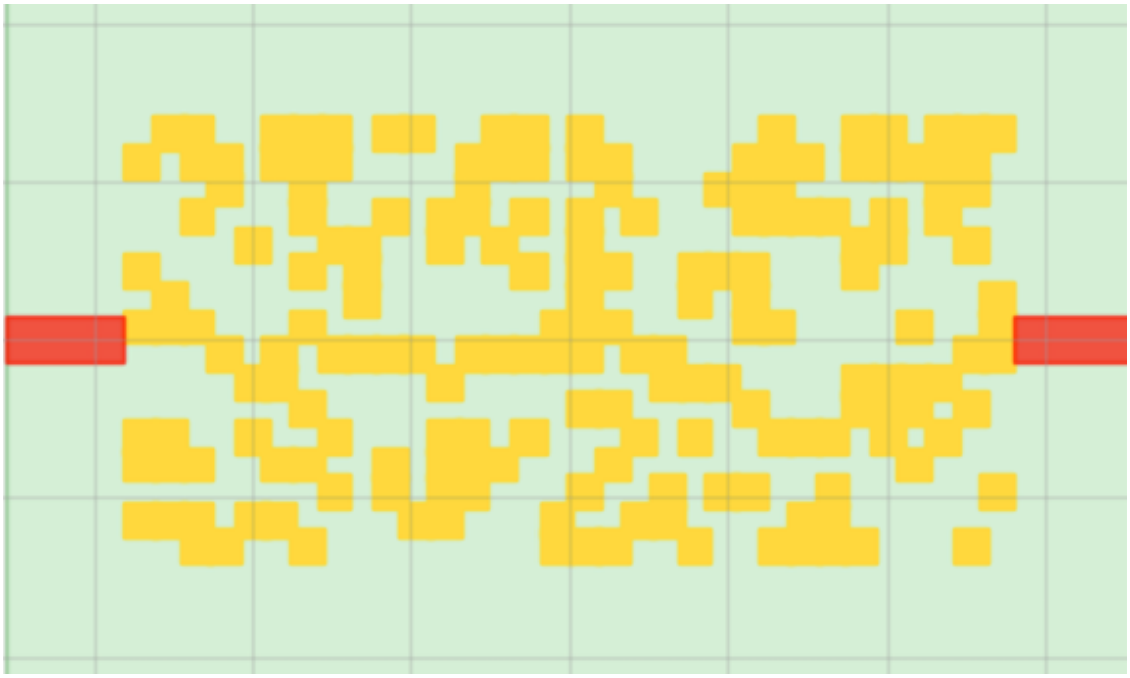

Data-Efficient AI-Driven RF Circuit Design via Multi-Fidelity Surrogates

50x reduction in full-wave simulation requirement

Degree project report in Wireless, Photonics, and Space Engineering

Yan Yan

DEPARTMENT OF MICROTECHNOLOGY and NANOSCIENCE
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026
www.chalmers.se

Degree project report 2026

Data-Efficient AI-Driven RF Circuit Design via Multi-Fidelity Surrogates

50x reduction in full-wave simulation requirement

Yan Yan

Supervisor: Yin Zeng, ScalinQ AB

Examiner: Jan Grahn, Chalmers

Department of Microtechnology and Nanoscience
Chalmers University of Technology
Gothenburg, Sweden 2026

Data-Efficient AI-Driven RF Circuit Design via Multi-Fidelity Surrogates
50x full-wave simulation requirement reduction
Yan Yan

Supervisor: Yin Zeng, Senior RF Engineer, ScalinQ AB
Examiner: Jan Grahn, full professor, Department of Microtechnology and Nanoscience, Chalmers

Degree project report 2026
Department of Microtechnology and Nanoscience
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone +46 79 332 2094

Cover: Layout of a 9 GHz low-pass filter from inverse design workflow using 16×32 pixelation.

Typeset in LaTeX
Gothenburg, Sweden 2026

Abstract

AI-driven inverse design of RF circuits offers a promising approach to achieving a high degree of design automation. However, training an underlying neural network surrogate typically requires a massive number of training samples. In addition, these surrogates often suffer from limited generalizability. For instance, redefining the design space often requires completely rebuilding the full-wave electromagnetic simulation dataset. Thus, data generation remains a major computational bottleneck. This thesis investigates how to mitigate this data-demanding challenge. We develop a fully open-source, multi-fidelity framework where a neural network surrogate is pre-trained on a large-scale dataset derived from fast analytical approximations, and subsequently refined via transfer learning using a sparse set of high-fidelity EM simulations. This approach is validated by designing low-pass filters on a two-layer PCB with an 8×8 grid. The framework achieves target predictive accuracy while reducing the required high-fidelity EM training samples from 10k to 200, representing a 50-fold reduction in training data requirement. Consequently, the total data generation time drops from 83.3 hours to 5.2 hours, yielding a 16-fold computational speedup. This methodology provides a practical and data-efficient strategy to improve the overall efficiency of automated RF hardware design.

List of Abbreviations

CNN convolutional neural network. 1, 13, 24

FWS full-wave simulation. 1, 3–8, 10, 14, 15, 18–22, 24

GA genetic algorithm. 3, 15, 22, 24

LPF low-pass filter. 1

MAPES Multiport Analytical Pixel Electromagnetic Simulator. 6, 7, 11–15, 18–22, 24

MSL micro-strip line. 7–9, 11, 12, 24

NN neural network. 1, 3–6, 8, 12–15, 18, 22, 25

PNGF precomputed numerical Green function. 6

RMSE root mean square error. 15, 18–22, 24

SMM Scattering Matrix Method. 6

TL transfer learning. 1, 5–7, 14, 15, 18–22, 24, 25

Contents

1
Introduction

The demand for compact, wide-band, and high-performance passive components in modern RF and microwave systems continues to grow. Consequently, designing structures like filters and matching networks has become highly time-consuming. Traditional design workflows rely heavily on expert intuition and repetitive full-wave simulations (FWSs). Since every geometric adjustment requires a new simulation, this process is slow, computationally expensive, and difficult to scale to more complex structures.

Inverse design offers a promising alternative. Instead of manually tuning the parameters, the designer specifies a target EM behavior and an automated optimizer searches the layout space for geometries that meet the criteria. The constitution of the layout space often determines the upper limit of performance. Among the possible ways to expand the design space, pixelation provides high geometric flexibility and can capture unconventional structures that are difficult to derive analytically. However, the success of this approach depends on evaluating the performance of a huge number of candidates both quickly and accurately. Surrogate models, especially neural network (NN) surrogates, can predict EM responses orders of magnitude faster than FWS, but typically require large amounts of high-fidelity training data and often generalize poorly across different design settings. Consequently, changes to the design space (e.g., a different pixel grid, substrate, or feed topology) frequently require generating expensive full-wave datasets from scratch, and once again facing the computational bottleneck.

This thesis proposes a data-efficient, multi-fidelity surrogate training strategy to address these limitations. The core idea is to leverage fast analytical EM approximations to generate large low-fidelity datasets for pretraining, and then refine the pretrained NN model with a small set of high-fidelity FWSs via transfer learning (TL). By capturing global physical trends with inexpensive data and correcting residual errors with a sparse high-fidelity set, the approach aims to preserve predictive accuracy while dramatically reducing the number of FWSs required for inverse design. The main contributions of this work are as follows.

- A **fully open-source** inverse design workflow that utilizes a surrogate model based on convolutional neural network (CNN) with residual block.
- A multi-fidelity approach that significantly reduces the requirement of high-fidelity data for NN training.
- A demonstration of low-pass filter (LPF) design using an 8×8 pixel grid on PCB, showing a reduction in required high-fidelity training samples from 10,000 to 200 and a corresponding reduction in total data generation time from 83.3 hours to 5.2 hours.

The remainder of this thesis is organized as follows. Chapter 2 provides an overview of previous works on inverse design, surrogate modeling, analytical EM solvers, and multi-fidelity learning, revealing the computation bottleneck and proposing a method to address this challenge. Chapter 3 describes step by step how the proposed approach was validated, followed by Chapter 4 where experimental results are presented and explained. Finally, Chapter 5 concludes the thesis and directions for future research.

2

Background & Literature Review

2.1 AI-enabled RF inverse design

Recent advances in artificial intelligence (AI) have made it a powerful tool for automation and decision-making across various engineering domains. For instance, large language models [1] and their multimodal variants [2] can act as agents to assist the analog circuit design workflows. In such workflows, AI agents can guide circuit topology exploration and device sizing by evaluating circuit performance through schematic-level simulations and determining optimization directions.

However, applying similar agent-based approaches to RF circuit design introduces challenges. Specifically, embedding the necessary RF domain knowledge into an AI agent to guide optimization effectively is difficult for two main reasons. First, unlike analog circuits that operate primarily on lumped-element models, RF circuits involve with distributed EM phenomena, where parasitic and coupling effects need to be carefully accounted for. Second, analytical approximations often fail to capture EM behaviors, leaving Maxwell's equations as the fundamental starting point. Hence, evaluating the performance of RF circuits requires FWS, which is significantly more time-consuming than standard schematic-level analysis. As a result, deploying a fully automated AI-agent-driven, FWS-based, trial-and-error loop for RF design remains computational impractical.

To improve design efficiency, researchers have explored several techniques, notably inverse design, NN-based surrogate modeling, and expanding the design space through pixelation [3, 4, 5, 6, 7, 8, 9, 10].

Inverse design and surrogate modeling primarily aim to shorten the design cycle by minimizing repetitive manual tuning and performance verification. In a typical inverse design workflow (Fig. 2.1), the designer starts from a set of specifications that are defined by the design target. Then a design space is constructed by the specifications such as materials and footprint. Meanwhile, an objective function is determined by the expected results of the EM simulation. The design space includes all possible configurations (i.e., EM structures) that can be explored. The objective function evaluates the performance of each configuration, based on the EM response of the structure.

This EM response can be obtained either directly from an EM simulator or from a surrogate model, which is usually trained on a large dataset generated by EM simulations. With an optimization algorithm such as Bayesian optimization or genetic algorithm (GA), the design space is then explored iteratively. The iteration continues until a satisfactory score is reached, then candidate layouts are produced.

Although this specification-to-layout workflow is possible using pure FWS [11], NN surrogates are highly preferred. A single FWS might take minutes to hours to complete, while a well-trained NN could execute thousands of predictions in just a second on a modern GPU. This acceleration is particularly critical when navigating a vast design space.

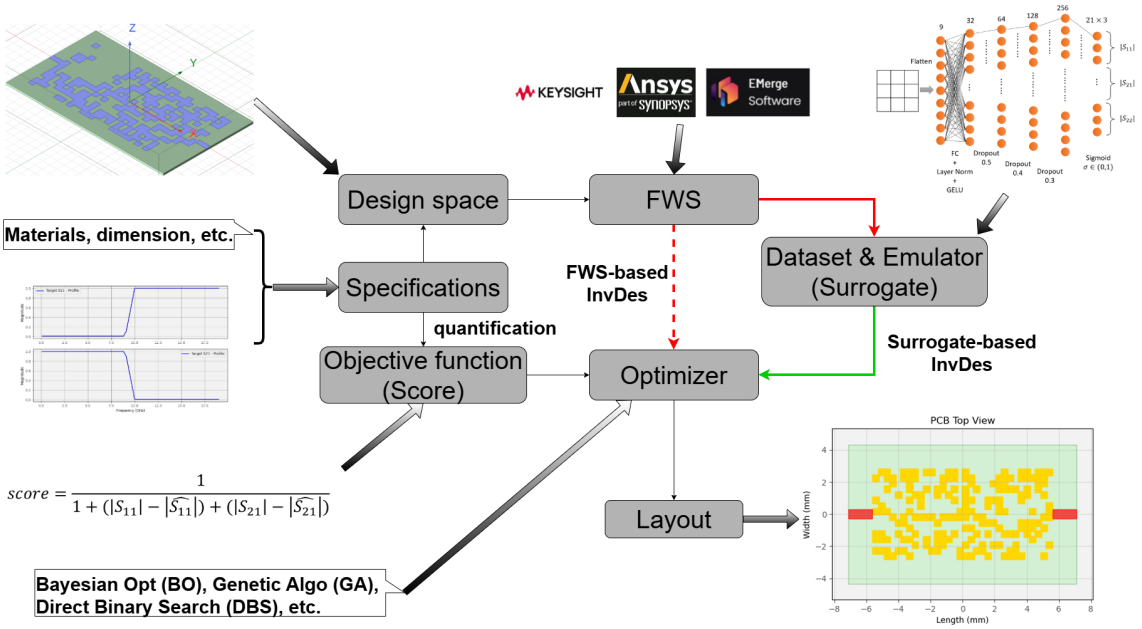

Figure 2.1: A typical inverse design workflow for RF circuits.

Traditional template-based design often suffers from a limited design space. As a result, the final design may turn out to be far away from the global optimum, as illustrated in Fig. 2.2. To expand the design space, template-seeded pixelation [8, 9] has been proposed. This approach provides a balance between the size of the design space and the requirement of training dataset.
Ultimately, full pixelation [3, 4, 7, 10, 5] offers the largest design space among all the discussed methods. Prior work [5] has demonstrated that inverse design with full pixelation can surpass traditional template-based designs created by human engineers.
However, these AI-based surrogates face two fundamental engineering challenges:

- Massive data requirements:
  Training a high-fidelity surrogate typically requires hundreds of thousands of FWSs. As summarized in Table. 2.1. Even with high-performance computing resources [4, 12], generating such datasets remains time-consuming. With standard computing hardware [9], the required time can easily become impractical.
- Limited generalizability:
  Even large datasets cover only an infinitesimal fraction of the design space (e.g., $\approx 10^{-92}$ coverage for 1 million data samples with 18×18 pixelation). As a result, surrogates may perform well on test sets but still fail on unseen geometries, especially those far from the dataset distribution. Another problem related to NN generalization occurs when design space is to be changed,

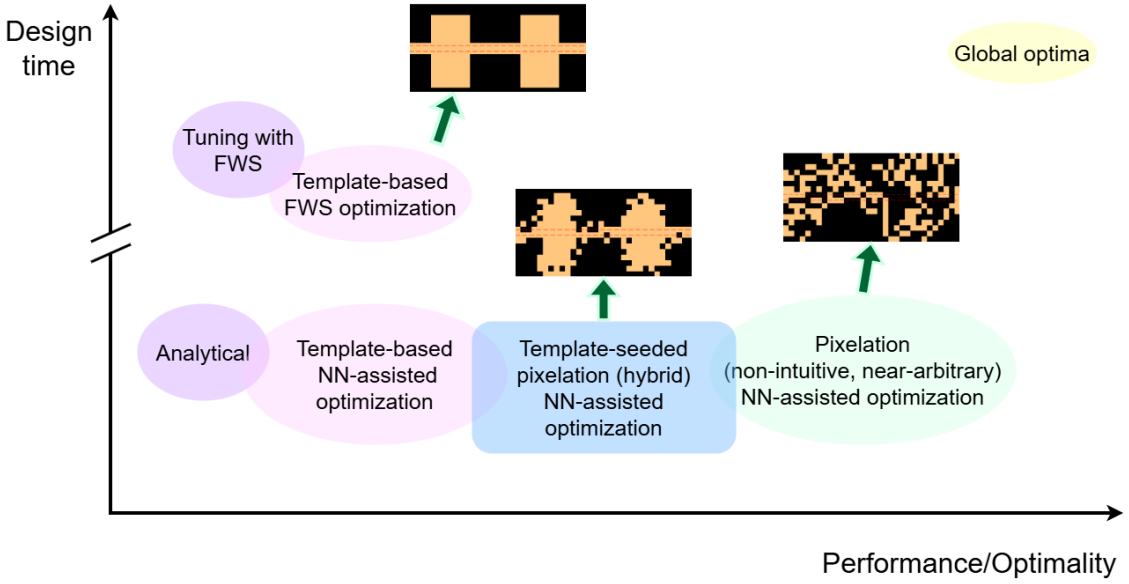

Figure 2.2: Qualitative comparison for different RF design methods

e.g., with different material or pixelation grid. Then the dataset has to be generated from scratch.

Table 2.1: Review on the data sample requirements for different design space.

Paper	Design space	# of FWSs (k)	FWS time (h)	Hardware
Emir et al. [4]	18x18 pixelated	250	14	Princeton HPC, 400 CPU core
		75	96	
		15	18	
Yizhou XU et al. [12]	9-param template	5	72-96	1024-core cluster
Chenhao CHU et al. [9]	18x18x2 hybrid	17	850*	64-core

*850h is an estimation according to the 3min/simulation that mentioned in the paper.

These limitations motivate the search for more generalizable and data-efficient approaches. One of the most promising ways is TL.

2.2 Transfer learning

TL has been applied in several prior works [3, 5, 13, 14]. Existing attempts mainly fall into two categories:

- Multi-fidelity approaches:
  An NN model is first pre-trained on a large set of low-fidelity data generated using simplified geometries or simulation setups (e.g., replacing the substrate with an air box or using a coarse mesh). The model is then fine-tuned using a small number of high-fidelity samples.
- Domain-transfer approaches:
  A previously fine-tuned NN model is further fine-tuned within the same design space but under different material properties or physical dimensions.

In both categories, the pretraining or domain-transfer dataset still relies on FWS, leaving the core computational bottleneck unresolved. In addition, there are limited

quantitative studies on TL for RF inverse design [15], which motivates our work.

2.3 Analytical EM simulators

To fundamentally bypass the FWS data bottleneck, researchers have explored analytical or semi-analytical EM solvers that estimate EM responses using closed-form expressions. Scattering Matrix Method (SMM) was introduced in [16] for 'cross-like' pixel structures, but its applicability is limited by the restricted geometry. To better support 'square-like' pixelation, precomputed numerical Green function (PNGF) [17] and Multiport Analytical Pixel Electromagnetic Simulator (MAPES) [18] provide analytical formulations. MAPES was reported to achieve full-wave-level accuracy with a 600×–2000× speedup over CST across all pixel patterns, although its accuracy depends strongly on the geometric setup and requires careful calibration. PNGF, an approximation-free accelerated EM solver, claims a 4 to 5 order of magnitude speedup relative to FWS. However, its implementation complexity places it beyond the practical scope of this thesis.

2.4 Motivations

Given existing limitations, a highly promising direction is to build a multi-fidelity framework that reduces the reliance on FWS in the RF inverse design workflow, with MAPES serving as the backbone for TL.
A possible structure of such a framework is outlined below:

1. Generate a large-scale low-fidelity dataset using MAPES.
2. Generate a small-scale high-fidelity dataset using FWS.
3. Train a NN-based surrogate model on the MAPES dataset.
4. Fine-tune the pretrained model using the high-fidelity dataset.

In addition, this thesis aims to implement this workflow using entirely open-source tools.

3

Methods & Experiments

To build a specification-to-layout inverse design workflow and validate the benefits of transfer learning, steps follow the flow shown in Fig. 2.1.

1. Define the design goal and target specifications.
2. Consider the design constraints and define the search space.
3. Formulate an objective (fitness) function to score each design.
4. Select an open-source EM simulator for FWS. Benchmark the simulator against *HFSS*.
5. Implement MAPES.
6. Prepare datasets, including high-fidelity and low-fidelity data samples.
7. Train surrogate models with different approaches,(i.e., pure low-fidelity, pure FWS, and multi-fidelity) and compare their performance.
8. Specify an optimizer and explore the design space using the TL-enabled surrogate model.

3.1 Design goal & Specifications

A demonstration objective is set to design a 9 GHz low-pass filter (LPF) on a double-layer PCB. For simplicity, only the magnitudes of S_{11} and S_{21} are considered in the specifications. The target responses are defined as

$$|S_{11}(f \text{ in GHz})| = \begin{cases} 0 & f < 9, \\ 1 & f > 10, \\ f - 9 & 9 \leq f \leq 10 \end{cases}$$

and

$$|S_{21}| = 1 - |S_{11}|$$

as illustrated in Fig. 3.1.
This formulation enforces a near ideal passband up to 9 GHz, followed by a sharp transition region and a fully reflective stopband beyond 10 GHz, providing a clean and interpretable target for inverse design.

3.2 Design space

The EM structure consists of two 50 Ω micro-strip line (MSL) feeds, each with a fixed length of 1.5 mm, placed on opposite sides of the pixelated design region. RO4350B

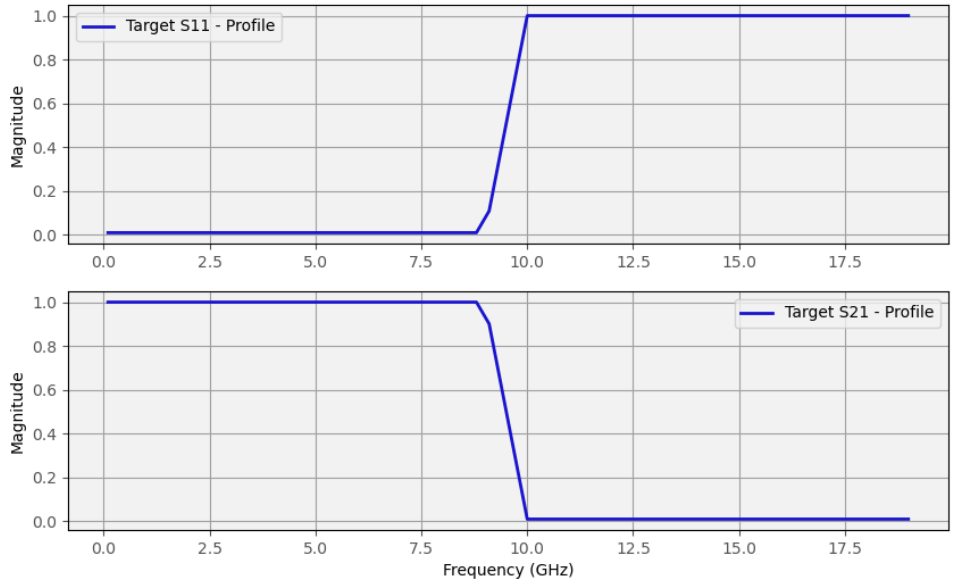

Figure 3.1: Target magnitude profiles for $|S_{11}|$ and $|S_{21}|$ used in the inverse design task. The specification enforces a ideal passband below 9 GHz, a sharp transition region between 9 and 10 GHz, and a fully reflective stopband beyond 10 GHz.

($\epsilon_r \approx 3.48$, $\delta \approx 0.0037$) is selected as the substrate material. A small overlap of 0.08 mm is introduced between adjacent pixels to satisfy practical fabrication constraints.

Note that trade-offs exist among pixel size, grid, NN performance, and computational overhead. For the LPF design considered here, the primary requirement is to ensure sufficient electrical length to realize the necessary inductive and capacitive behavior for cutoff formation, roll-off, and stopband attenuation. A common rule of thumb is that the physical length of the structure should exceed at least one quarter of the effective wavelength at the target frequency.

The pixel cannot be excessively large or excessively small. If the pixels are too large, the structure loses the inductive behavior required for low-pass operation, and a single pixel flip can cause an abrupt change in the EM response. This would introduce strong non-linearity in the mapping from geometry to S-parameters, which makes NN difficult to learn.

Conversely, if the pixels are too small, the geometry becomes more expressive and closer to an arbitrary continuous layout, which is desirable from a design-flexibility perspective. However, the computational cost would increase sharply in two ways. First, maintaining the required electrical length would demand a much larger pixelation grid, which expands both the training data requirement and the size of the design space to be explored. Second, smaller pixels introduce fine geometric details that require a finer mesh in FWS, hence increasing the runtime of each FWS.

Given these trade-offs, the pixel size is chosen as 0.6 times the MSL width plus the 0.08 mm overlap. An 8×8 pixelation grid is adopted to provide sufficient design

geometric flexibility while keeping the simulation time manageable for our experiment. A substrate thickness of 0.508 mm is used to maintain adequate electrical length for the operating frequency range.
In the end, at 9 GHz, the effective wavelength is

$$\lambda_{eff} \approx 19.7 \text{ mm}$$

The width of a 50 Ω MSL on the chosen substrate is

$$W_{msl} \approx 1.1 \text{ mm}$$

The pixel size is

$$s_{px} = 0.6W_{msl} \approx 0.66 \text{ mm} \approx \frac{\lambda_{eff}}{30}$$

With an 8×8 grid, the length of pixelation region is

$$W = 8 \times s_{px} + 0.08 \approx 5.36 \text{ mm}$$

which satisfies

$$W > \frac{\lambda_{eff}}{4}$$

3.3 Objective function

A cost function is first defined as the sum of squared errors between the predicted S-parameters and the target profiles:

$$C = \sum_{f}[(|\hat{S}_{11}| - |S_{11}|)^2 + (|\hat{S}_{21}| - |S_{21}|)^2]$$

where $|S_{11}|$ and $|S_{21}|$ denotes the target responses. And the symbol $\hat{}$ indicates the predicted values. The target profiles are discretized over the frequency range up to 20 GHz, with linear interpolation applied in the transition band between 9 and 10 GHz.
The fitness score is then calculated as

$$\text{fitness} = \frac{1}{1 + C}$$

such that higher fitness values correspond to designs whose predicted responses more closely match the specifications.

3.4 Full-wave EM simulator

EMerge [19] is selected as the full-wave EM solver. It is a Python-based framework that enables the construction of complex EM structures directly through coding and provides a frequency-domain FEM solver suitable for rapid prototyping and automated workflows.

To assess the simulation accuracy of *EMerge*, its results are benchmarked against *HFSS*. A randomly generated 16×32 pixelated pattern on a 254 μm RO4350B substrate is used for this comparison. This configuration intentionally represents a high-resolution and geometrically challenging case, making it well suited for stress testing numerical stability, scalability, and accuracy of the EM solver under demanding conditions. The corresponding EM structure is shown in Fig. 3.2. To import the EM structure into *HFSS*, the footprint is first exported as a `.dxf` file and then reconstructed within *HFSS*.

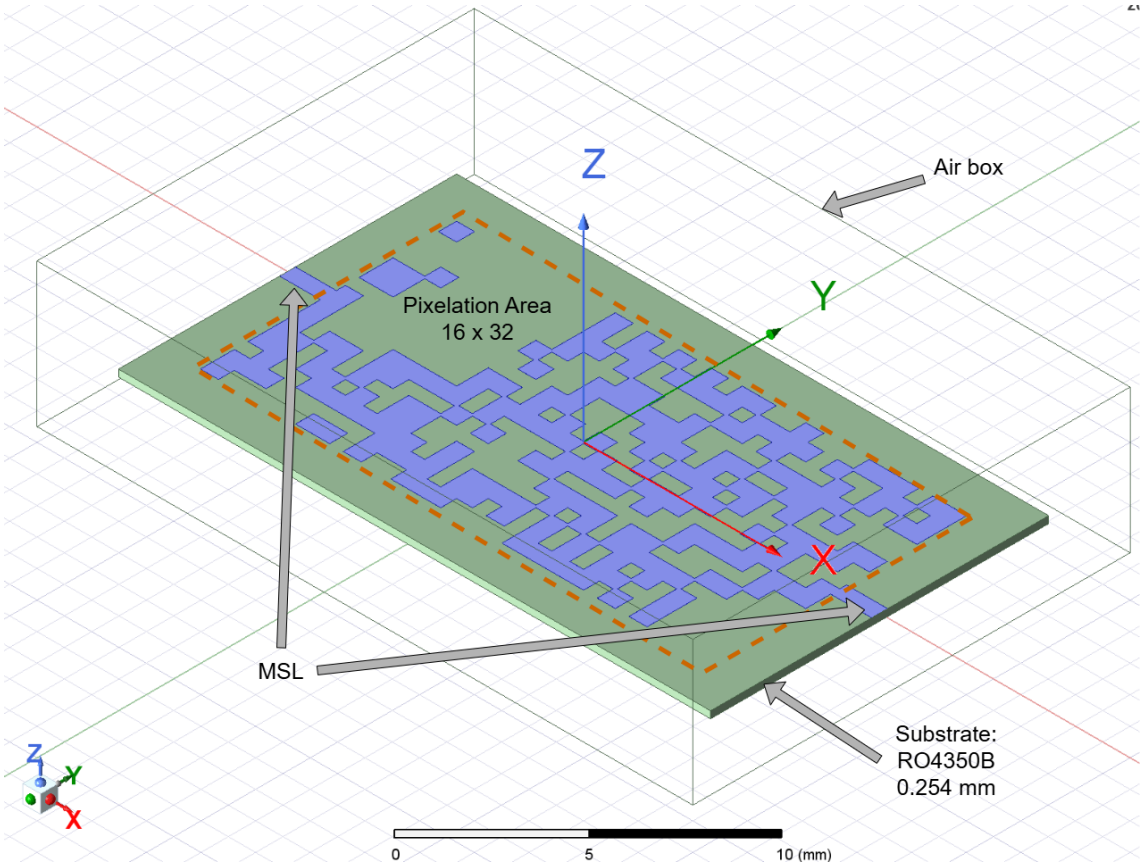

Figure 3.2: A 16×32 pixelated EM structure imported into HFSS for benchmarking the full-wave solver.

Note that for FWS, the meshing strategy affects both the simulation accuracy and the computational cost. To achieve the fastest possible simulations without sacrificing fidelity, different meshing configurations are evaluated. In *EMerge*, two parameters are particularly relevant: the overall mesh resolution and the conductor-edge mesh refinement. The former controls the maximum FEM cell size and is related to the highest frequency considered in the simulation. The latter introduces additional refinement along conductor-dielectric interfaces, to better capture strong field variations in these regions.
Two experiments are conducted:

1. Benchmarking:
   The overall mesh resolution is fixed at 0.33, corresponding to a maximum cell size equal to 0.33 of the effective wavelength at 20 GHz for the chosen

substrate. The conductor-edge refinement is swept from 0.2 mm to 1.0 mm. Both the magnitude and phase of S_{11} and S_{21} are compared with *HFSS* results to assess accuracy.

2. Speedup:
   The conductor-edge refinement is fixed at 1.0 mm, while the overall mesh resolution is varied from 0.2 to 0.4 to evaluate the trade-off between simulation speed and accuracy.

Results will be shown in the first section of the next Chapter.

3.5 Analytical EM simulator

3.5.1 Implementation

The geometric parameter definitions are illustrated in Fig. 3.3. Note that in the original MAPES formulation, no MSL feeds were included as physical I/O ports. To adapt the method for this work, virtual guard ports are introduced between each MSL feed and the adjacent virtual pixels. The physical I/O ports are explicitly assigned to the two MSL feeds located at the two ends of the EM structure, and are implemented as modal ports. All remaining virtual ports are modeled as lumped ports connecting pairs of conductor segments. Small clearances are added around the conducting regions to ensure proper spacing among the vertical, horizontal, and diagonal virtual ports. The following parameters are used:

- Gap between virtual pixels, and between the MSL and virtual pixels: 0.08 mm.
- All clearances: 0.01 mm.
- Virtual pixel size: $0.6W_{msl}$ − gap.
- Virtual diagonal pixel size: 0.5 gap by gap+2 ext, where ext = 1 mm is the vertical extension of the virtual diagonal ports.

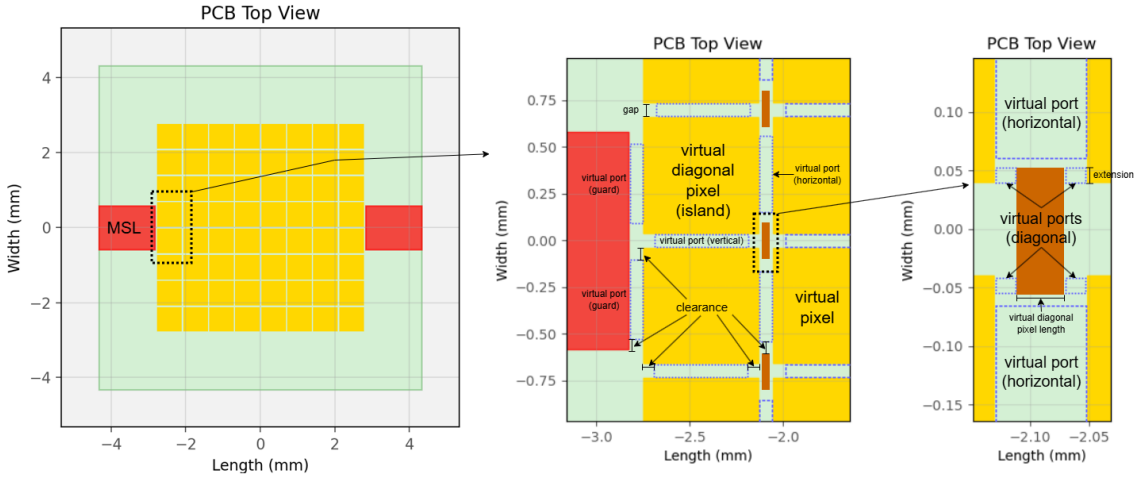

Figure 3.3: Geometry parameter definition used in the adapted MAPES implementation. The left panel shows the overall pixelation region and MSL placement. The middle panel illustrates the arrangement of virtual ports, gaps, and clearances. The right panel details the diagonal port configuration and associated geometric extensions.

3.5.2 Benchmarking

MAPES is validated on an 8×8 grid using RO4350B substrate with thicknesses of 254 μm and 508 μm. Several predefined patterns are tested to assess the accuracy of this analytical simulator. The use of two substrate thicknesses highlights the sensitivity of MAPES to geometric configuration, allowing the benchmarking process to reveal how changes in physical dimensions influence the fidelity of the analytical predictions.
Benchmarking results are shown in the second section of the next Chapter.

3.6 Datasets

NNs are data-driven, and the quality of the dataset plays an important role in determining model performance. In this thesis, dataset construction primarily concerns two aspects: the distribution of pixelation patterns and the formation of labels for each sample.

3.6.1 Patterns

For the 8×8 grid, the number of possible configurations in which two feeds remain electrically connected exceeds 10^{18}. Thus, it is infeasible for any dataset to cover the full design space. In previous work [7], QR-style patterns were generated using

$$p_{metal} \sim \mathcal{N}(0.5,\ 0.15^2) \in [0.2,\ 0.8]$$

where p_{metal} denotes the fraction of metalized pixels and $\mathcal{N}(\mu,\ \sigma^2)$ is the normal distribution with mean μ and standard deviation σ. In this work, to reduce the effective design space and facilitate NN training, a uniform distribution is adopted, with

$$p_{metal} \sim \mathcal{U}(0.4, 0.6)$$

For each generated pattern, the electrical connectivity between the two MSL feeds is verified and any pattern with disconnected feeds is discarded. Furthermore, symmetry is taken into account so that all the mirrored or flipped variants of a pattern are removed to avoid redundant samples.

3.6.2 Labels

Although only $|S_{11}|$ and $|S_{21}|$ are used in the objective function, each label contains the real and imaginary parts of S_{11}, S_{21}, and S_{22} to maintain generality for two-port passive EM components. Thus, every frequency point is associated with 6 S-parameter values. As discussed in section 3.3, the upper frequency limit is set to 20 GHz. Since very low frequencies (e.g., DC) are unnecessary, the frequency range is defined as

$$f \in [0.1,\ 20] \text{ GHz}$$

To emphasize the roll-off characteristics, a non-uniform sampling scheme is adopted. The sampling steps are 0.3 GHz and 1 GHz, from 0.1 to 10 GHz and from 10

to 20 GHz, respectively. This results in a total of 43 frequency points, yielding $43 \times 6 = 258$ values per label.

3.6.3 Scale

In previous work [7], a 13×13 pixelation grid required approximately 60k samples to achieve satisfactory surrogate model performance. In this thesis, the design space is smaller due to the 8×8 grid and the pattern distribution is also narrower. As a result, a total of 22k high-fidelity samples is sufficient. In addition, 50k low-fidelity samples are generated using MAPES. Among the 22k high-fidelity samples, 4k were reserved exclusively for evaluation.

3.7 Surrogate models

3.7.1 Neural network

In previous works [3, 4, 5, 7], CNNs with large convolution kernels were employed, and down samplings were applied only in the final stages of the network. These design choices ensure a sufficiently large receptive field, which enables the model to capture global EM interactions rather than only short-range coupling. This is essential because flipping a single pixel can influence the EM field distribution far from its physical location.
However, such NN architecture typically contain a large number of parameters, which limits both the training speed and inference efficiency. To address this, the proposed NN replaces large kernels with dilated convolutions to expand the receptive field more efficiently. In addition, residual modules are adopted to facilitate the training of deeper CNN and improve gradient flow. The architecture is shown in Fig. 3.4.
The network predicts S_{11}, S_{21}, and S_{22}, consistent with the label definition in the previous section. A `tanh` activation is applied to enforce the numerical constraint that both the real and imaginary parts of the S-parameters lie within the interval $[-1,\ 1]$.

3.7.2 Training strategy

The training configuration is summarized below:

- Data augmentation: on-the-fly transforms, including horizontal and vertical flips, where the pattern is mirrored about its central vertical or horizontal axis. This strategy increases the diversity of configurations encountered during training and improves the generalizability of the surrogate model. The term *on-the-fly* indicates that the transforms are applied during batch construction rather than pre-generating augmented samples, thereby avoiding additional storage overhead. The augmentation settings are as follows:
  - horizontal flip probability: $p_{flip,h} = 0.5$
  - vertical flip probability: $p_{flip.v} = 0.5$

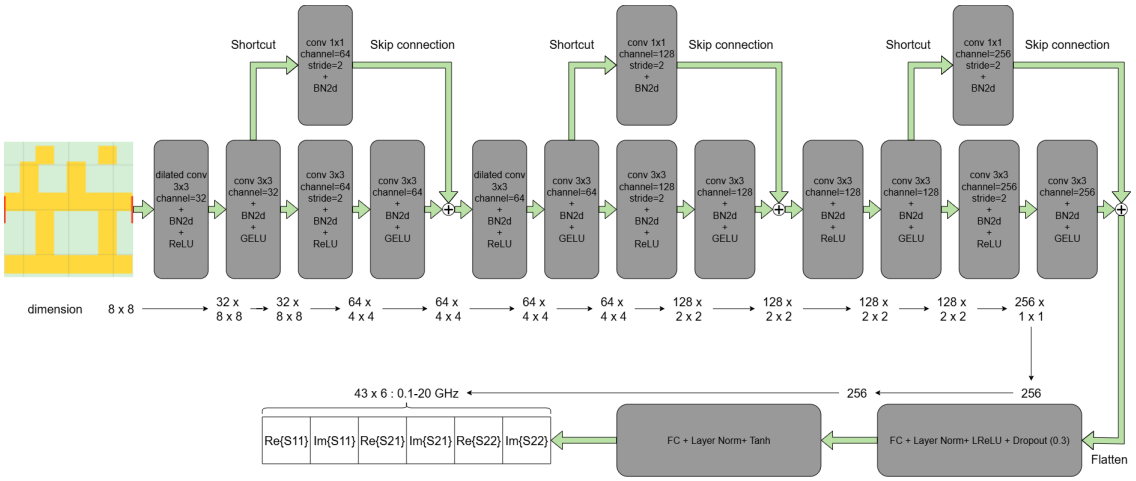

Figure 3.4: NN architecture used for surrogate model. The input 8×8 pattern is processed through a sequence of dilated and standard convolutional blocks with residual connections and stride-based down-sampling. The feature maps progress from 32 to 256 channels before being flattened and passed through two fully connected layers.

For horizontal flips, the ports are effectively exchanged. Therefore, S_{11} and S_{22} should be swapped, while S_{21} remains unchanged for reciprocal two-port passive structures.

- Loss function: L1 (mean absolute error)
- Optimizer: `AdamW` with a weight decay of 0.01
- Initial learning rate: 10^{-3}
- Scheduler: `CosineAnnealingLR` with a minimum learning rate of 10^{-4}
- Batch size: 1024
- Number of epochs: dependent on dataset scale

For TL, all the convolutional layers are frozen, and only the parameters of the last two fully connected layers are updated. The learning rate for fine-tuning is fixed at 10^{-4}.

3.7.3 Performance study

The following groups of NN surrogate models are trained to quantitatively assess the functionality of TL.

- Baseline models:
  These models are trained using different amounts of pure FWS data. They establish the reference relationship between the size of the high-fidelity training set and the achievable prediction accuracy.
- Surrogate models for analytical EM simulations:
  These models are trained on varying numbers of low-fidelity samples. They serve two purposes: to evaluate the accuracy of MAPES and to determine how many low-fidelity samples are required to maximize the overall efficiency in data preparation.
- Transfer-learned models:

These models are initialized from the low-fidelity surrogates and fine-tuned using different amounts of pure FWS data. By comparing their accuracy with that of the baseline models, the reduction in high-fidelity data requirements can be quantified.

Table. 3.1 summarizes the detailed training configurations for the baseline models, the MAPES-based surrogates, and the transfer-learned models.

Table 3.1: Training configurations for the three categories of surrogate models.

Dataset (FWS) scale (k)	0.1	0.5	1	4	10	15
Training epochs	20000	5000	3000	2000	1000	800

(a)

Baseline models trained on different amounts of high-fidelity FWS data.

Dataset (MAPES) scale (k)	0.1	0.4	1	4	10	40
Training epochs	20000	5000	3000	2000	1000	800

(b)

Surrogate models trained on varying numbers of low-fidelity MAPES samples.

Dataset (FWS) scale (k)	0.1	0.5	1	2	10
Fine-tune epochs	20000	5000	3000	2000	1000

(c)

Transfer-learned models fine-tuned using different scales of high-fidelity data.

The average root mean square error (RMSE) is used to evaluate the performance of NN surrogate models. For each sample, the RMSE is defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{3N_f}\sum_{f}\sum_{i=11,21,22}(|\hat{S}_i| - |S_i|)^2}$$

where the symbol $\hat{}$ denotes the predictions from surrogate models.

This metric provides a standardized basis for comparison with existing literature. For example, a prior study [7] on LPF design reported an RMSE of approximately 0.06. In this context, an evaluation RMSE below 0.1 can be considered sufficiently accurate for optimization tasks.

The performance study results are shown in section 4.3.

3.8 Optimizer

A GA is used as the optimizer in the inverse-design demonstration. The algorithm is intentionally kept as simple as possible, as the primary objective is to rapidly validate the fine-tuned surrogate model obtained through TL. The optimization procedure is summarized below.

1. Initialize a population of 1000 patterns (individuals).
2. Predict the S-parameters of all individuals using the surrogate model.

3. Compute the fitness of each individual and identify the best one.
4. Build the next generation using 500 newly generated random patterns, the best individual, and 499 mutated variants of the best individual.
5. Repeat from Step 3 for up to 300 iterations.

Mutation is implemented as an independent pixel-flip operation applied to each pixel with a fixed probability of 0.01.

4

Results & Discussion

4.1 Full-wave EM simulator

Fig. 4.1 shows the benchmarking results comparing *EMerge* with *HFSS*. Refining the mesh near the conductor edges from 1 mm down to 0.2 mm does not yield any noticeable improvement in accuracy. In contrast, the simulation time increases dramatically, from less than 2 minutes to roughly 1 hour. Therefore, the mesh refinement parameter is fixed at 1.0 mm for all subsequent simulations.

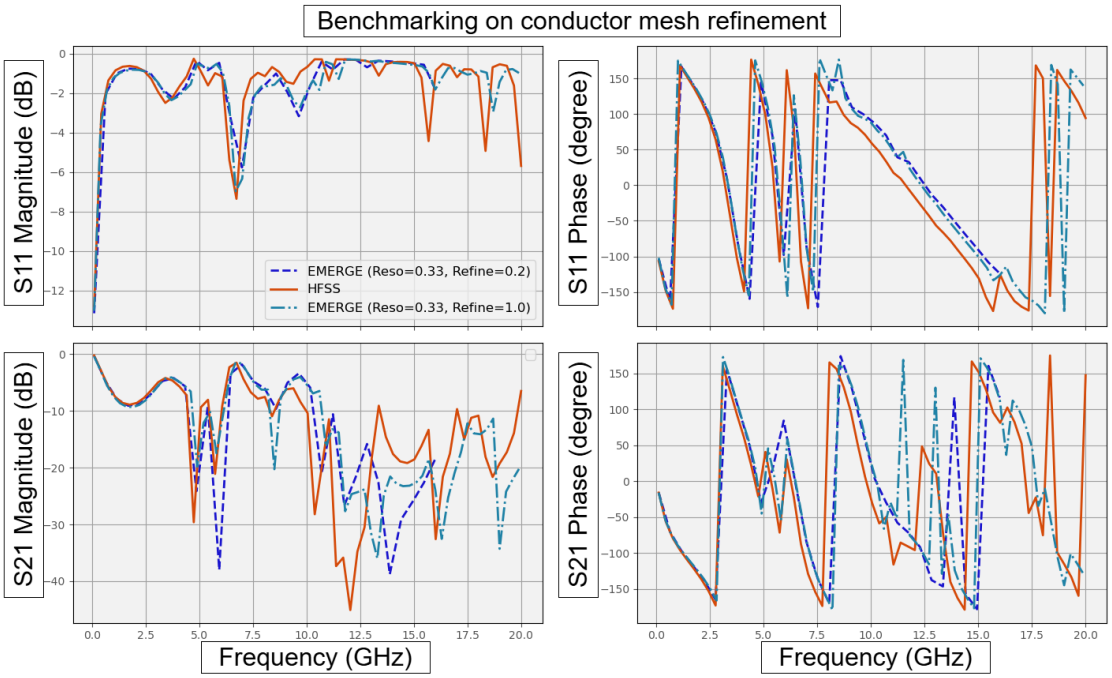

Figure 4.1: Benchmarking results comparing simulation results from *EMerge* and *HFSS*. '*Reso*' represents the overall mesh resolution. '*Refine*' values specify the mesh size around the edges of conductor.

Fig. 4.2 summarizes the speed-accuracy trade-off when varying the overall mesh resolution. The S-parameter magnitude and phase obtained with resolution of 0.20, 0.30, and 0.33 are nearly identical across the entire frequency range. Since finer meshes offer no observable accuracy benefit, a resolution of 0.33 is selected to minimize computational overhead.
The final EM simulation settings therefore use a overall mesh resolution of 0.33 together with a conductor edge refinement of 1.0 mm.

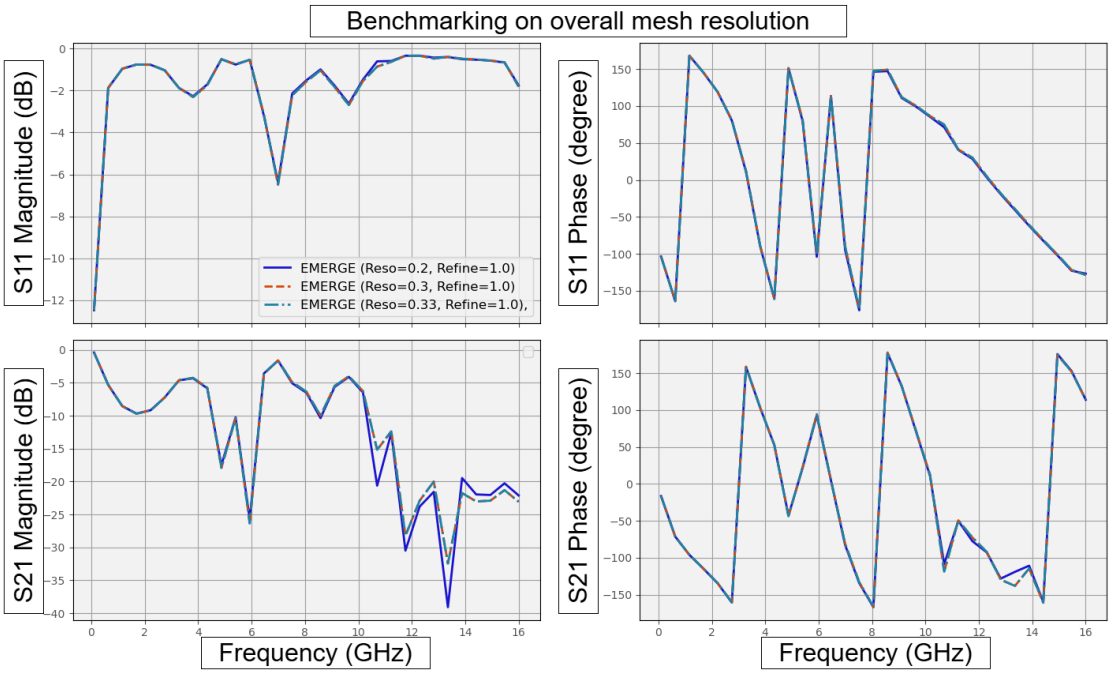

Figure 4.2: FWS results comparison with different overall resolution setup.

4.2 Analytical EM simulator

Fig. 4.3, compares the MAPES estimations with FWSs results for two patterns on a 254 μm RO4350B substrate. Across all three S-parameters (S_{11}, S_{21}, and S_{22}), the analytical EM simulator achieves good accuracy under this geometric configuration. However, when the RO4350B thickness is increased to 508 μm while keeping all other geometric and port settings unchanged, MAPES exhibits large prediction errors, as shown in Fig. 4.4. This degradation confirms that MAPES does not inherently guarantee high accuracy across different configurations. Instead, its performance is highly sensitive to the numerous tunable geometric parameters involved in the algorithm implementation, as previously discussed in Chapter 3.
Despite the noticeable errors observed for the 508 μm RO4350B case, no additional calibration was applied to MAPES. As demonstrated in the next section, the analytical simulator still provides a sufficiently good backbone for TL.

4.3 Transfer learning

To evaluate the effectiveness of transfer learning, we compare the performance of surrogate models trained under different data-scale configurations. Specifically, models trained solely on low-fidelity MAPES data ('MAPES Surrogate') are examined alongside baseline models trained on varying amounts of high-fidelity FWS data ('EM Surrogate'). This comparison reveals how much high-fidelity data can be saved when a pretrained low-fidelity surrogate is used as the starting point for fine-tuning. Fig. 4.5 shows the comparison results. The MAPES-only surrogate converges to an RMSE slightly below 0.2, which is comparable to the performance of an NN trained

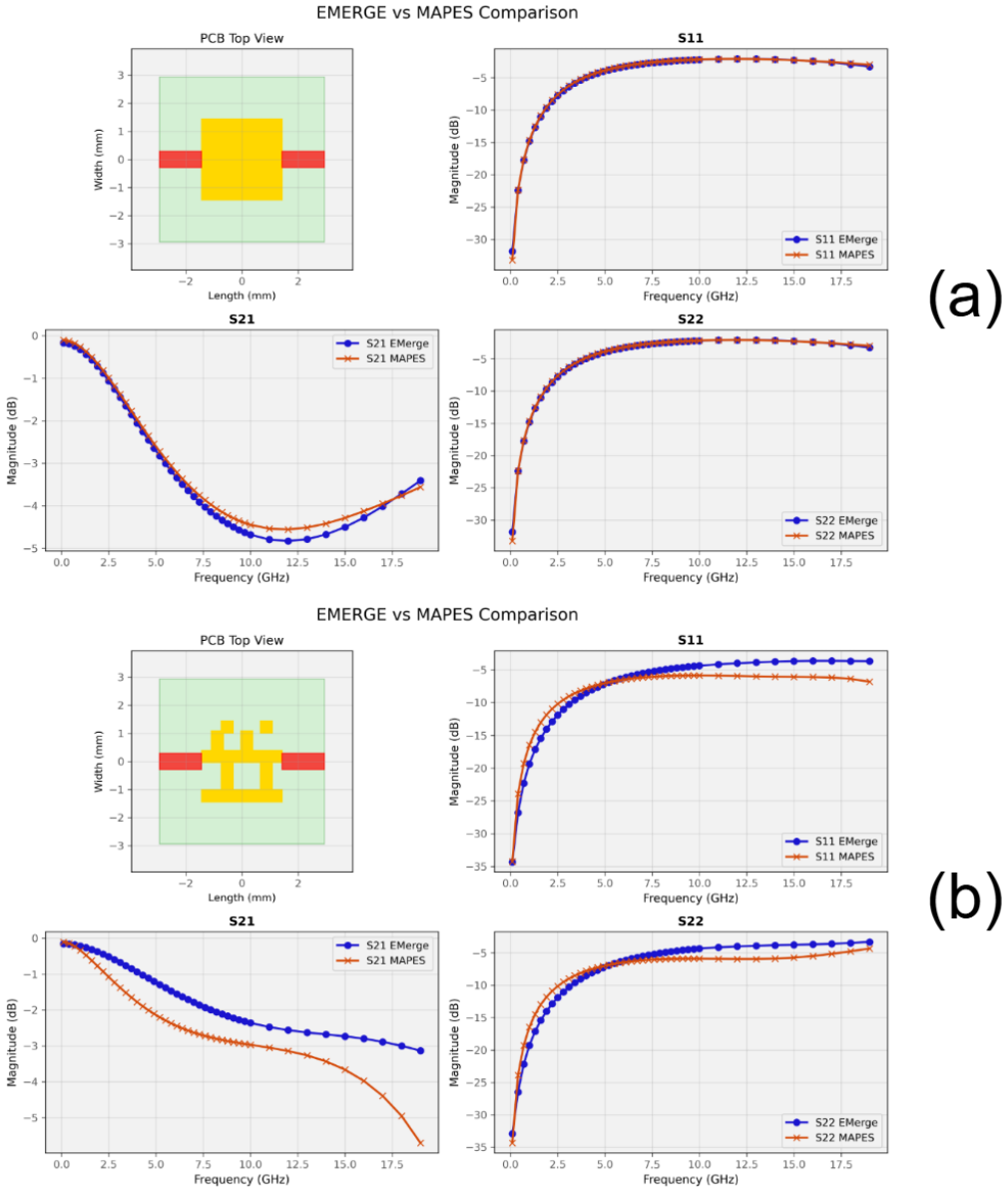

Figure 4.3: Comparison between MAPES and FWSs (*EMerge*) for two 8×8 pixelated patterns on a 254 μm RO4350B substrate. (a) and (b) show two different PCB layouts. For each pattern, the magnitude responses of S_{11}, S_{21}, and S_{22} are plotted. Across all three S-parameters, MAPES closely matches the full-wave results over the entire frequency range.

on 1k high-fidelity samples. This demonstrates that, despite the simulation error discussed in Section 4.2, MAPES dataset still provides a sufficiently strong feature backbone for surrogate modeling by capturing global EM trends.

Applying TL fundamentally shifts the training curve. Starting with a model pre-trained on 40k low-fidelity samples, fine-tuning with merely 100 high-fidelity FWS samples already yields an immediate RMSE reduction of approximately 0.1. This demonstrates that even a minimal amount of high-fidelity data is sufficient to substantially improve the surrogate model once a strong low-fidelity backbone is avail-

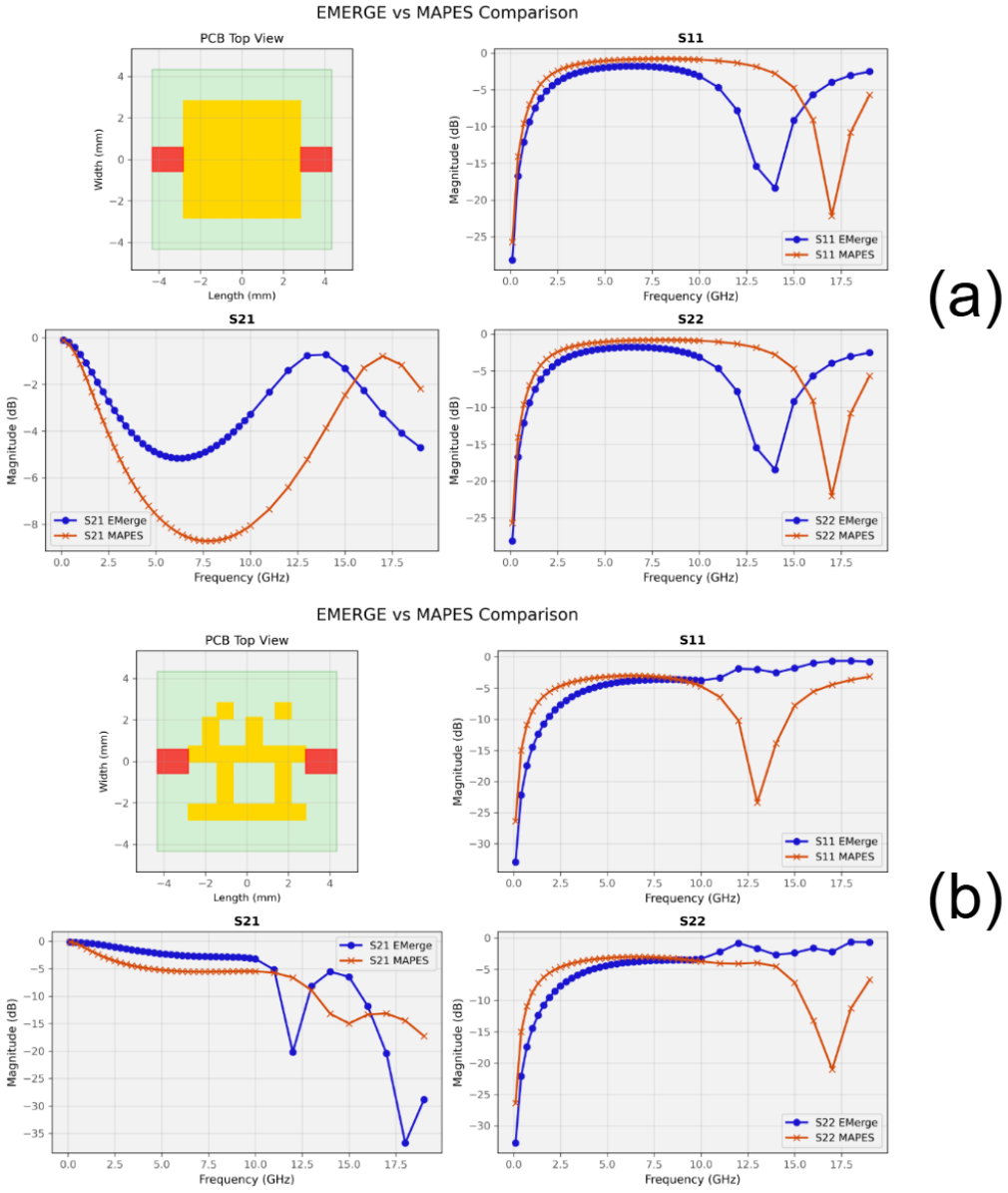

Figure 4.4: Comparison between MAPES and FWSs for two 8×8 patterns on a 508 μm RO4350B substrate. (a) and (b) show two PCB layouts together with the corresponding magnitude responses of S_{11}, S_{21}, and S_{22}. The plots illustrate how the analytical EM simulator behaves under a thicker substrate configuration.

able.

Fig. 4.6 further explores how the scale of the MAPES pretraining dataset influences the final fine-tuned model. The results show that the initial RMSE before applying TL depends heavily on the amount of low-fidelity data used during pretraining. As the MAPES dataset increases from 100 to 40k samples, the starting RMSE progressively approaches the intrinsic error floor of the MAPES. Overall, a larger and more diverse low-fidelity dataset enables the network to learn a more generalizable physical representation, thereby reducing the amount of high-fidelity data required during the TL stage.

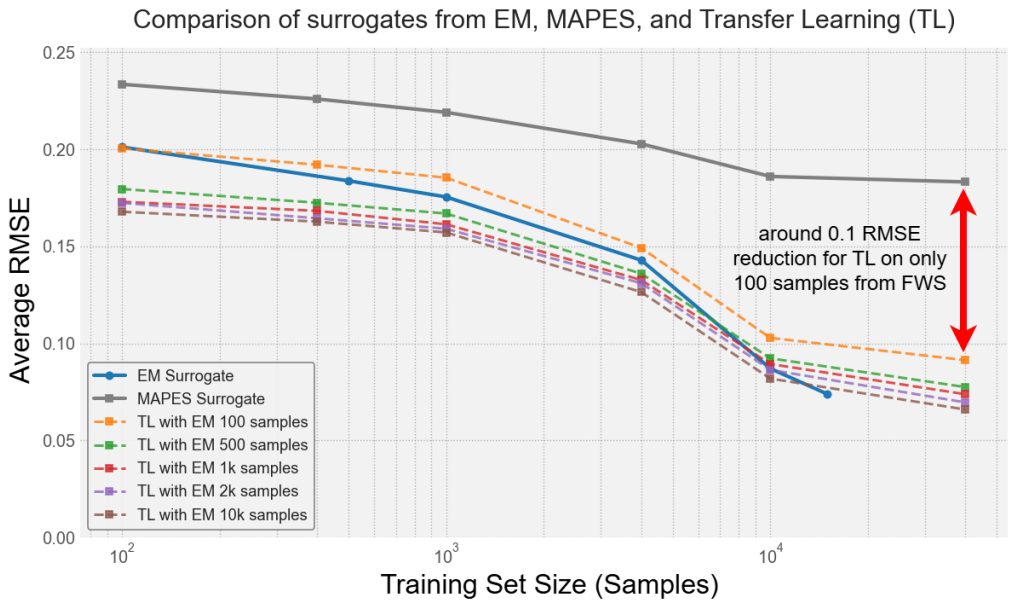

Figure 4.5: Learning curve comparison between conventional surrogate training and TL. The MAPES-only surrogate and the EM-only surrogate are shown alongside TL models fine-tuned with different amounts of high-fidelity FWS data.

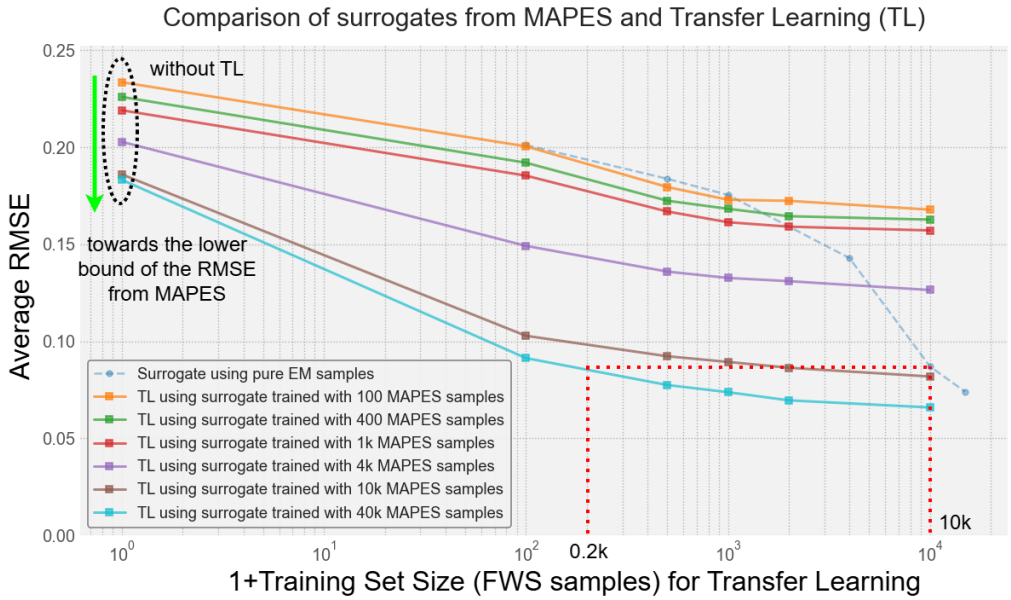

Figure 4.6: Effect of MAPES pretraining dataset size on TL performance. The curves show TL models initialized with MAPES surrogates trained on datasets ranging from 100 to 40k samples, together with a baseline trained purely on high-fidelity FWS data.

Ultimately, comparing the high-fidelity sample requirements highlights the substantial computational savings enabled by TL. To surpass the target RMSE threshold of 0.1, a conventional surrogate model requires approximately 10k FWS samples. In contrast, a TL-assisted model pretrained on 40k MAPES samples needs only 200

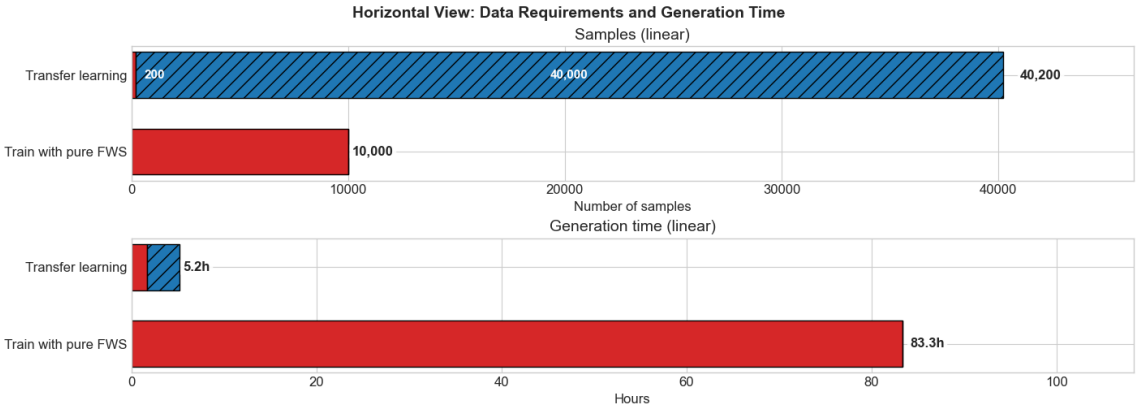

Figure 4.7: Comparison of data and generation time requirements for achieving the same prediction accuracy using pure FWS-based training versus TL. The TL approach combines 40k low-fidelity MAPES samples with only 200 high-fidelity FWS samples, whereas the conventional method requires 10k FWS samples.

FWS samples to reach comparable accuracy. As shown in Fig. 4.7, this reduces the high-fidelity data requirement by a factor of 50, and lowers the total dataset generation time from 83.3 hours to just 5.2 hours, measured on a single workstation equipped with an Intel Core i9-14900K (24 cores), with single process.

4.4 Inverse design

Using the transfer-learned model pretrained on 40k low-fidelity samples and fine-tuned with 200 high-fidelity FWS samples, the surrogate achieves an RMSE of 0.081. Based on this model, 300 rounds of design-space exploration were performed using the objective function and GA introduced in the previous chapter. The optimization took around 10 min. One representative layout is shown in Fig. 4.8.
The resulting pattern presents a geometry that is not constrained by conventional human-designed topologies. However, the NN surrogate does not perfectly reproduce the full-wave responses for this particular layout. The errors are unexpectedly large between the NN-predicted and simulated S-parameter curves. This mismatch suggests that the inverse-designed pattern may lie partially outside the distribution covered by the training data, either due to the random flipping strategy exploring unconventional configurations or limitations in the surrogate's generalizability. Consequently, while the model is sufficiently accurate to guide the search toward promising regions of the design space, additional refinement or active learning would be required to achieve fully reliable predictions for such out-of-distribution structures. Another point is that the roll-off was not satisfactory. The reasons might be two folds. First, as just discussed, the surrogate model failed to generalize beyond the dataset distribution, which led to a large prediction error. Second, from the perspective of filter theory, the short electrical length might prevent the filter to achieve high order.

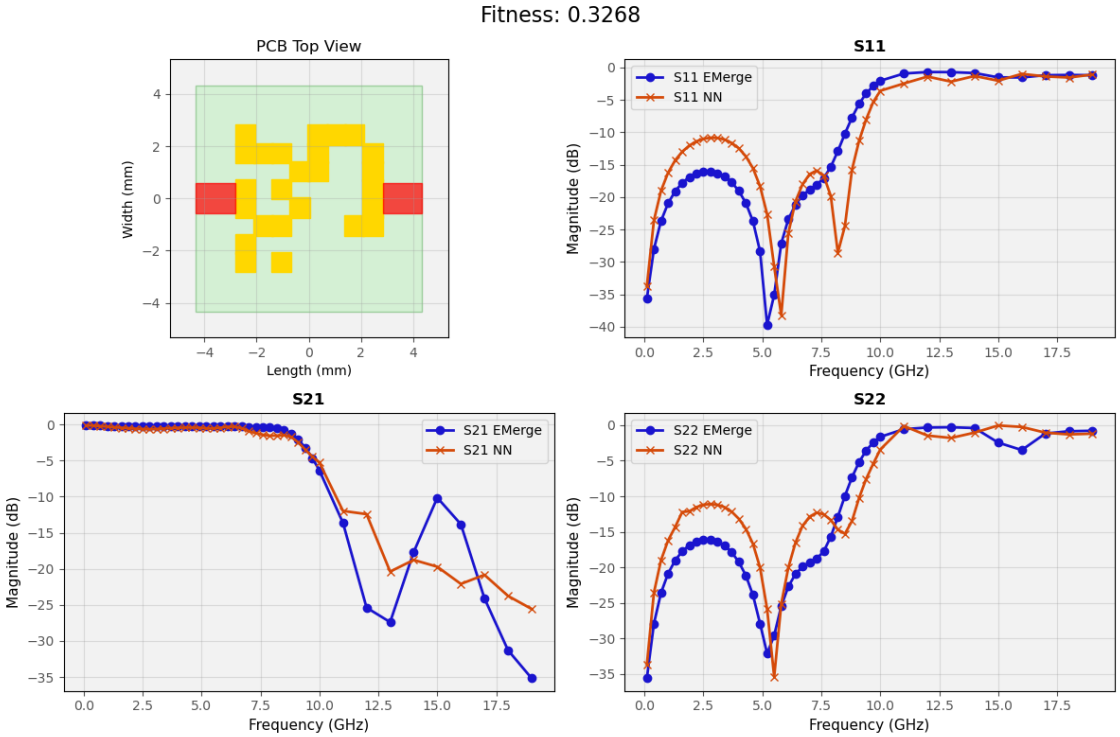

Figure 4.8: Layout obtained from the inverse design demonstration, together with the S-parameter comparison between the transfer-learned surrogate and the full-wave simulator (*EMerge*).

5

Conclusion & Future work

5.1 Conclusion

This work presented a multi-fidelity surrogate modeling framework for fast LPF design, combining analytical EM simulations, high-fidelity full-wave solvers, and TL. An analytical EM simulator based on MAPES was adapted to pixelated structures with MSL feeds, enabling rapid low-fidelity data generation. Although the accuracy of MAPES is not guaranteed due to its sensitivity to geometric and meshing parameters, the results demonstrated that it provides a sufficiently strong backbone for surrogate modeling.

A high-fidelity dataset was generated using *EMerge*, and a CNN with residual modules was trained to predict S-parameters. Benchmarking showed that the surrogate model trained purely on high-fidelity data requires a large dataset to achieve acceptable accuracy. By contrast, TL significantly reduced the need for high-fidelity samples. With 40k low-fidelity samples and only 200 high-fidelity samples, the transfer-learned model achieved an RMSE of 0.081, which is comparable to models trained with 50 times more high-fidelity data. This highlights the substantial efficiency gains enabled by transfer learning, reducing the high-fidelity data-generation effort by roughly 16-fold.

Finally, an inverse-design demo using a simple GA validated the practical usefulness of the surrogate model. The transfer-learned surrogate successfully guided the optimizer to produce a 9 GHz LPF design, which confirmed the feasibility of the proposed multi-fidelity workflow.

Overall, this work shows that combining analytical EM models with transfer learning provides a highly efficient path toward fast, data-driven microwave component design, dramatically reducing the reliance on expensive FWSs.

5.2 Future work

Several directions can further enhance the proposed framework:

1. Improved analytical EM modeling:
   The current MAPES implementation involves multiple tunable parameters, and its accuracy is sensitive to both geometric configuration and meshing strategy. Developing a systematic calibration procedure, or an automated parameter-search algorithm, could substantially improve its predictive accuracy and robustness across different structures. A more accurate analytical EM simulator would, in turn, further reduce the amount of FWS data required

for effective TL, thereby improving the overall efficiency of the multi-fidelity workflow.

2. Adaptive optimizer aware of dataset distribution:
   To address the issue that the NN surrogate model may exhibit poor prediction accuracy when queried outside the distribution of the training data, it is worthwhile to develop an optimizer that is explicitly aware of the dataset distribution. Instead of freely exploring the entire design space, such an optimizer would adaptively guide the search toward regions where the surrogate model is reliable, for example by incorporating density estimation, uncertainty prediction, or latent-space constraints. This strategy could prevent the optimizer from generating out-of-distribution candidates, thereby improving both the stability of inverse design and the overall quality of the obtained solutions.
3. Expansion of pixelation scale:
   Achieving truly arbitrary geometries requires a finer pixelation resolution, which inevitably leads to an expansion of the design grid. Increasing the pixelation scale would allow the surrogate model to represent more detailed electromagnetic features and support higher-order filter behaviors. However, this also increases the dimensionality of the design space and the computational cost of data generation. Future work may explore hierarchical pixelation, multi-resolution representations, or dimensionality-reduction techniques to enable finer geometric detail without excessively increasing the model complexity.
4. Extension to multilayer or 3D structures:
   The current study focuses on double-layer PCB structures. Extending the proposed framework to multilayer or fully 3D components would significantly broaden its applicability to practical RF systems, where vertical coupling, substrate stacking, and enclosure effects often play critical roles. Such an extension would require adapting both the analytical EM modeling and the surrogate architecture to handle increased geometric complexity and additional degrees of freedom. Nonetheless, achieving this capability would enable the framework to support a wider range of RF devices.

Acknowledgment

I would like to sincerely thank my supervisor, Dr. Yin Zeng, whose guidance, patience, and expertise have shaped this thesis from the very beginning. Your mentorship has been invaluable both academically and personally.
I am also grateful to Prof. Jan Grahn for the feedback and support.
Special thanks go to my colleagues at ScalinQ AB for their technical assistance, discussions, and support throughout this project.
Lastly, I want to express my heartfelt gratitude to my family and friends. Their encouragement and belief in me have been a constant source of motivation.

Bibliography

[1] C. Liu and D. Chitnis, "EEsizer: LLM-Based AI Agent for Sizing of Analog and Mixed Signal Circuit," 2025. [Online]. Available: https://arxiv.org/abs/2509.25510

[2] Y. Lai, S. Poddar, S. Lee, G. Chen, M. Hu, B. Yu, P. Luo, and D. Z. Pan, "AnalogCoder-Pro: Unifying Analog Circuit Generation and Optimization via Multi-modal LLMs," 2025. [Online]. Available: https://arxiv.org/abs/2508.02518

[3] Z. Liu, E. A. Karahan, and K. Sengupta, "Deep Learning-Enabled Inverse Design of 30–94 GHz Psat,3dB SiGe PA Supporting Concurrent Multiband Operation at Multi-Gb/s," *IEEE Microwave and Wireless Components Letters*, vol. 32, no. 6, pp. 724–727, 2022.

[4] E. A. Karahan, Z. Liu, and K. Sengupta, "Deep-Learning-Based Inverse-Designed Millimeter-Wave Passives and Power Amplifiers," *IEEE Journal of Solid-State Circuits*, vol. 58, no. 11, pp. 3074–3088, 2023.

[5] E. A. Karahan, Z. Liu, A. Gupta, Z. Shao, J. Zhou, U. Khankhoje, and K. Sengupta, "Deep-learning enabled generalized inverse design of multi-port radio-frequency and sub-terahertz passives and integrated circuits." *Nat Commun 15, 10734 (2024)*, 2024.

[6] N. Tran, C. Hao, A. Stameroff, A.-V. Pham, and Y. Chen, "Alpha-RF: Automated RF-Filter-Circuit Design with Neural Simulator and Reinforcement Learning," 2026. [Online]. Available: https://arxiv.org/abs/2603.00104

[7] M. Sjödin, O. Talcoth, H. Chang, H. Zhou, and K. Andersson, "Comparison of Circuit Models for ML-Assisted Microwave Circuit Design," *IEEE Journal of Microwaves*, vol. 5, no. 6, pp. 1358–1369, 2025.

[8] C. Chu, E. Liu, Y. Liu, B. Lin, M. Eleraky, B. Abdelaziz, M. Ghorbanpoor, T.-Y. Huang, A. Wang, and H. Wang, "Deep Learning-Assisted RFIC Design With Dual-Metal-Layer Passive Matching Networks: A 15-22 GHz CMOS PA for 6G in 22nm FDX+," in *2025 16th German Microwave Conference (GeMiC)*, 2025, pp. 76–79.

[9] C. Chu, J. Xu, Y. Liu, J. Zeng, A. Wang, T. Torii, S. Shinjo, K. Yamanaka, and H. Wang, "AI-Assisted Template-Seeded Pixelated Design for Multi-Metal-Layer High-Coupling EM Structures: A Ku-Band 6G FR3 PA in 22nm FDX+," in *2025 IEEE/MTT-S International Microwave Symposium - IMS 2025*, 2025, pp. 922–925.

[10] E. A. Karahan, Z. Liu, and K. Sengupta, "IMS Deep Learning Enabled Generalized Synthesis of Multi-Port Electromagnetic Structures and Circuits for

mmWave Power Amplifiers," in *2024 IEEE/MTT-S International Microwave Symposium - IMS 2024*, 2024, pp. 184–187.

[11] J. Lee, W. Jia, B. S. Rodriguez, and J. S. Walling, "Pixelated RF: Random Metasurface Based Electromagnetic Filters," in *2023 21st IEEE Interregional NEWCAS Conference (NEWCAS)*, 2023, pp. 1–5.

[12] Y. Xu. and Others, "Open-source Automatic AI-driven Ultra-broadband Passive-active Co-design Workflow: A T-Coil Peaked Wireline Driver Example," https://github.com/ETH-IDEAS/AI4TCOIL/blob/main/TCoil_Dataset_Generator_and_Training.ipynb, 2026, gitHub repository.

[13] C. Chu, Y. Mao, and H. Wang, "Transfer Learning Assisted Fast Design Migration Over Technology Nodes: A Study on Transformer Matching Network," in *2024 IEEE/MTT-S International Microwave Symposium - IMS 2024*, 2024, pp. 188–191.

[14] H. He, Y. Xu, L. Xia, Y. Hu, F. Cai, and T. Chi, "MOTIF-RF: Multi-template On-chip Transformer Synthesis Incorporating Frequency-domain Self-transfer Learning for RFIC Design Automation," in *2026 31st Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2026, pp. 1138–1144.

[15] M. Prousali and S. Tsitsos, "Surrogate-Based EM Design of RF and Microwave Components: A Systematic Review of Workflow Roles, Inverse Design, Multifidelity, and Active Learning," *Sensors*, vol. 26, no. 8, 2026. [Online]. Available: https://www.mdpi.com/1424-8220/26/8/2504

[16] N. Morrison, X. He, T. Xie, and E. Y. Ma, "Rapid inverse design of microwave devices with scattering matrices," *Phys. Rev. Appl.*, vol. 24, p. 044100, Oct 2025. [Online]. Available: https://link.aps.org/doi/10.1103/qlfy-4qhw

[17] J.-H. Sun, M. Elsawaf, Y. Zheng, H.-C. Lin, C. W. Hsu, and C. Sideris, "Near real-time full-wave inverse design of electromagnetic devices," *Nature Communications*, 2026.

[18] J. Rao, Y. Liu, J. Zhang, Z. Ming, T. Qiao, Y. Zhang, C. Y. Chiu, H. Wang, and R. Murch, "Multiport Analytical Pixel Electromagnetic Simulator (MAPES) for AI-assisted RFIC and Microwave Circuit Design," 2025. [Online]. Available: https://arxiv.org/abs/2511.21274

[19] https://www.emerge-software.com/, open source EM simulator, EM tools for Python.