the ray tracer (RT) produces the features $R(c|\theta)$ of a number P of propagation paths. Based on this information, the DT can obtain a model $H(c|\theta)$ of the channel conditions between transmitter and receiver. To keep a faithful representation of its PT, during a calibration phase, the DT compares its model predictions $H(c|\theta)$ to measured channel realizations $\mathcal{D} = \{(H_n, c_n)\}_{i=1}^{N}$ in order to optimize the requirements on pilot transmissions and channel measurements.

In practice, calibration of the RT is achieved by adapting the materials parameters θ in a way that yields a good agreement between the RT-simulated channel $H(c_n|\theta)$ and its corresponding measurement H_n. Accordingly, we assume the availability of a dataset $\mathcal{D} = \{(H_n, c_n)\}_{n=1}^{N}$ of N channel responses H_n at locations $c_n = (c_n^{\mathrm{Rx}}, c_n^{\mathrm{Tx}})$, where $c_n^{\mathrm{Rx}} \in \mathbb{R}^3$ and $c_n^{\mathrm{Tx}} \in \mathbb{R}^3$ denote the n-th receive and transmit positions, respectively.

3.1.2 Effect of Geometric Errors

A shortcoming of wireless RT simulations is that the interference pattern predicted at the receiver is very sensitive to errors in the geometry used to represent the objects at the deployment location [7]. As illustrated in Figure 55, geometric discrepancies of even a fraction of the carrier's wavelength affect the phase of the predicted path, which can turn a constructive interference at the PT into a destructive interference simulated at the DT.

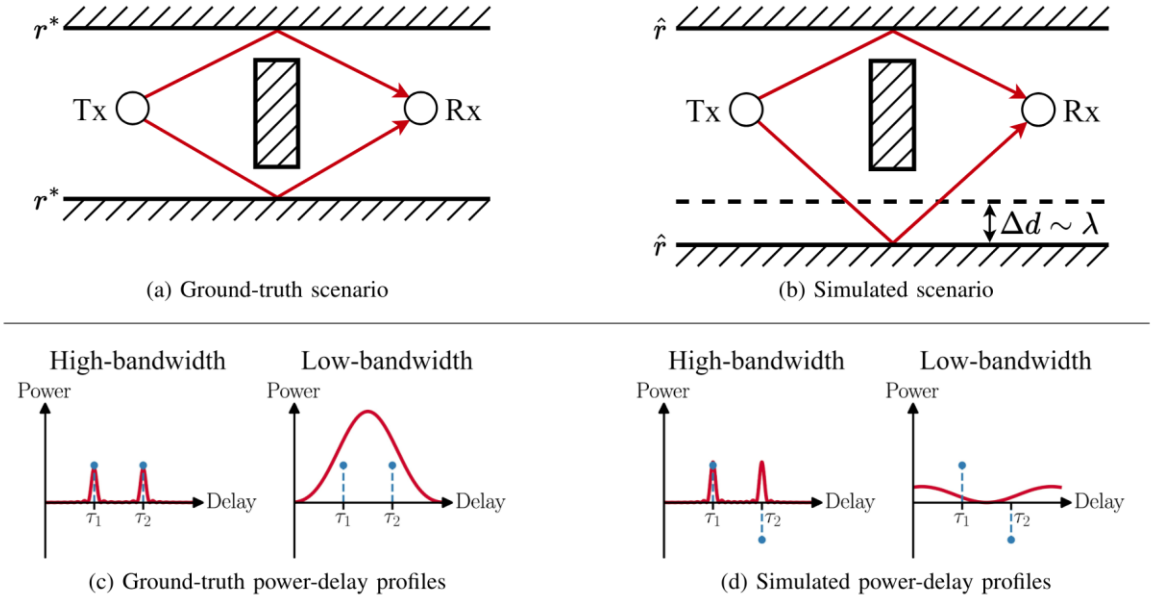

Figure 5: Toy example illustrating two propagation paths between a pair of transmit (Tx) and a receive (Rx) devices. Though the paths interfere constructively at the receiver under ground-truth conditions, they are predicted to interfere destructively in the simulated scenario due to an inaccuracy Δd of the order of the carrier's wavelength λ in the geometric model. This difference is illustrated as the blue dashed lines in (c) and (d), which represent the signed amplitudes of each path. Under high-bandwidth conditions (i.e., high temporal resolution), the simulated power profile (d) retrieves the correct power of each path in (c). However, when the system bandwidth is too low to separate the contribution of each path, the inaccurately predicted interference pattern yields an erroneous simulated power profile.

Under such circumstances, directly comparing the predicted channel $H(c_n|\theta)$ (dashed lines in Figure 55d) to the measured channel H_n (dashed lines in Figure 55c) during calibration results in inaccurate material parameters θ that try to compensate for errors in the predicted interference pattern. Comparing the ground-truth and simulated power profiles is a sound alternative at higher bandwidths but suffers from the same limitation in the more practical setting where bandwidth is limited. Our contribution is a calibration procedure that learns accurate material parameters θ on both high and low bandwidth settings by comparing the

measured channel H_n to a phase error-corrected channel response, accounting for potential errors in the geometry.

3.2 Phase Error Channel Model

In the following, a multi-carrier communication setting with S subcarriers is assumed to take place between two receive and transmit devices at location c equipped with arrays of N^{Rx} and N^{Tx} antennas, respectively.

3.2.1 Ray Traced Propagation Paths

The RT acts as a deterministic map from the input coordinates c and material parameters θ to the features of P propagation paths given by

$$R(c|\theta) = \left\{\alpha_p(\theta), \tau_p, \Phi_p^{\mathrm{AoD}}, \Phi_p^{\mathrm{AoA}}\right\}_{p=1}^{P},$$

where $\alpha_p(\theta) \in \mathbb{C}$ denotes the complex amplitude of the p-th path, τ_p denotes its delay [s], and where $\Phi_p^{\mathrm{AoD}} \in [0, \pi) \times [0, 2\pi)$ and $\Phi_p^{\mathrm{AoA}} \in [0, \pi) \times [0, 2\pi)$ are its angles of departure and arrival, respectively.

3.2.2 Path Response

Given paths parameters $R(c|\theta)$, the frequency response of the p-th path is given by $\alpha_p(\theta)\, a\left(\tau_p, \Phi_p^{\mathrm{AoD}}, \Phi_p^{\mathrm{AoA}}\right) \in \mathbb{C}^{S\, N^{\mathrm{Rx}}\, N^{\mathrm{Tx}}}$, where

$$a\left(\tau_p, \Phi_p^{\mathrm{AoD}}, \Phi_p^{\mathrm{AoA}}\right) = w(\tau_p) \otimes a^{\mathrm{Rx}}\left(\Phi_p^{\mathrm{AoD}}\right) \otimes a^{\mathrm{Tx}}\left(\Phi_p^{\mathrm{AoA}}\right)$$

encompasses the phase contributions of all combinations of subcarriers and antenna pairs as the tensor product of the Fourier basis $w(\tau) \in \mathbb{C}^S$ with the receive $a^{\mathrm{Rx}}(\Phi^{\mathrm{AoD}}) \in \mathbb{C}^{N^{\mathrm{Rx}}}$ and transmit $a^{\mathrm{Tx}}(\Phi^{\mathrm{AoA}}) \in \mathbb{C}^{N^{\mathrm{Tx}}}$ steering vectors.

3.2.3 Simulated Channel Response

In the unlikely scenario where the phase of each path is perfectly predicted by the RT, the simulated channel is simply computed as the sum of path responses $H(c|\theta) = \sum_{p=1}^{P} \alpha_p(\theta)\, a\left(\tau_p, \Phi_p^{\mathrm{AoD}}, \Phi_p^{\mathrm{AoA}}\right)$. To take into account the effect of errors in the predicted phase, we introduce a phase error channel model

$$H(c|\theta, \kappa_0) = \sum_{p=1}^{P} e^{jz_p} \alpha_p(\theta)\, a\left(\tau_p, \Phi_p^{\mathrm{AoD}}, \Phi_p^{\mathrm{AoA}}\right),$$

where the stochastic phase errors $Z = (z_1, \ldots, z_P) \sim \mathcal{VM}(0, \kappa_0)$ are assumed to be independent and identically distributed (i.i.d.) with respect to a von Mises distribution $\mathcal{VM}(0, \kappa_0)$ with concentration $\kappa_0 \geq 0$.

3.3 Phase Error-Aware Calibration

3.3.1 Expectation Maximization

In line with the proposed phase error channel model, our contribution is a *phase error-aware calibration* scheme that builds on the expectation-maximization framework [2] to estimate the material parameters from the data $\mathcal{D}$ in presence of predicted phase errors. Accordingly, for each measurement $(H_n, c_n) \in \mathcal{D}$, we estimate the phase errors $Z_n = (z_{n,1}, \dots, z_{n,P})$ during the expectation step (E-step). The maximization step (M-step) then uses the local phase error estimates $\{Z_n\}_{n=1}^N$ to update the material parameters θ by comparing the measurements H_n with phase-corrected channel prediction.

3.3.2 Expectation Step

The estimate of each phase error vector Z_n is represented by a variational distribution $Q(Z_n|\mu_n, \kappa_n) = \prod_{p=1}^P \mathcal{VM}(z_{n,p}|\mu_{n,p}, \kappa_{n,p})$ comprising P mean parameters $\mu_n = (\mu_{n,1}, \dots, \mu_{n,P})$ and P concentration parameters $\kappa_n = (\kappa_{n,1}, \dots, \kappa_{n,P})$ to be optimized. This choice of variational distribution allows for closed-evaluation of the parameters $\{\mu_n, \kappa_n\}_{n=1}^N$ that minimize the free-energy criterion for a fixed set of material parameters θ, yielding an efficient E-step computation.

3.3.3 Maximization Step

The M-step optimizes the material parameters θ by minimizing the free-energy criterion for a fixed set of phase correction parameters $\{\mu_n, \kappa_n\}_{n=1}^N$. This minimization problem is tackled through gradient descent by exploiting the differentiability of the predicted channel responses $H(c_n|\theta)$ with respect to the electromagnetic parameters θ, which is ensured by Sionna RT [19], [20].

3.4 Experimental Validation

In the following experiments, the proposed phase error-aware calibration scheme is compared to two baselines: a *phase error-oblivious calibration* approach which directly compares the predicted deterministic channel $H(c_n|\theta)$ to its corresponding measurement H_n disregarding potential phase errors [19]; and a *uniform phase error calibration* scheme which compares the angle-delay power profiles of the predicted and measured channels under the assumption that the predicted phase is uniformly distributed [21, 22, 23, 24, 25].

3.4.1 Urban Scene with Synthetic Data

We evaluate the proposed calibration scheme and baselines in a synthetic environment depicted in Figure 7, where all objects are assumed to share a single unknown material parameter vector θ^{true}. The dataset $\mathcal{D}$ is synthetically generated by simulating $N = 50$ independent channel realizations at a single location via RT under the unknown material parameters θ^{true}. Phase errors are generated by either random displacements of the receiver position (Figure 66a), or by drawing the channel realizations from the phase error channel

model $H(c|\theta^{\text{true}}, \kappa_0^{\text{true}})$ with a phase error standard deviation level given by κ_0^{true} (Figure 6b). Accuracy of the calibrated parameters θ is evaluated by the normalized received power error $\left|P_{\text{sig}}(c|\theta) - P_{\text{sig}}(c|\theta^{\text{true}})\right| / P_{\text{sig}}(c|\theta^{\text{true}})$ at the calibration location c, where $P_{\text{sig}}(c|\theta) = \sum_{p=1}^{P}\left|\alpha_p(\theta)\right|^2$ represents the total signal power on the receiver side.

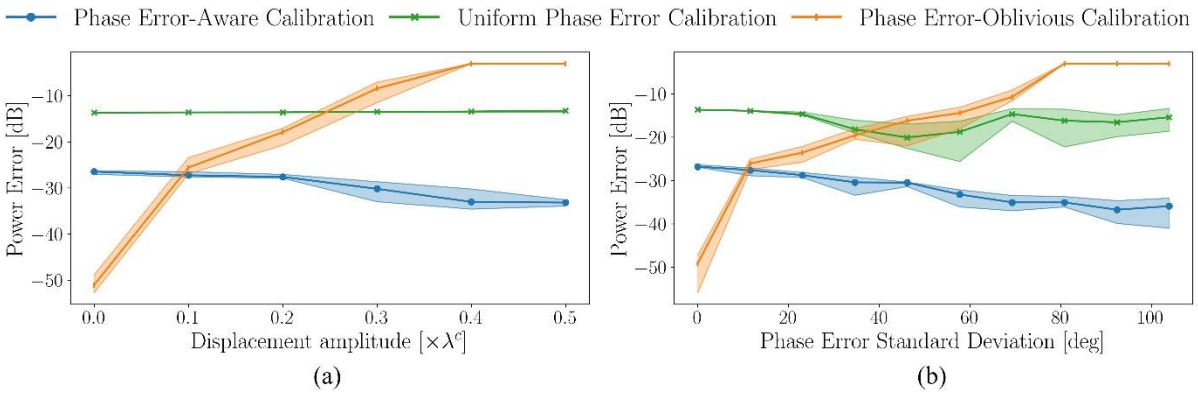

Figure 6: Calibration power estimation errors at the receiver as a function of the magnitude of the random receiver displacements (a), and as a function of the standard deviation of the independently sampled phase errors (b). Lines represent the median error across ten independent channel observation and calibration runs, an SNR of 20 dB. Shaded areas represent the first and third quartiles.

Generalization to new receiver locations is studied for calibration on synthetic channel data sampled from the phase error channel model $H(c|\theta^{\text{true}}, \kappa_0^{\text{true}})$ with no phase errors ($\kappa_0^{\text{true}} \to +\infty$) and uniform phase errors ($\kappa_0^{\text{true}} = 0$), as illustrated in Figure 77. The proposed phase error-aware calibration method outperforms both baselines on all settings, with the exception of the unrealistic setting of perfectly predicted phases.

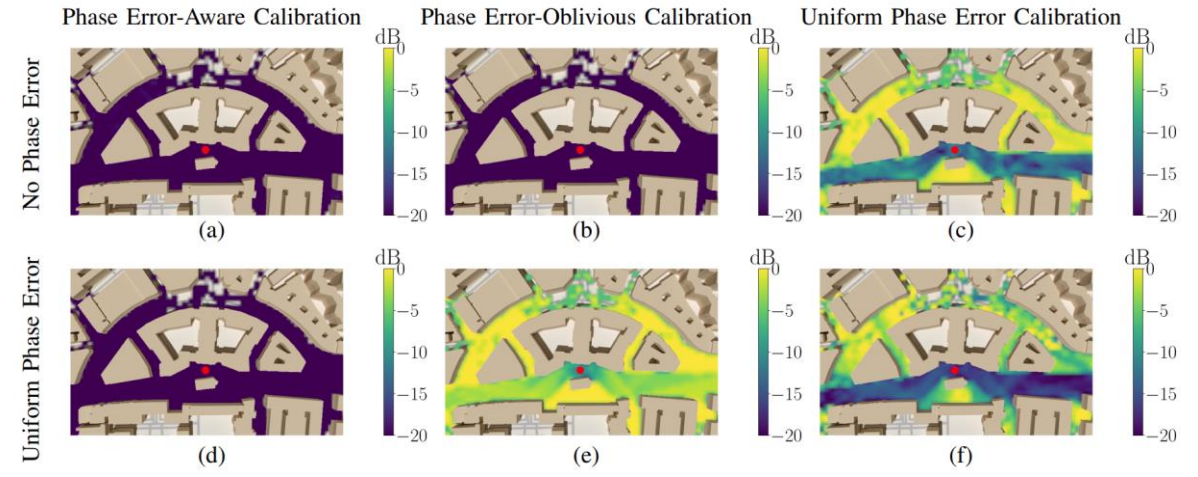

Figure 7: Relative estimation errors (in dB) for the average predicted power of the signal sent by a single transmitter (red dot) under phase error-aware calibration ((a) and (d)), phase error-oblivious calibration ((b) and (e)), and uniform phase error calibration ((c) and (f)) in the absence of phase errors $\kappa_0^{true} \to +\infty$ ((a), (b) and (c)), and for uniform phase errors $\kappa_0^{true} = 0$ ((d), (e) and (f)). The estimation errors at each position are averaged across ten independent channel observation and calibration procedures, with an SNR of 20 dB during calibration.

3.4.2 Wave Simulation Data

The presented calibration schemes are also evaluated on channel observations $\mathcal{D}$ generated by solving Maxwell's equations via finite-difference time-domain (FDTD) simulations. Given the computational complexity of FDTD simulations, the analysis is restricted to a small-sized two-dimensional setting similar to Figure 55 with a single material with unknown parameters θ^{true}. Phase errors are simulated by adding a small displacement $\Delta d = 0.25\lambda$ m to the lower wall in the geometry available at the DT. The scenario comprises one transmitter and three receivers, where data from the first two receivers is used to calibrate the material parameters θ. Data from the third receiver is used to evaluate the accuracy of calibration under different bandwidth settings by comparing the calibrated impulse responses to the ground-truth response in Figure 88. The proposed calibration yields channel impulse responses that closely match the FDTD-simulated responses, outperforming both baselines.

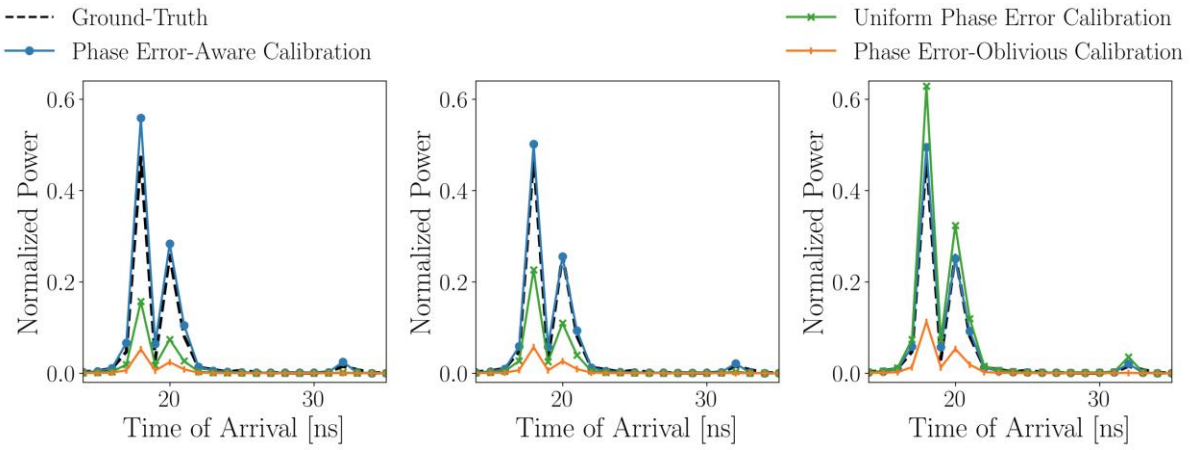

Figure 8: Power-delay profiles for the ground-truth FDTD-simulated channel (dashed line) and the RT-simulated channels (solid lines) using material parameters calibrated with bandwidths (a) B = 100 MHz, (b) B = 200 MHz, and (c) B = 500 MHz. Lines represent the median power value across ten independent channel observation and calibration runs, with an SNR equal to 20 dB.

4 Reliable Semi-Supervised Learning via Digital Twins

4.1 Problem Formulation and Background

Consider a scenario in which acquiring high-quality labeled data is costly, while unlabeled samples are abundant. Specifically, a labeled dataset $D = \{X_i, Y_i\}_{i=1}^{n}$ of independent and identically distributed (i.i.d.) samples from an unknown distribution $P_{XY} = P_X . P_{Y|X}$ is available, along with an unlabeled dataset $\widetilde{D} = \{\tilde{X}_i\}_{i=1}^{N}$ of i.i.d. samples drawn from the marginal distribution P_X. The unlabeled dataset is typically much larger than the labeled dataset, i.e., $N \gg n$.

Given a convex loss function $\ell_\theta(X, Y)$, the objective is to reliably estimate the minimizer $\theta^\star$ of the population loss $L(\theta)$, i.e.,

$$\theta^\star = \arg min_\theta \; L(\theta), \text{ where } \; L(\theta) = \mathbb{E}[\ell_\theta(X, Y)].$$

An unbiased estimator of the population loss can be obtained by using only the labeled data, yielding the classical empirical risk minimization (ERM) estimator. Assume the availability of a *model* $f(X)$ that provides an estimate of the label Y. The model $f(X)$ can be a pretrained ML model or any other given predictor. Conventional semi-supervised (SS) learning addresses the problem by combining both datasets to estimate the population loss [26]. Accordingly, conventional semi-supervised learning optimizes a biased estimate of the population loss, and the bias may cause significant performance degradation when the model $f(X)$ is not sufficiently accurate [27].

Prediction-powered inference (PPI) [27] and its variants, tuned PPI [28] and cross-PPI [29], provide principled alter- natives to the conventional semi-supervised estimator, which have been shown to have the desirable theoretical guarantees and empirical performance. We briefly review the variants of PPI, providing the necessary background for the introduction of the proposed tuned cross-PPI.

Prediction-powered inference (PPI): PPI uses the labeled data to quantify, and compensate for, the prediction bias of the model $f(X)$ as compared to the ground-truth labels. Specifically, PPI estimates the population loss as

$$L^{PP}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \ell_\theta\big(\tilde{X}_i, f(\tilde{X}_i)\big) - \left[\frac{1}{n}\sum_{i=1}^{n} \ell_\theta\big(X_i, f(X_i)\big) - \frac{1}{n}\sum_{i=1}^{n} \ell_\theta(X_i, Y_i)\right].$$

Tuned PPI: PPI is not guaranteed to improve over ERM when the model $f(X)$ is not sufficiently accurate. To address this issue, reference [28] proposed *tuned PPI*, a variant of PPI that automatically adapts to the quality of the prediction model. Tuned PPI selects the parameter vector that minimizes the following loss

$$L_{\lambda}^{PP}(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell_{\theta}(X_i, Y_i) - \lambda\left[\frac{1}{N}\sum_{i=1}^{N} \ell_{\theta}\left(\tilde{X}_i, f\left(\tilde{X}_i\right)\right) - \frac{1}{n}\sum_{i=1}^{n} \ell_{\theta}\left(X_i, f(X_i)\right)\right],$$

where $\lambda \in [0,1]$ is a tuning parameter.

Cross PPI (CPPI): PPI and tuned PPI assume the availability of a model $f(X)$. In practice, however, such model may have to be trained using labeled data. Therefore, the available labeled dataset must be shared between the task of obtaining the prediction model and the task of estimating the parameter vector. CPPI addresses this problem via cross-validation, where the labeled dataset is divided into K folds, $D^1, \ldots, D^K$. For each k, a model $f^k(X)$ is trained on all labeled data except fold D^k. Then, the following loss estimate is used [29]

$$L_{\lambda}^{PP}(\theta) = \frac{1}{NK}\sum_{k=1}^{K}\sum_{i=1}^{n} \ell_{\theta}\left(\tilde{X}_i, f^k\left(\tilde{X}_i\right)\right) - \left[\sum_{k=1}^{K}\sum_{i\in D^k} \ell_{\theta}\left(X_i, f^k(X_i)\right) - \frac{1}{n}\sum_{i=1}^{n} \ell_{\theta}\left(X_i, f(X_i)\right)\right],$$

The first term is the empirical loss that uses the predictions of the K models on the unlabeled data, while the second, rectifier term, corrects the bias caused by the use of the trained models.

4.2 Tuned Cross-Prediction-Powered Inference

The quality of the CPPI estimate depends on the accuracy of the trained models $\{f^k(X)\}_{k=1}^{K}$. Inspired by tuned PPI, we propose *tuned CPPI*, which provides the flexibility to judiciously adapt depending on the quality of the trained models. Specifically, the proposed tuned CPPI estimator is given by

$$L_{\lambda}^{CP}(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell_{\theta}\left(X_i, f(X_i)\right) + \lambda\left[\frac{1}{NK}\sum_{k=1}^{K}\sum_{i=1}^{n} \ell_{\theta}\left(\tilde{X}_i, f^k\left(\tilde{X}_i\right)\right) - \sum_{k=1}^{K}\sum_{i\in D^k} \ell_{\theta}\left(X_i, f^k(X_i)\right)\right].$$

Tuned CPPI offers the flexibility to tune the parameter $\lambda \in [0,1]$ as a function of the quality of the trained models. This is done with the aim of minimizing the mean squared error (MSE) of the parameter estimate. To this end, an asymptotic convergence analysis is conducted in the regime where the size of labeled data grows large, which allows to obtain an asymptotic equivalent of the MSE. Then, the minimizer of the asymptotic is obtained in closed form expression that can be estimated from the available labeled and unlabeled datasets, as detailed in our paper [30].

4.3 Application: Beam Alignment in mmWave Massive MIMO

Beamforming design is a crucial task in mmWave massive MIMO systems [31], as beamforming is necessary to compensate for the more severe path loss experienced at higher carrier frequencies. Codebook-based beam alignment consists of selecting the best beam from codebooks of predefined beams based on beam sweeping, which requires transmission of pilot signals. Recently, a new approach has emerged that alleviates the training overhead

by leveraging the concept of a channel knowledge map (CKM). A CKM is a site-specific database of channel information linked to transmitter and receiver locations [32, 33].

Applying the proposed tuned CPPI, we develop a method that trains a mapping between a user's location and a pair of beams within their respective codebooks by leveraging both labeled and unlabeled data. Labeled data consists of a user's location and the corresponding channel state information (CSI), while unlabeled data only includes a user's location. Furthermore, for the unlabeled inputs *X*, CSI parameters are estimated using a CKM. The unlabeled dataset is generated by sampling users' locations from the coverage map of the transmitter. The latter can be obtained in the DT using RT techniques. We compare the performance of the proposed tuned CPPI approach to the benchmark schemes ERM, SS, PPI, tuned PPI, and CPPI.

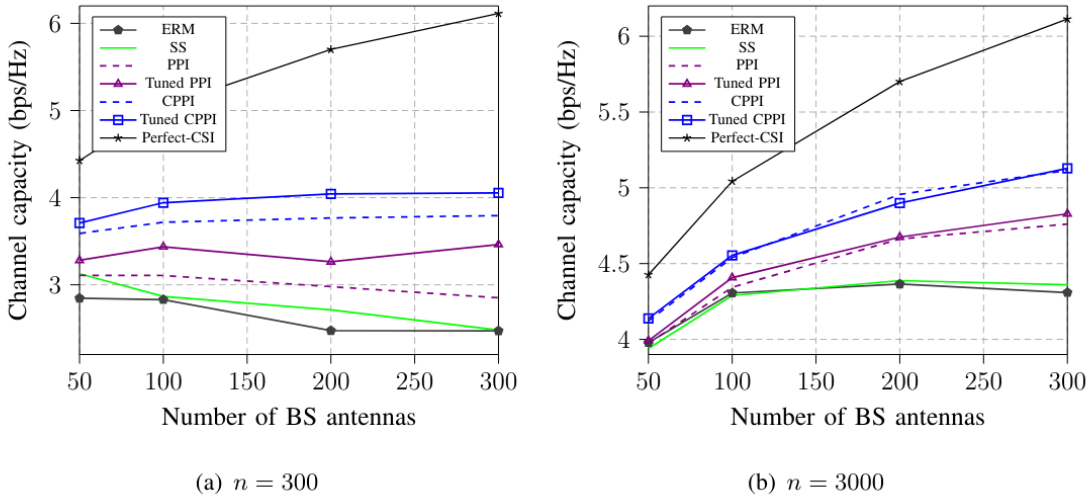

Figure 9*: Channel capacity as a function of the number of BS antennas for different values of the labeled dataset size n.*

We consider the same physical environment and dataset as in [32]. The dataset contains ground-truth multi-path channel information generated by using the *ray tracing* software Remcom Wireless Insite. The BS is equipped with a uniform planar array, and the user equipment (UE) is equipped with a single antenna. Furthermore, Kronecker product-based beamforming codebooks are employed [49]. The total number of samples available in the dataset is 38038, from which *n* samples are reserved as labeled data, and the remaining $N = 38038 - n$ samples are considered as unlabeled data.

In Figure 9, we report the performance in terms of the channel capacity for all schemes. We vary the number of transmit antennas at the BS while setting the size of the labeled dataset to *n* = 300 and *n* = 3000. It is observed that the proposed tuned CPPI outperforms all the benchmark schemes while exhibiting the same performance as CPPI when the number of labeled data points, *n*, is large enough. We also remark from the figure that conventional semi-supervised learning, which does not take into consideration the bias from the trained prediction models, fails to provide any gains by incorporating the unlabeled data.

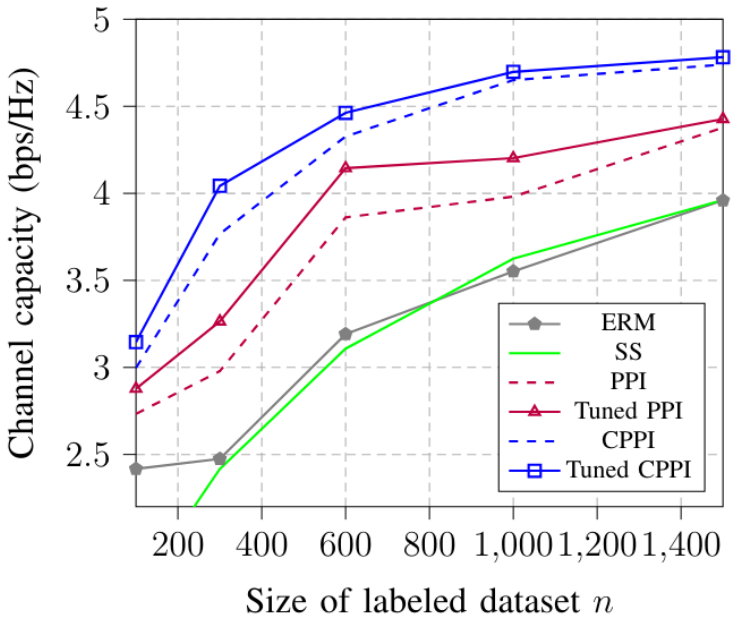

Figure 10: Channel capacity as a function of the size of labeled dataset n when the number of BS antennas is fixed to $N^{TX} = 200$.

To further elaborate on the impact of the number of labeled data points, *n*, we report the channel capacity as a function of the labeled dataset size *n* in Figure 10. As seen, the proposed tuned PPI provides the best performance among all the benchmark schemes, with more significant gains for smaller values of the labeled dataset size *n*. In fact, in this regime, the trained CKM is not sufficiently accurate, and tuned CPPI, which can adapt to the quality of the trained models via the tuning parameter λ, yields better performance.

5 Conclusion

This deliverable presented a comprehensive of the algorithms developed within T2.3 of WP2, aimed at improving the reliability of DT platforms for training and monitoring AI methods in 6G networks. Our comprehensive approach includes managing AI model uncertainty, calibrating data within the DT, and integrating real data from PTs with synthetic data. Key innovations include a Bayesian framework to manage data limitations, enhancing DT functionalities like multi-agent reinforcement learning, anomaly detection, and predictive analysis. Experimental validation showed this framework outperforms traditional frequentist models. Additionally, a novel calibration scheme for wireless RT corrects phase errors using the variational expectation maximization algorithm, significantly improving prediction accuracy. Finally, the introduced semi-supervised learning method, tuned CPPI, effectively leverages synthetic labels while reducing bias. Applied to beam alignment in mmWave MIMO systems, this method demonstrated superior performance over conventional approaches. These solutions collectively enhance the robustness and precision of DT platforms, contributing to more reliable and efficient 6G network operations.

References

[1] D. J. MacKay, Information theory, inference and learning algorithms, Cambridge University Press, 2003.

[2] O. Simeone, Machine learning for engineers, Cambridge University Press, 2022.

[3] A. Alkhateeb, S. Jiang and G. Charan, "Real-time digital twins: Vision and research directions for 6G and beyond," *IEEE Communications Magazine,* vol. 61, no. 11, pp. 128-134, 2023.

[4] B. Salehi, U. Demir, D. Roy, S. Pradhan, J. Dy, S. Ioannidis and K. Chowdhury, "Multiverse at the edge: interacting real world and digital twins for wireless beamforming," *IEEE/ACM Transactions on Networking,* 2024.

[5] V.-H. Nguyen, V. Corlay, N. Gresset and C. Ciochina, "Probabilistic Ray-Tracing Aided Positioning at mmWave frequencies," in *nternational Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2023.

[6] P. Ferrand, M. Guillaud, C. Studer and O. Tirkkonen, "Wireless channel charting: Theory, practice, and applications," *IEEE Communications Magazine,* vol. 61, no. 6, pp. 124-130, 2023.

[7] A. F. Molisch, Wireless communications, John Wiley & Sons, 2012.

[8] Y. Sun, M. Peng, Y. Zhou, Y. Huang and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Communications Surveys & Tutorials,* vol. 21, no. 4, pp. 3072--3108, 2019.

[9] X. Yang, Z. Song, I. King and Z. Xu, "A survey on deep semi-supervised learning," *IEEE Transactions on Knowledge and Data Engineering,* vol. 35, no. 9, pp. 8934--8954, 2022.

[10] R. Sutton and A. Barto, Reinforcement learning: An introduction, MIT Press, 2018.

[11] F. A. Oliehoek and C. Amato, A concise introduction to decentralized POMDPs, Springer, 2016.

[12] G. J. Laurent, L. Matignon and L. Fort-Piat, "The world of independent learners is not Markovian," *International Journal of Knowledge-based and Intelligent Engineering Systems,* vol. 15, no. 1, pp. 55--64, 2011.

[13] J. Foerster, G. Farquhar, T. Afouras, N. Nardelli and S. Whiteson, "Counterfactual multi-agent policy gradients," in *Proceedings of the AAAI conference on artificial intelligence*, 2018.

[14] Q. Zhang, C. Lu, A. Garg and J. Foerster, "Centralized model and exploration policy for multi-agent RL," *arXiv preprint arXiv:2107.06434,* 2021.

[15] P. Shyam, W. Jaskowski and F. Gomez, "Model-based active exploration," in *International conference on machine learning*, 2019.

[16] L. Hui, M. Wang, L. Zhang, L. Lu and Y. Cui, "Digital twin for networking: A data-driven performance modeling perspective," *IEEE Network,* vol. 37, no. 3, pp. 202--209, 2022.

[17] W. Larry, "All of statistics: a concise course in statistical inference," Springer, 2004.

[18] Z. Yun and M. F. Iskander, "Ray tracing for radio propagation modeling: Principles and applications," *IEEE access,* vol. 3, pp. 1089--1100, 2015.

[19] J. Hoydis, F. A. Aoudia, S. Cammerer, M. Nimier-David, N. Binder, G. Marcus and A. Keller, "Sionna RT: Differentiable ray tracing for radio propagation modeling," in *IEEE Globecom Workshops*, 2023.

[20] J. Hoydis, S. Cammerer, F. A. Aoudia, A. Vem, N. Binder, G. Marcus and A. Keller, "Sionna: An open-source library for next-generation physical layer research," *arXiv preprint arXiv:2203.11854,* 2022.

[21] J. Jemai, P. C. Eggers, G. F. Pedersen and T. Kurner, "Calibration of a UWB sub-band channel model using simulated annealing," *IEEE Transactions on Antennas and Propagation,* vol. 57, no. 10, pp. 3439--3443, 2009.

[22] R. Charbonnier, C. Lai, T. Tenoux, D. Caudill, G. Gougeon, J. Senic, C. Gentile, Y. Corre, J. Chuang and N. Golmie, "Calibration of ray-tracing with diffuse scattering against 28-GHz directional urban channel measurements," *IEEE Transactions on Vehicular Technology,* vol. 69, no. 12, pp. 14264--14276, 2020.

[23] S. Priebe, M. Jacob and T. Kürner, "Calibrated broadband ray tracing for the simulation of wave propagation in mm and sub-mm wave indoor communication channels," in *18th European Wireless Conference*, 2012.

[24] O. Kanhere and T. S. Rappaport, "Calibration of NYURay, a 3D mmWave and sub-THz ray tracer using indoor, outdoor, and factory channel measurements," in *IEEE International Conference on Communications*, 2023.

[25] G. S. Bhatia, Y. Corre and M. Di Renzo, "Tuning of ray-based channel model for 5G indoor industrial scenarios," in *IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*, 2023.

[26] O. Chapelle, B. Schölkopf and Z. Alexander, Semi-supervised learning, The MIT Press, 2006.

[27] A. N. Angelopoulos, S. Bates, C. Fannjiang, M. I. Jordan and T. Zrnic, "Prediction-powered inference," *Science,* vol. 382, no. 6671, pp. 669--674, 2023.

[28] A. N. Angelopoulos, J. C. Duchi and T. Zrnic, "PPI++: Efficient prediction-powered inference," *arXiv preprint arXiv:2311.01453,* 2023.

[29] T. Zrnic and E. J. Candès, "Cross-prediction-powered inference," *Proceedings of the National Academy of Sciences,* vol. 121, no. 15, p. e2322083121, 2024.

[30] H. Sifaou and O. Simeone, "Semi-Supervised Learning via Cross-Prediction-Powered Inference for Wireless Systems," *arXiv preprint arXiv:2405.15415,* 2024.

[31] M. Giordani, M. Polese, A. Roy, D. Castor and M. Zorzi, "A tutorial on beam management for 3GPP NR at mmWave frequencies," *IEEE Communications Surveys & Tutorials,* vol. 21, no. 1, pp. 173--196, 2018.

[32] Y. Zeng and X. Xu, "Toward environment-aware 6G communications via channel knowledge map," *IEEE Wireless Communications,* vol. 28, no. 3, pp. 84--91, 2021.

[33] D. Wu, Y. Zeng, S. Jin and R. Zhang, "Environment-aware and training-free beam alignment for mmWave massive MIMO via channel knowledge map," in *IEEE International Conference on Communications Workshops*, 2021.