End-to-End Machine-Learning Framework for Electromagnetic Inverse Design: From Practical Constraints to Optimized Structures

Zhao Zhou, Zhaohui Wei, *Member, IEEE*, Jian Ren, *Member, IEEE*, Yingzeng Yin, *Member, IEEE*,
Jinna Li, *Senior Member, IEEE*, Tse-Tin Chan, *Member, IEEE*

***Abstract*—Existing machine learning (ML)-based inverse design methods for electromagnetic (EM) structures encounter two major challenges: impractical input requirements and non-uniqueness effects. This paper introduces an end-to-end EM design framework based on a modified conditional variational autoencoder (MCVAE), which directly maps practical design constraints to optimized structure configurations. The proposed framework incorporates a modified decoder that interprets practical constraints, such as desired operating frequency, bandwidth, maximum allowable value for reflection coefficients, and transmission phase, into detailed EM responses over the target frequency band. A modified encoder mitigates the effects of non-uniqueness common in inverse EM design, where multiple structures yield similar responses, by integrating Gaussian noise for robust latent space exploration and employing a forward-model-based loss function to enforce structural accuracy, thereby enhancing output reliability and model performance. To validate the effectiveness of the proposed method, multiple practical implementations are presented: a linear-to-circular polarization converter design, a Fourier-phased metasurface design, a slot-line filter design, a loop polarization converter design, and a polarization converter design with double and triple-expanded parameter ranges. Compared to traditional optimization-based methods, the proposed method significantly improves design efficiency while maintaining high accuracy. This method offers a generalized framework for end-to-end EM design, bridging practical constraints to optimized structure realizations.**

***Index Terms*—Electromagnetic design, end-to-end, inverse design, machine learning, non-uniqueness, practical constraints.**

I. INTRODUCTION

ELECTROMAGNETIC (EM) design aims to obtain an EM structure that generates the desired EM response. Conventional design methods often depend on trial-and-error approaches due to the lack of a quantitative mapping between structures and their corresponding responses.

Trial-and-error methods often require numerous iterations to achieve a structure that meets the desired response. Each iteration relies on full-wave simulation and the designers' reasoning ability to evaluate and update the current structure. The structure is iteratively modified to approach the desired performance. Optimization algorithm-based methods replace this manual reasoning with optimization algorithms, such as genetic algorithms (GA) [1]–[3] and particle swarm optimization (PSO) [4]–[7]. The machine learning (ML) forward-model-based methods further replace full-wave simulation with forward surrogate models to evaluate structures' responses [8]–[10], [12]–[14], [16]–[18], [20]–[24]. Trained on historical data, these surrogate models learn the relationships between structures and their responses. In addition, the ML generative model-based method utilizes generative models to intelligently initialize and refine structures [25]–[28]. By learning the characteristics of existing structures, generative models can mimic human designers by initializing designs intelligently rather than randomly. Generative models refine structures by adjusting compressed feature vectors in a latent space.

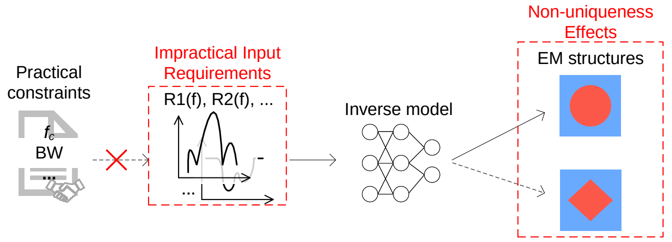


Fig. 1. Typical ML-based inverse design methods encounter two key challenges: impractical input requirements and non-uniqueness effects

In contrast with trial-and-error methods, end-to-end methods directly yield an optimal structure that produces the desired EM responses. Many works utilize ML-based inverse models to realize end-to-end EM design [34]–[38]. The ML-based methods are often criticized for high computational cost in developing an accurate surrogate model, since the trial-and-error methods complete each design task at a lower computational cost. However, end-to-end methods can reuse the trained model for similar design tasks, gradually reducing the average computational cost for each design task as the number of design tasks increases. By contrast, the computational cost of the trial-and-error methods continuously grows.

Existing end-to-end design approaches suffer from impractical input requirements and non-uniqueness effects, as depicted in Fig. 1. Typical design specifications are broad, for example, $|S_{11}| < -10\,\mathrm{dB}$ from $1\,\mathrm{GHz}$ to $2\,\mathrm{GHz}$. They use single

This work was supported in part by the Research Matching Grant Scheme from the Research Grants Council of Hong Kong, in part by the National Natural Science Foundation of China under Grant 62073158, in part by the Basic Research Project of Education Department of Liaoning Province under Grant LJKZ0401, and in part by the "Xingliao Talent Program" of Liaoning Province under Grants XLYC2402025. *(Corresponding author: Tse-Tin Chan.)*

Zhao Zhou and Tse-Tin Chan are with the Department of Mathematics and Information Technology, The Education University of Hong Kong, Hong Kong SAR, China (e-mail: zzhou.xd@gmail.com; tsetinchan@eduhk.hk).

Zhaohui Wei, Jian Ren, and Yingzeng Yin are with the National Key Laboratory of Radar Detection and Sensing, Xidian University, Xi'an, 710071, China (e-mail: zhaohui_wei@163.com; renjianroy@gmail.com; yzyin@mail.xidian.edu.cn).

Jinna Li is with the School of Information and Control Engineering, Liaoning Petrochemical University, Fushun, 113001, China (e-mail: jnli@lnpu.edu.cn).

values to approximately summarize the characteristics over the whole band, such as the desired minimum operating frequency (1 GHz), maximum operating frequency (2 GHz), and maximum allowable value for reflection coefficients (-10 dB). Other information like the desired center frequency f_c and bandwidth BW may be included as needed. However, existing methods require a plot of the desired $|S_{11}|$ at a large number of frequency points sampled within the frequency band of interest, which is difficult to obtain from real-world design constraints. This limits the usefulness and application of existing approaches.

Non-uniqueness causes traditional inverse models to struggle when multiple structures produce similar responses. This is because the loss function, which measures the difference between predicted and actual structures, can become discontinuous and non-differentiable. Suppose there are two very different structures, A and B, with almost identical responses. A tiny change in the desired response might cause the model to suddenly jump from predicting something close to structure A to something close to structure B, making the loss function unstable. This instability severely harms the model's convergence and accuracy.

Several studies have attempted to address either the impractical input requirements or non-uniqueness effects, but not both simultaneously. Z. Gu *et al.* used a generative adversarial network (GAN) to convert practical frequency requirements into a $|S_{11}|$ curve [39]. Enriching the $|S_{11}|$ information helped weaken the effects of non-uniqueness but could not guarantee optimized designs. Even with enriched features, multiple structures might still generate similar $|S_{11}|$ curves, causing the model to struggle. C. C. Nadell *et al.* proposed a fast-forward dictionary search to mitigate the effects of non-uniqueness in an all-dielectric metasurface design application [40]. Using a forward model, they generated a dictionary of all possible structure-response pairs comprising 8.157×10^8 data points. For a given desired response, designers searched the dictionary to find a list of responses closely matching the desired response and selected an optimal design from the corresponding structures. While effective, this approach required significant computational resources, such as a powerful GPU, to generate and operate the dictionary.

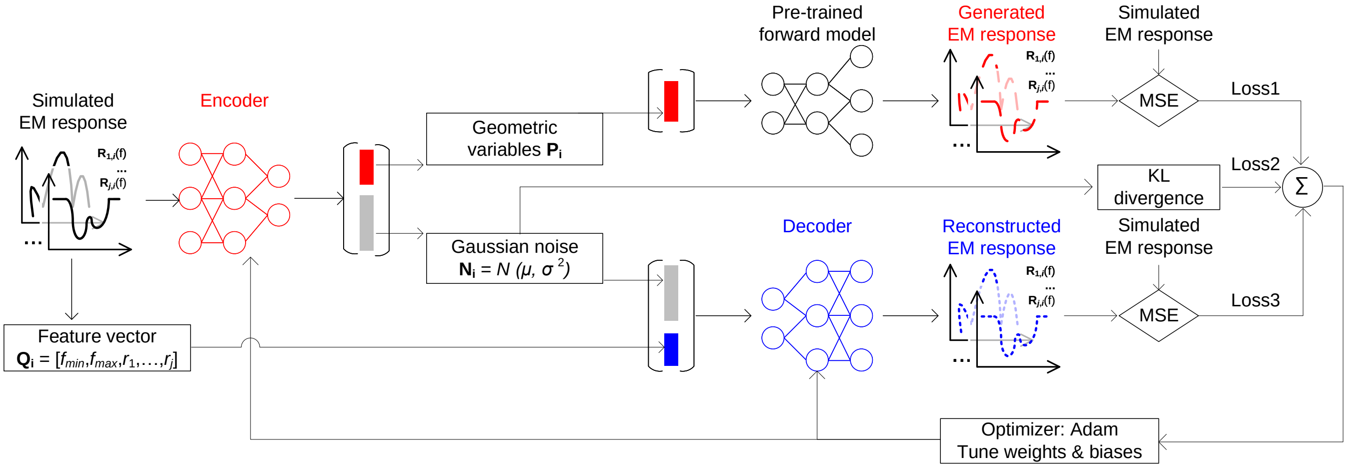
C. Zhang *et al.* proposed a multivalued neural network to associate each EM response input with N structures as output [41]. N should equal the maximum number of possible output structures to cover all the possibilities. Afterward, they used a pre-trained forward model to filter out M ($M \leq N$) valid ones from the N output structures. The multi-circuit process was then utilized to select a sub-optimal solution from the M valid structures. However, it is difficult to determine the best value for N. An unnecessarily large N value wastes computational resources, and a small N value could exclude the desired output structure from the N outputs. L. Yuan *et al.* mitigated the effects of non-uniqueness by establishing an improved transfer function-based artificial neural network model [42]. They utilized a pole-and-residue-based transfer function to convert the desired $|S_{11}|$ curves into poles and residues as input. The generated multiple structures were fed into pole-based and residue-based forward branches to filter out a sub-optimal output. However, it still requires detailed curves as input, which is impractical in real-world engineering scenarios. The above-mentioned reference works could not realize end-to-end design from practical constraints to optimized structures.

This paper proposes an end-to-end EM design framework that solves the impractical input issue and mitigates the effects of non-uniqueness simultaneously. A modified conditional variational autoencoder (MCVAE) is developed to map practical constraints to optimized structure designs directly. The objectives and operational principles of the proposed model distinctly differ from those of traditional VAEs and CVAEs. They are utilized for data augmentation by creating new data that resembles real data across various domains, including medical data [43], spectral information [44], industrial processes [45], spectroscopy measurements [46], remote sensing [47], and battery life [48], particularly where original data are scarce or difficult to obtain. Even though some research has claimed the application of CVAEs for the inverse design of airfoils [49] or molecules [50], their methodology closely resembles data augmentation. They accomplished inverse design by utilizing the decoder to produce new molecules or airfoils similar to existing ones. The conventional VAEs or CVAEs may fail to converge or produce less accurate outputs due to the non-uniqueness effects. Unlike these previous CVAEs-related studies, which focus on various forms of data augmentation, the proposed model seeks to connect practical constraints with optimized structures for end-to-end EM design. The constraints are enriched and converted into detailed responses over the desired band using a modified conditional decoder. The detailed responses are then fed into the modified encoder to determine the geometric variables of the structure. A pre-trained forward model evaluates the generated EM responses of the output structure and further mitigates the effects of non-uniqueness. Compared to existing end-to-end methods that suffer from impractical input requirements or non-uniqueness effects, the proposed end-to-end framework directly generates optimized structure designs based on practical constraints. The effectiveness of the proposed method is validated in multiple implementations: a linear-to-circular polarization converter design, a Fourier-phased metasurface design, a slotline filter design, a loop polarization converter design, and a polarization converter design with double and triple-expanded parameter ranges.

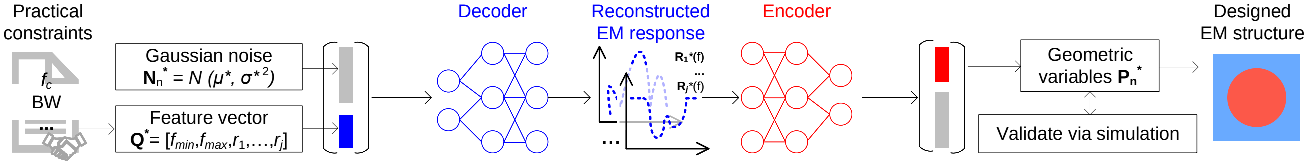
The remaining content is organized as follows. Section II introduces the working principle of the proposed design framework. Section III describes the first three implementations to validate the proposed method and compares it with existing end-to-end methods. Section IV discusses the capability of handling multiple design objectives and the effects of wide parameter ranges through another three implementations. Section V gives the conclusion.

II. PROPOSED DESIGN FRAMEWORK

Existing ML-based end-to-end design methods suffer from either impractical input requirements or non-uniqueness effects. We propose a modified conditional variational autoen-



(a) Training stage



(b) Design stage

Fig. 2. The proposed end-to-end design framework addresses the limitations of typical methods by accommodating practical input and generating optimized structures. (a) Training stage. (b) Design stage.

coder (MCVAE) to achieve the end-to-end design from practical EM response requirements to optimized structure designs, as shown in Fig. 2. It resolves the impractical input issue and mitigates the effects of non-uniqueness simultaneously.

A. Practical Constraints

Existing end-to-end methods require impractical input, such as detailed EM responses across the entire frequency range of interest. These are difficult or impractical to derive directly from real-world constraints. Practical constraints usually consist of only single values that approximately summarize the overall characteristics within the desired frequency band. These include the minimum frequency, maximum frequency, maximum allowable value for reflection coefficients, average transmission phase (for filters or transmissive metasurfaces), maximum or average Gain (for antennas), overall size, and other relevant information. It is impractical to obtain, for example, the entire reflection coefficient curve based on its maximum allowable value. These summary values from the practical constraints are the practical input for the end-to-end design.

B. Dataset Preparation

As with all ML-based design methods, sufficient data are required to train the model. The additional time and computational costs associated with dataset preparation are the main drawbacks often cited for ML-based methods. The trial-and-error methods (including the improved versions supported with optimization algorithms) involve fewer data computations during the iterative design process for a given design task. However, the trial-and-error methods need a similar amount of data computations for every design task, which means its time and computational costs continuously grow as the number of design tasks increases. Although the ML-based design methods spend extra time and computational costs in dataset preparation, they do not require repeated data generation in future design tasks. By comparison, the ML-based design methods significantly reduce the time and computational costs on average, especially when dealing with multiple similar design tasks.

The proposed model requires training, validation, and test datasets in a ratio of $7:2:1$. Each data entry contains a combination of geometric variables $\mathbf{P}_i$, corresponding EM responses $\mathbf{R}_{j,i}$ ($j \in [1, N_R]$), and associated feature vectors $\mathbf{Q}_i$. $\mathbf{P}_i$ is composed of all the tunable geometric variables. N_R denotes the number of EM response types of interest. Different EM responses can be in different sizes. $i \in [1, N_{data}]$, N_{data} equals the data amount. N_{data} depends on the dimensionality of geometric variables and the requirement for accuracy. A larger dataset generally improves the accuracy of an ML model up to a point. The rate of improvement typically follows a logarithmic curve, eventually plateauing when the data have sufficiently covered the search space. For simplicity and a fair comparison with existing methods, N_{data} pairs of data are predetermined independently instead of determined through sequential sampling. Latin Hypercube Sampling is employed to acquire the N_{data} data to capture the design space sufficiently. As the complexity and dimensionality increase, it may be difficult to acquire sufficient data. Advanced sampling strategies, data augmentation techniques, and small-sample machine learning methods can be applied to improve the model convergence and accuracy. $\mathbf{Q}_i = [f_{min}, f_{max}, r_{1,i}, ..., r_{j,i}]$ describes a set of practical constraints that best matches $\mathbf{R}_{j,i}$ ($j \in [1, N_R]$). f_{min} and f_{max} define the desired frequency range. $r_{j,i}$ ($j \in [1, N_R]$) denotes the practical requirement for $\mathbf{R}_{j,i}$. The collected data are converted or normalized into a range between 0 and 1. The conversion or normalization manner depends on the data type, which will be introduced in specific implementations.

C. Training Stage

The architecture and working principle of the proposed MCVAE in its training stage are shown in Fig. 2(a). The typical VAE or CVAE encoder is only responsible for converting the input information into a latent space. The latent space for CVAEs is modeled as a Gaussian distribution, which simplifies sampling, optimization, and analysis. Also, because of the Central Limit Theorem, Gaussian noise naturally approximates complex real-world distributions. Additionally, Gaussian noise is smooth and continuous, enabling controlled perturbations that assess the latent space's smoothness and the model's generalization ability. Its mathematical properties, such as well-defined mean and variance, further enhance its utility for efficient optimization and evaluation. Together, these characteristics make Gaussian noise an ideal choice for analyzing and refining the latent space in CVAEs.

Our modified encoder ($E(\cdot) = [E_1(\cdot),\ E_2](\cdot)$) not only converts ($E_2(\cdot)$) each input simulated responses $\mathbf{R}_{j,i}$ ($j \in [1, N_R]$) into a distinct Gaussian noise vector ($\mathbf{N}_i$) in a latent space, but also predicts ($E_1(\cdot)$) the geometric variables $\mathbf{P}'_i$ that generate the input responses,

$$\begin{aligned} [\mathbf{P}'_i,\ \mathbf{N}_i] &= E(\mathbf{R}_{j,i}), \\ &= [E_1(\mathbf{R}_{j,i}),\ E_2(\mathbf{R}_{j,i})] \end{aligned} \tag{1}$$

The Gaussian noise is applied independently to each sample, ensuring that the latent space is stochastic and that each sample has a distinct latent representation. During training, the encoder outputs μ and $\log(\sigma^2)$ for each input sample. μ and σ^2 define the approximate distribution N(μ, σ^2). The approximate distribution N(μ, σ^2) is regularized to be close to a standard Gaussian distribution N(0,1) by minimizing the Kullback-Leibler (KL) divergence (referred to as $Loss2$ included in the loss function). By doing so, each input sample is represented into a distinct latent variable z computed using the reparameterization trick: $z = \mu + \sigma \times \epsilon$, where ϵ is a random noise vector drawn from a standard normal distribution. A distinct Gaussian noise vector z is combined with each feature vector $\mathbf{Q}_i$ and fed into the modified decoder ($D(\cdot)$) to reconstruct the responses $\mathbf{R}''_{j,i}$ ($j \in [1, N_R]$),

$$\begin{aligned} \mathbf{R}''_{j,i} &= D(\mathbf{N}_i, \mathbf{Q}_i), \\ &= D(E_2(\mathbf{R}_{j,i}), \mathbf{Q}_i). \end{aligned} \tag{2}$$

The predicted geometric variables $\mathbf{P}'_i$ is fed into the pre-trained forward model (F) to evaluate the EM responses $\mathbf{R}'_{j,i}$ ($j \in [1, N_R]$) generated by the predicted geometric variables $\mathbf{P}'_i$. The forward model is pre-trained using the same datasets. A more accurate forward model improves the end-to-end model's performance, enhancing design precision.

The loss function ($Loss$) is defined as a weighted sum of three sub-losses: the generation loss ($Loss1$), the KL-divergence of the latent space ($Loss2$), and the reconstruction loss ($Loss3$),

$$Loss = \frac{1-\beta}{2} \times (Loss1 + Loss3) + \beta \times Loss2, \tag{3}$$

where β is a weight that gradually increases from 0.1 to 0.8 over epoch,

$$\beta = 0.1 + \frac{0.7}{(Epoch_{max} - 1)} \times (Epoch_{current} - 1). \tag{4}$$

Here, $Epoch_{max}$ equals the maximum number of epochs, and $Epoch_{current}$ denotes the index of the current epoch. β forces the model to focus on the generation ability ($Loss1$) and reconstruction ability ($Loss3$) at the beginning, and then gradually enforce the latent space structure ($Loss2$).

$Loss1$ equals the mean squared error (MSE) between the generated responses $\mathbf{R}'_{j,i}$ ($j \in [1, N_R]$) and simulated responses $\mathbf{R}_{j,i}$ ($j \in [1, N_R]$) to evaluate the predicted geometric variables,

$$\begin{aligned} Loss1 &= \frac{1}{N_{train}} \sum_{i \in N_{train}} \sum_{j}^{N_R} |\mathbf{R}'_{j,i} - \mathbf{R}_{j,i}|^2, \\ &= \frac{1}{N_{train}} \sum_{i \in N_{train}} \sum_{j}^{N_R} |F(\mathbf{P}'_i) - \mathbf{R}_{j,i}|^2, \\ &= \frac{1}{N_{train}} \sum_{i \in N_{train}} \sum_{j}^{N_R} |F(E_1(\mathbf{R}_{j,i})) - \mathbf{R}_{j,i}|^2. \end{aligned} \tag{5}$$

$Loss2$ is the Kullback-Leibler (KL) divergence of the normal distribution to assess the conversion into the latent space,

$$\begin{aligned} Loss2 &= \frac{1}{N_{train}} \sum_{i \in N_{train}} KL\ (N_i(\mu,\ \sigma^2)\ ||\ N(0,\ 1)), \\ &= \frac{1}{N_{train}} \sum_{i \in N_{train}} KL\ (E_2(\mathbf{R}_{j,i})\ ||\ N(0,\ 1)). \end{aligned} \tag{6}$$

$Loss3$ equals the MSE between the reconstructed responses $\mathbf{R}''_{j,i}$ ($j \in [1, N_R]$) and simulated responses $\mathbf{R}_{j,i}$ ($j \in [1, N_R]$) to evaluate the reconstruction ability of the decoder (D),

$$\begin{aligned} Loss3 &= \frac{1}{N_{train}} \sum_{i \in N_{train}} \sum_{j}^{N_R} |\mathbf{R}''_{j,i} - \mathbf{R}_{j,i}|^2, \\ &= \frac{1}{N_{train}} \sum_{i \in N_{train}} \sum_{j}^{N_R} |D(\mathbf{N}_i, \mathbf{Q}_i) - \mathbf{R}_{j,i}|^2, \\ &= \frac{1}{N_{train}} \sum_{i \in N_{train}} \sum_{j}^{N_R} |D(E_2(\mathbf{R}_{j,i}), \mathbf{Q}_i) - \mathbf{R}_{j,i}|^2. \end{aligned} \tag{7}$$

The sum of $Loss1$, $Loss2$, and $Loss3$ is differentiated and back-propagated to optimize the weights and biases of the modified encoder and decoder.

In specific implementations, the architecture and hyperparameters of the MCVAE (including the encoder, decoder, and forward model) are optimized using Bayesian optimization. The MCVAE is trained by minimizing a defined loss function to achieve optimal accuracy, as its accuracy directly influences the quality of the resulting designs. Tunable hyperparameters include the number of hidden layers, number of neurons, number of epochs, dimensionality of the latent space (dimensionality of Gaussian noise vector), learning rate, and batch size.

Here is the step-by-step guideline for the training stage of the proposed method:

1) For the EM structure under design, select the tunable geometric parameters and determine their sampling ranges based on the desired design freedom and acceptable solving complexity (a wide sampling range provides high design freedom but increases the solving complexity, while a narrow sampling range is easy to solve but limits the design freedom);
2) Define N_{data} combinations of geometric parameters $\mathbf{P}_i$ ($i \in [1, N_{data}]$) through Latin Hypercube Sampling, corresponding to N_{data} varying structures, where N_{data} is decided based on design experience and available computational power (an appropriate amount N_{data} of data sufficiently interprets the solving space with the minimum computation cost);
3) Simulate the N_{data} structures to obtain the EM responses of interest $\mathbf{R}_{j,i}$ ($i \in [1, N_{data}]$, $j \in [1, N_R]$), where $N_R > 1$ indicate that multiple types of EM responses are considered;
4) Define a feature vector $\mathbf{Q}_i$ for each set of $\mathbf{R}_{j,i}$ (definition of $\mathbf{Q}_i$ is presented for each implementation in Section III);
5) Normalize geometric parameters, EM responses, and feature vectors between 0 and 1 (normalization procedure is explained for each implementation in Section III);
6) Divide the N_{data} sets of geometric parameters, simulated responses, and feature vectors into a training, validation, and test dataset in a ratio of $7:2:1$;
7) Train a forward model that can take the geometric parameters as input and predict the EM responses;
8) Integrating the pre-trained forward model, train a proposed model that can take practical design specifications as input and generate optimized EM structures.

D. Design Stage

Fig. 2(b) demonstrates how the proposed MCVAE works during the design process:

1) The given practical constraints are described as a feature vector $\mathbf{Q}^* = [f_{min}, f_{max}, r_1, ..., r_j]$. The size of the feature vector N_R is minimized to contain only the necessary information in the constraints. The default elements include f_{min}, f_{max}, and r_1 that define the desired frequency range and maximum allowable value for reflection coefficients.
2) A new Gaussian noise vector $\mathbf{N}_n^*$ is initialized arbitrarily. Its size equals the dimensionality of the latent space determined in the training stage.
3) Each feature vector $\mathbf{Q}^*$ is concatenated with an initialized Gaussian noise vector $\mathbf{N}_n^*$. The concatenated vector is fed into the decoder to reconstruct the EM responses over frequency $\mathbf{R}_{j,n}^*$ ($j \in [1, N_R]$) that fit the predefined constraints $\mathbf{Q}^*$.
4) The reconstructed responses $\mathbf{R}_{j,n}^*$ ($j \in [1, N_R]$) are fed into the encoder to predict the geometric variables $\mathbf{P}_n^*$ of an optimized EM structure design.
5) Repeat step 2-4 to obtain multiple structures $\mathbf{P}_n^*$, where n can be set between 1 and N_n. N_n is set as 3 in the implementations in this paper. N_n can be increased if severe non-uniqueness occurs or better structures are needed.
6) The designed structures are simulated via CST to evaluate the responses. The simulated responses are evaluated to select the optimal EM structure design that best aligns with the specified practical constraints.

There might be concerns that reliance on random noise as input could lead to inconsistent and suboptimal designs. It is true that the output design is not fixed with a random noise as input. The proposed model is specifically developed and trained to generate output designs that are consistent with the input practical constraints provided. Given the same input practical constraints, the proposed model should produce output designs that align with these constraints. The random noise introduces variability into the output, allowing the model to generate diverse output designs even when the same practical constraints are provided. The noise is managed by sampling from a standard Gaussian distribution to ensure consistency. This variability allows the model to explore different plausible output designs. It actually increases the possibility of finding an optimal design by exploring a wider design space. While the output designs are diverse, they are still consistent with the input practical constraints.

It is important to differentiate between consistency and diversity. Consistency refers to its ability to generate output designs that are true to the practical constraints, and diversity refers to the range of output designs that the model can produce under the same input practical constraints. The proposed model aims to generate consistent output designs that fit the practical constraints while allowing diversity to explore more optimal designs.

Although human intervention may still be required to refine or validate the final design, especially for complex and novel designs, the proposed method significantly accelerates the whole optimization and design process, greatly reducing reliance on human experience and intervention. With more data involved to improve its accuracy and more implementations to validate its effectiveness, it may further reduce the need for human intervention.

E. Mitigating the Effects of Non-Uniqueness

We mitigate the effects of non-uniqueness through two aspects: incorporating a Gaussian noise vector and implementing a forward-model-based loss term ($Loss1$). While Gaussian noise allows exploration of the latent space, the effects of non-uniqueness are primarily mitigated through the forward-model-based loss function, which ensures that only structures yielding the correct EM response are retained. By doing so, the proposed framework generates valid designs meeting practical constraints, even if multiple solutions exist.

Let us consider a non-uniqueness condition where two different combinations of geometric variables generate similar EM responses, and $j = 1$, $\mathbf{R}_{j,i}$ is expressed as $\mathbf{R}_i$ for the sake of simplicity,

$$|\mathbf{R}_x - \mathbf{R}_y| < \epsilon, \tag{8}$$

$$|\mathbf{P}_x - \mathbf{P}_y| \gg \epsilon. \tag{9}$$

Here, ϵ is a minimum threshold. Note that the mechanism is the same when $j > 1$ and multiple combinations of geometric variables generate similar EM responses.

The existing ML inverse model-based methods define the loss function as the difference between predicted and real geometric variables,

$$\begin{aligned} Loss_i &= |\mathbf{P}'_i - \mathbf{P}_i|^2, \\ &= |I(\mathbf{R}_i) - \mathbf{P}_i|^2, \end{aligned} \tag{10}$$

$$\begin{aligned} \lim_{\mathbf{R}_i \to \mathbf{R}_x} Loss_x &= \begin{cases} |I(\mathbf{R}_x) - \mathbf{P}_x|^2, \mathbf{R}_i = \mathbf{R}_x, \\ |I(\mathbf{R}_y) - \mathbf{P}_x|^2, \mathbf{R}_i = \mathbf{R}_x \pm \epsilon = \mathbf{R}_y, \end{cases} \\ &= \begin{cases} |\mathbf{P}'_x - \mathbf{P}_x|^2, \mathbf{R}_i = \mathbf{R}_x, \\ |\mathbf{P}'_y - \mathbf{P}_x|^2, \mathbf{R}_i = \mathbf{R}_x \pm \epsilon = \mathbf{R}_y, \end{cases} \end{aligned} \tag{11}$$

$$\begin{aligned} \lim_{\mathbf{R}_i \to \mathbf{R}_y} Loss_y &= \begin{cases} |I(\mathbf{R}_y) - \mathbf{P}_y|^2, \mathbf{R}_i = \mathbf{R}_y, \\ |I(\mathbf{R}_x) - \mathbf{P}_y|^2, \mathbf{R}_i = \mathbf{R}_y \pm \epsilon = \mathbf{R}_x, \end{cases} \\ &= \begin{cases} |\mathbf{P}'_y - \mathbf{P}_y|^2, \mathbf{R}_i = \mathbf{R}_y, \\ |\mathbf{P}'_x - \mathbf{P}_y|^2, \mathbf{R}_i = \mathbf{R}_y \pm \epsilon = \mathbf{R}_x. \end{cases} \end{aligned} \tag{12}$$

It can be observed that when $\mathbf{R}_i$ gets close to $\mathbf{R}_x$ and $\mathbf{R}_y$, the similarity between $\mathbf{R}_x$ and $\mathbf{R}_y$ could cause discontinuity of $Loss_x$ and $Loss_y$,

$$\Delta \lim_{\mathbf{R}_i \to \mathbf{R}_x} Loss_x = \left| \left|\mathbf{P}'_x - \mathbf{P}_x\right|^2 - \left|\mathbf{P}'_y - \mathbf{P}_x\right|^2 \right| \gg \epsilon, \tag{13}$$

$$\Delta \lim_{\mathbf{R}_i \to \mathbf{R}_y} Loss_y = \left| \left|\mathbf{P}'_y - \mathbf{P}_y\right|^2 - \left|\mathbf{P}'_x - \mathbf{P}_y\right|^2 \right| \gg \epsilon. \tag{14}$$

This discontinuity would lead to non-differentiability of the loss function at $\mathbf{R}_x$ and $\mathbf{R}_y$, deteriorating the model convergence and accuracy.

The involvement of Gaussian noise $\mathbf{N}_i$ helps mitigate the ambiguity caused by the non-uniqueness effects. The added noise $\mathbf{N}_i$ introduces stochasticity into the decoding process. It prevents the decoder from always producing the same reconstructed response for similar inputs. Even if two different structures ($\mathbf{P}_x$ and $\mathbf{P}_y$) have very similar responses ($\mathbf{R}_x$ and $\mathbf{R}_y$), the decoder will likely produce two different reconstructed responses ($\mathbf{R}''_x$ and $\mathbf{R}''_y$) because of the random noise $\mathbf{N}_i$. The ambiguity between $\mathbf{R}_x$ and $\mathbf{R}_y$ is alleviated, reducing the discontinuity and non-differentiability. This forces the encoder to learn a more robust mapping, as it must account for the variations introduced by the noise. In case of non-uniqueness, the encoder is forced to distinguish between similar responses based on $\mathbf{N}_i$. Essentially, $\mathbf{N}_i$ prevents the encoder from collapsing different structures onto the same output, thereby enhancing the model convergence and accuracy.

Integrating the forward-model-based loss term ($Loss1$) further mitigates the non-uniqueness effects and encourages the generation of optimized designs that satisfy the practical constraints, even when multiple valid solutions exist. In the assumed non-uniqueness condition, $Loss1$ can be described as,

$$\begin{aligned} Loss1_i &= |\mathbf{R}'_{j,i} - \mathbf{R}_i|^2, \\ &= |F(\mathbf{P}_i{}') - \mathbf{R}_i|^2, \\ &= |F(E_1(\mathbf{R}_i)) - \mathbf{R}_i|^2, \end{aligned} \tag{15}$$

$$\lim_{\mathbf{R}_i \to \mathbf{R}_x} Loss1_x = \begin{cases} |F(E_1(\mathbf{R}_x)) - \mathbf{R}_x|^2, \mathbf{R}_i = \mathbf{R}_x, \\ |F(E_1(\mathbf{R}_y)) - \mathbf{R}_x|^2, \mathbf{R}_i = \mathbf{R}_x \pm \epsilon = \mathbf{R}_y, \end{cases} \tag{16}$$

$$\lim_{\mathbf{R}_i \to \mathbf{R}_y} Loss1_y = \begin{cases} |F(E_1(\mathbf{R}_y)) - \mathbf{R}_y|^2, \mathbf{R}_i = \mathbf{R}_y, \\ |F(E_1(\mathbf{R}_x)) - \mathbf{R}_y|^2, \mathbf{R}_i = \mathbf{R}_y \pm \epsilon = \mathbf{R}_x. \end{cases} \tag{17}$$

$Loss1$ remains continuous,

$$\begin{aligned} &\Delta \lim_{\mathbf{R}_i \to \mathbf{R}_x} Loss1_x \\ &= \left| |F(E_1(\mathbf{R}_x)) - \mathbf{R}_x|^2 - |F(E_1(\mathbf{R}_y)) - \mathbf{R}_x|^2 \right| < \epsilon, \end{aligned} \tag{18}$$

$$\begin{aligned} &\Delta \lim_{\mathbf{R}_i \to \mathbf{R}_y} Loss1_y \\ &= \left| |F(E_1(\mathbf{R}_y)) - \mathbf{R}_y|^2 - |F(E_1(\mathbf{R}_x)) - \mathbf{R}_y|^2 \right| < \epsilon. \end{aligned} \tag{19}$$

Therefore, $Loss1$ is always differentiable, ensuring a good model convergence level and high accuracy. The consequences of the ambiguity between $\mathbf{R}_x$ and $\mathbf{R}_y$ are canceled out by the pre-trained forward model (F). The modified encoder may output various geometric variables that generate similar EM responses. $Loss1$ does not directly compare predicted ($\mathbf{P}'_{x/y}$) and actual ($\mathbf{P}_{x/y}$) structures. Instead, it compares the responses ($\mathbf{R}'_{x/y}$) generated by the predicted structure (using the forward model) with the target responses ($\mathbf{R}_{x/y}$). The forward model (F) can tolerate these variances of the predicted geometric variables and solely evaluate the generated EM responses. This is key because even if the encoder predicts different structures, as long as those structures produce the correct responses, $Loss1$ will be low. It forces the encoder to learn to produce structures that generate the desired responses, aligning well with the goal of end-to-end design. Given a set of practical constraints, the well-trained modified encoder will produce an optimized output structure, that generates the desired responses, during each design process. It outputs a new output structure for the fixed requirements if a new Gaussian noise vector is initialized in a new design process. Designers can obtain multiple valid structures by running the design process multiple times N_n. The designed structures can be evaluated using full-wave simulation, from which designers can obtain an optimal EM structure to match the most closely with the specified practical constraints.

III. IMPLEMENTATION

The effectiveness of the proposed design framework is validated through the first three implementations: a linear-to-circular polarization converter design, a Fourier-phased metasurface design, and a slotline filter design. Each implementation includes four arbitrarily chosen design scenarios. Genetic Algorithm (GA)- and Particle Swarm Optimization (PSO)-based design methods are employed for comparison. The implementation results indicate that the proposed framework achieves an end-to-end design from practical constraints to optimized structure designs. Various constraints can be integrated into $\mathbf{Q}$ as input, and different types of EM structures can be represented as output. For example, designers can define additional parameters, such as r_{j+1} for average Gain and $[r_{j+2},\ r_{j+3},\ r_{j+4}]$ for maximum $3D$ size, to integrate antenna and size requirements.

A. Implementation I: Linear-to-Circular Polarization Converter Design

1) Structure of the Linear-to-Circular Polarization Converter: The polarization converter presented in [52] transforms a linearly polarized wave into a circularly polarized wave. It operates from 5 GHz to 15 GHz and consists of a 2-D arrangement of sub-wavelength unit cells. The design requirements are represented as $\mathbf{Q} = [f_{min},\ f_{max},\ r_1]$. f_{min} and f_{max} define the desired operating frequency range. r_1 indicates the maximum reflection coefficient value within the desired frequency range, and the default value of r_1 is $-10\,\mathrm{dB}$. Fig. 3 shows the unit cell structure. It comprises four capacitive patch layers, two wire grid layers, four substrate layers, and three bonding layers. The patch and wire grid layers are metal layers etched on the surfaces of the substrate layers. The top and bottom substrate layers have two patch layers etched on their outer sides, respectively. The remaining two patch layers are etched onto the inner sides of the two middle substrate layers, while their outer sides are etched with the two wire grid layers, respectively. Each pair of adjacent substrate layers is separated and bonded with a bonding layer. The substrate layer uses *Rogers RT/duroid 6010* with a dielectric constant of $\varepsilon_r = 10.2$ and a thickness of $h_1 = 1.27\,\mathrm{mm}$. The bonding layer is made of *Rogers 4450F* with a dielectric constant of $\varepsilon_r = 3.52$ and a thickness of $h_2 = 0.1\,\mathrm{mm}$. The remaining geometric variables ($\mathbf{P} = [p_{1x},\ p_{1y},\ w_{2x},\ w_{2y},\ p_{3x},\ p_{3y}]$) are designated as tuning parameters, as listed in Table I.

2) Design Using Proposed Method: Training, validation, and test datasets are collected to train the proposed model. We arbitrarily define 300 sets of geometric variables ($\mathbf{P}_i$, $i \in [1,\ 300]$) within their tuning ranges, corresponding to 300 different structures. They are simulated via CST to collect the 300 reflection coefficient curves ($|S_{11}|$s). The values of geometric variables are normalized and converted into $\mathbf{P}_i = [\bar{p_{1x}},\ \bar{p_{1y}},\ \bar{w_{2x}},\ \bar{w_{2y}},\ \bar{p_{3x}},\ \bar{p_{3y}}]$, $i \in [1, 300]$. For example, $\bar{p_{1x}} = \frac{p_{1x}-\min(p_{1x})}{\max(p_{1x})-\min(p_{1x})}$.

Each $|S_{11}|$ curve is discretized into a vector of size 51 ($\mathbf{R}_{1,i}$), containing $|S_{11}|$ values evaluated at 51 equally spaced frequency points from 5 GHz to 15 GHz. The $|S_{11}|$ values, expressed in decibels (dB), are converted to a linear magnitude scale between 0 and 1. For each $\mathbf{R}_{1,i}$, a $\mathbf{Q}_i$ is assigned, which represents the practical design requirements that align well with $\mathbf{R}_{1,i}$. The definition of $\mathbf{Q}_i$ for $\mathbf{R}_{1,i}$ follows six steps (a)-(f):

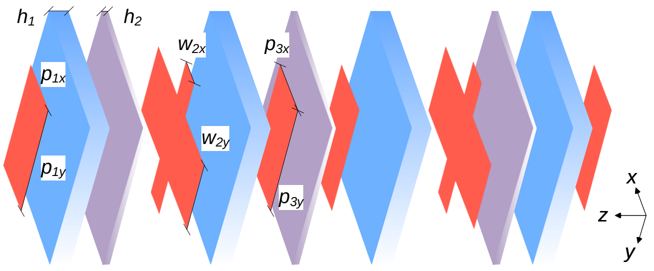


Fig. 3. Unit cell structure of the linear-to-circular polarization converter.

TABLE I

IMPLEMENTATION I: TUNABLE AND FIXED GEOMETRIC PARAMETERS FOR LINEAR-TO-CIRCULAR POLARIZATION CONVERTER DESIGN

Parameter	Value (mm)	Parameter	Value (mm)
p_{1x}	[2.50, 3.30]	p_{1y}	[0.60, 1.40]
w_{2x}	[0.40, 1.20]	w_{2y}	[0.10, 0.30]
p_{3x}	[2.10, 2.90]	p_{3y}	[1.20, 2.00]
h_1	1.27	h_2	0.10

(a) Set $r_1 = -10\,\mathrm{dB}$;
(b) Find the minimum frequency f_{min} where $\mathbf{R}_{1,i} \leq r_1$ by $\min_{f_{min}} \mathbf{R}_{1,i(f_{min})} \leq r_1$;
(c) Find the maximum frequency f_{max} where $\mathbf{R}_{1,i} \leq r_1$ by $\max_{f_{max}} \mathbf{R}_{1,i(f_{max})} \leq r_1$;
(d) Set r_{new} as the maximum $\mathbf{R}_{1,i}$ between f_{min} and f_{max};
(e) If $r_{new} > r_1$, update r_1 by $r_1 = r_{new}$, and repeat steps (b)-(e);
(f) Set $\mathbf{Q}_i = [f_{min}, f_{max}, r_1]$.

Assisted by programming in Python, the definition of $\mathbf{Q}_i$ for all the $\mathbf{R}_{1,i}$ ($i \in [1,\ 300]$) can be completed within several seconds. We normalize $\mathbf{Q}_i = [f_{min}, f_{max}, r_1]$ to $\mathbf{Q}_i = [\bar{f_{min}}, \bar{f_{max}}, \bar{r_1}]$ by

$$\bar{f_{min}} = \frac{f_{min} - 5\,\mathrm{GHz}}{15\,\mathrm{GHz} - 5\,\mathrm{GHz}}, \tag{20}$$

$$\bar{f_{max}} = \frac{f_{max} - 5\,\mathrm{GHz}}{15\,\mathrm{GHz} - 5\,\mathrm{GHz}}, \tag{21}$$

$$\bar{r_1} = \frac{\max\{-10\,\mathrm{dB},\ \max|S_{11}|_{(f_{min},\ f_{max})}\} + 10\,\mathrm{dB}}{10\,\mathrm{dB}}. \tag{22}$$

The 300 combinations of $\mathbf{P}_i$, $\mathbf{R}_{1,i}$, and $\mathbf{Q}_i$ are arbitrarily divided into a training, a validation, and a test dataset at the ratio of $7:2:1$.

The end-to-end model is developed and optimized through Bayesian optimization. Multiple hyperparameters are explored, including the number of training epochs, the learning rate, the batch size, the number of hidden layers, the number of neurons, and the dimensionality of Gaussian noise. Specifically, the number of training epochs is sampled from a quantized uniform distribution ranging from 100 to 1000, with a step size of 100. The learning rate is optimized using a discrete set of values, $[0.0001,\ 0.001,\ 0.01]$. The batch size is sampled from a quantized uniform distribution between 5 and 30, with increments of 5. The number of neurons in each layer for the encoder, decoder, and forward model is chosen from the set $[16,\ 32,\ 64,\ 128,\ 256]$. The number of hidden layers

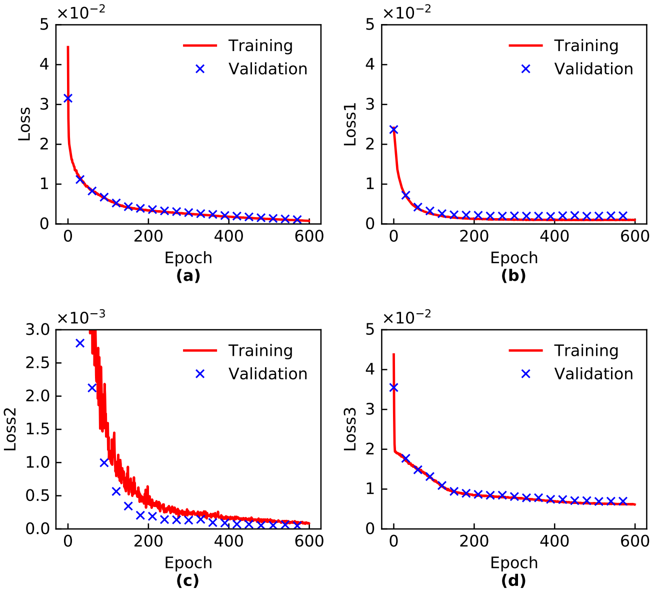


Fig. 4. Training and validation losses of the proposed model over epochs during the training process in Implementation I: Linear-to-Circular Polarization Converter Design. (a) $Loss$. (b) $Loss1$. (c) $Loss2$. (d) $Loss3$.

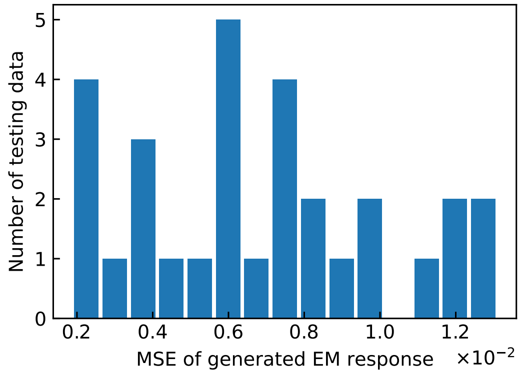


Fig. 5. Test loss of the proposed model in Implementation I: Linear-to-Circular Polarization Converter Design.

TABLE II
GEOMETRIC VARIABLES OF DESIGNED UNIT CELLS IN IMPLEMENTATION I: LINEAR-TO-CIRCULAR POLARIZATION CONVERTER DESIGN

Parameter (mm)	p_{1x}	p_{1y}	w_{2x}	w_{2y}	p_{3x}	p_{3y}
I-1	2.77	0.93	0.81	0.17	2.67	1.59
I-2	2.87	0.97	0.73	0.16	2.39	1.63
I-3	2.93	1.06	0.52	0.20	2.65	1.56
I-4	2.55	0.69	1.15	0.20	2.42	1.51

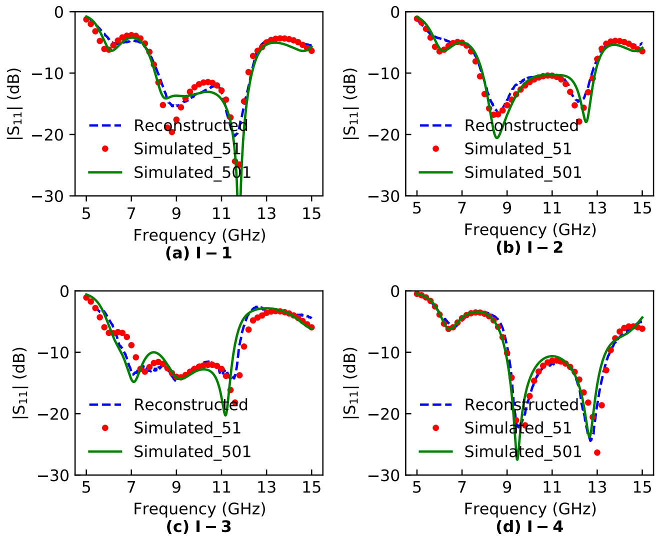


Fig. 6. Reconstructed and Simulated EM responses in Implementation I: Linear-to-Circular Polarization Converter Design. (a) I-1. (b) I-2. (c) I-3. (d) I-4. ("Simulated_51" and "Simulated_501" represent results using 51 and 501 frequency points, respectively. The reconstructed results are the same for both 51 and 501 frequency points and are denoted as "Reconstructed".)

for the encoder and decoder is sampled from a quantized uniform distribution between 1 and 5. The dimensionality of the Gaussian noise vector is sampled from a quantized uniform distribution between 3 and 10. The optimization is terminated after 50 iterations without improvement or upon reaching the maximum number of iterations, 1000. The validation loss on the validation dataset serves as the assessment metric.

After 1000 iterations, the best combinations of hyperparameters for the end-to-end model with the lowest validation losses are determined. After optimization, the encoder has 1 hidden layer of 256 neurons, and the decoder has 2 hidden layers of 16 and 128 neurons. The activation function is ReLU, except for the Linear function for the output layer. The dimensionality of latent space is 6. The batch size, number of epochs, and learning rate are 15, 600, and 0.001, respectively. Adam is utilized as the optimizer. The optimized forward model has 3 hidden layers of 256, 128, and 256 neurons. The batch size, number of epochs, and learning rate are 30, 900, and 0.0001, respectively. The remaining hyperparameters are the same as the encoder and decoder.

The forward model is pre-trained using the same datasets. It is optimized for the highest accuracy and is considered sufficiently accurate after Bayesian optimization, as its accuracy impacts the performance of the proposed end-to-end model. The pre-trained forward model is fixed in the following training stage of the proposed end-to-end model. Fig. 4 shows the trends of training and validation losses of the proposed end-to-end model over epochs during the training process, which converge to 7.0×10^{-3} and 9.1×10^{-3}, respectively. After training, the proposed model is evaluated using the test dataset. Fig. 5 exhibits the MSE distributions of the generated EM responses on the test dataset. The average MSE is 6.8×10^{-3}, and the maximum MSE is 1.3×10^{-2}. Higher model accuracy can be achieved using a relatively larger dataset ($N_{data} > 300$). Consequently, the time and computational cost required for data collection and model training also increase. However, the cost of model training remains negligible compared with data collection.

Four design cases are carried out to evaluate the proposed model performance. We predefine four settings of design

TABLE III
COMPARISON BETWEEN THE PROPOSED, GA-BASED, AND FORWARD-MODEL-BASED METHODS IN IMPLEMENTATION I: LINEAR-TO-CIRCULAR POLARIZATION CONVERTER DESIGN (UNIT: HOURS)

	Proposed	**GA-Based**	**Forward-model-based**
End-to-End	Yes	No	No
Design Performance	Offset $\leq \pm 0.1\,\mathrm{GHz}$	Offset $\leq \pm 0.1\,\mathrm{GHz}$	Offset $\leq \pm 0.1\,\mathrm{GHz}$
Data collection	4.32	-	4.32
Training	0.009	-	0.003
I-1	0.04	1.73	0.043
I-2	0.04	0.73	0.043
I-3	0.04	0.94	0.043
I-4	0.04	1.07	0.043
...	...	...	...
I-n	0.04	...	0.043
Total	$0.04n + 4.329$	$0.91n + 0.82$	$0.043n + 4.323$

requirements as $\mathbf{Q}^*_{1,2,3,4}$,

$$\mathbf{Q}^*_1 = [8.3\,\mathrm{GHz},\ 12.5\,\mathrm{GHz},\ -10\,\mathrm{dB}], \tag{23}$$

$$\mathbf{Q}^*_2 = [7.9\,\mathrm{GHz},\ 13.0\,\mathrm{GHz},\ -10\,\mathrm{dB}], \tag{24}$$

$$\mathbf{Q}^*_3 = [7.0\,\mathrm{GHz},\ 11.9\,\mathrm{GHz},\ -10\,\mathrm{dB}], \tag{25}$$

$$\mathbf{Q}^*_4 = [9.3\,\mathrm{GHz},\ 13.8\,\mathrm{GHz},\ -10\,\mathrm{dB}]. \tag{26}$$

They are normalized, combined with arbitrarily initialized Gaussian noise vectors $\mathbf{N}^*_n$, and fed into the proposed model to generate four linear-to-circular polarization converter unit cell designs. N_n is set as 3, and 3 optimized structures are generated for each $\mathbf{Q}^*_{1/2/3/4}$. They are simulated using CST to evaluate the generated $|S_{11}|_n$s, and an optimal structure is determined for each $\mathbf{Q}^*_{1/2/3/4}$. The geometric variables of the optimal structures are listed in Table II. Fig. 6 compares the generated $|S_{11}|$ curves (red dots) and reconstructed ones (blue dash lines) that the decoder outputs. Reconstructed curves represent the responses output by the decoder, while simulated curves represent CST full-wave simulated responses of the generated structures. They match well with the predefined requirements $\mathbf{Q}^*_{1,2,3,4}$, with a maximum frequency offset of $0.3\,\mathrm{GHz}$, $3\,\%$ with reference to the center frequency $10\,\mathrm{GHz}$.

We also validate Implementation I when increasing the frequency points from 51 to 501. A new model is optimized and trained, and the generated $|S_{11}|$ curves are represented as green solid lines. As the number of sampled frequency points increases from 51 to 501, the training data contains richer information revealing the interpolation characteristics, enabling the model to understand frequency-wise interpolation better. Consequently, the model trained with 501 frequency points generates more accurate $|S_{11}|$. The alignment between the generated responses and predefined requirements improves, reducing the maximum frequency offset from $0.3\,\mathrm{GHz}$ to $0.1\,\mathrm{GHz}$. A better design quality may be achieved if the number of training data can be increased from 300. This validates the effectiveness of the proposed end-to-end framework for the linear-to-circular polarization converter.

3) Comparison with Existing Methods: Existing ML-based methods require detailed $|S_{11}|$ curves as input and thus need significant modifications to enable end-to-end design directly from practical constraints $\mathbf{Q}^*_{1,2,3,4}$ to optimized converter designs in Implementation I: Linear-to-Circular Polarization Converter Design. A possible modification is integrating transfer functions into the existing ML-based methods to transform the constraints into detailed responses over frequency. However, transfer functions are typically established exclusively for specific EM responses, such as pole-based transfer functions for $|S_{11}|$. It is difficult to generalize to other types of EM structures.

For comparison, GA, as a representative of optimization algorithm-based design methods, is employed in Implementation I: Linear-to-Circular Polarization Converter Design. We use the built-in GA supported by CST to carry out the four design tasks $\mathbf{Q}^*_{1,2,3,4}$ individually. For a fair comparison, the optimized design for the first task $\mathbf{Q}^*_1$ is taken as the starting point of the following design tasks $\mathbf{Q}^*_{2,3,4}$. The initial population size and mutation rate are 40 and 0.6, respectively. The optimization stops when its best structure meets the predefined constraints $\mathbf{Q}^*_{1/2/3/4}$. Note that for $\mathbf{Q}^*_1$, the generated frequency band by the proposed method has a slight offset of $0.3\,\mathrm{GHz}$. Thus, the optimization process for $\mathbf{Q}^*_1$ stops when the offset of its generated frequency band does not exceed $0.1\,\mathrm{GHz}$.

The time needed for the optimization process in the four design scenarios is compared with the proposed method in Table III. Although the proposed method takes $4.33\,\mathrm{hours}$ for data collection and model training, it can determine the converter designs within $0.04\,\mathrm{hours}$ for design and $0.04\,\mathrm{hours}$ for validation. Despite that using the first optimized design as the starting point for the following design tasks $\mathbf{Q}^*_{2,3,4}$ reduces the required time for $\mathbf{Q}^*_{2,3,4}$, GA-based methods still require a much longer time for each design task. Compared to GA, the proposed method significantly reduces the required time and computational costs as the number of design tasks increases. As can be observed in Table III, the required time and computational costs are mainly determined by the simulation time. The simulation time is primarily determined by the complexity of the geometry, the mesh density, and the frequency range being simulated, rather than the number of frequency points. As the design complexity increases, the proposed method requires more time for data collection, while the GA-based methods require more time for each design task. As the number of design tasks grows, the total time consumption of the GA-based methods increases faster than the proposed method. In the long run, the proposed method still significantly reduces the computation cost and improves the design efficiency, even if the complexity of the EM structures and their simulation time increases.

Compared with forward-model-based methods, the proposed method excels in real-time applications by supporting end-to-end design. Forward-model-based methods necessitate an additional iterative optimization process during the design stage. As listed in Table III, $0.04\,\text{hours} + 0.003\,\text{hours}$ denotes the validation requiring $0.04\,\text{hours}$ and the extra iterative optimization process requiring $0.003\,\text{hours}$. Although this iterative optimization process is relatively fast ($0.003\,\text{hours}$), it limits the applicability of forward-model-based methods in real-time scenarios. In dynamic environments that change unpredictably and frequently, immediate adjustments to antenna parameters are essential, for example, during the onsite calibration of a phased array antenna. Real-time applications prioritize minimizing the time spent in the design stage, even if only by a small margin. In such scenarios, the proposed method is preferred due to its efficiency.

B. Implementation II: Fourier-phased Metasurface Design

1) Structure of the Fourier-phased Metasurface: The Fourier-phased metasurface is proposed for wideband RCS reduction in [53]. It is composed of 30×30 unit cells with specific reflection phases. The phase distribution is calculated by using the Fourier phase formula. The design requirements can be denoted as $\mathbf{Q} = [f_{min},\ f_{max},\ r_1,\ r_2]$. Here, f_{min} and f_{max} define the desired frequency range. r_1 represents the maximum reflection coefficient within this range, and the default value of r_1 is $-10\,\text{dB}$. r_2 equals the reflection phase calculated using the Fourier phase formula. As shown in Fig. 7, the unit cell contains a cross-shaped patch layer and a ground layer etched on the opposite sides of a substrate layer. The substrate layer has a dielectric constant of $\varepsilon_r = 4.4$ and thickness of $h = 2\,\text{mm}$. The unit cell's thickness h and length p are fixed values. The remaining 7 geometric parameters ($\mathbf{P} = [l,\ m,\ g,\ w,\ r,\ l_r,\ \beta]$) are set as tuning variables to modify the reflection phase. Their tuning ranges are listed in Table IV.

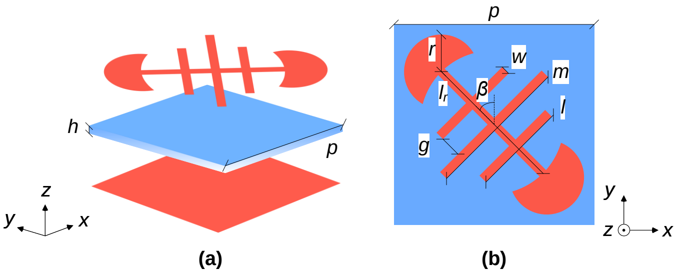


Fig. 7. Structure of the Fourier-phased metasurface.

TABLE IV
IMPLEMENTATION II: TUNABLE AND FIXED GEOMETRIC PARAMETERS FOR FOURIER-PHASED METASURFACE

Parameter	Value	Parameter	Value
l (mm)	[1.10, 2.10]	m (mm)	[2.50, 3.50]
g (mm)	[0.40, 0.60]	w (mm)	[0.10, 0.30]
r (mm)	[0.70, 0.90]	l_r (mm)	[2.50, 3.10]
β (degrees)	[0, 90]	h (mm)	2.00
p (mm)	5		

2) Design Using Proposed Method: 500 sets of tuning variables ($\mathbf{P}_i,\ i \in [1,\ 500]$), EM responses ($|S_{11\_\text{X-Pol.}}|$ and $\arg(S_{11\_\text{Co-Pol.}})$), and EM features ($\mathbf{Q}_i,\ i \in [1, 500]$) are collected to form training, validation, and test datasets at a ratio of $7:2:1$. Each combination of tuning variables is arbitrarily defined within the ranges in Table IV. Its corresponding $|S_{11\_\text{X-Pol.}}|$ and $\arg(S_{11\_\text{Co-Pol.}})$ at 51 discrete frequency points from $5\,\text{GHz}$ to $25\,\text{GHz}$ are simulated via CST. $\mathbf{Q}_i$ describes a set of design requirements that closely match with the i-th $|S_{11\_\text{X-Pol.}}|$ and $\arg(S_{11\_\text{Co-Pol.}})$. $\mathbf{Q}_i$ for $|S_{11\_\text{X-Pol.}}|$ and $\arg(S_{11\_\text{Co-Pol.}})$ is obtained in seven steps (a)-(g):

(a) Set $r_1 = -10\,\text{dB}$;
(b) Find the minimum frequency f_{min} where $|S_{11\_\text{X-Pol.}}| \leq r_1$ by $\min_{f_{min}} |S_{11\_\text{X-Pol.}}|_{(f_{min})} \leq r_1$;
(c) Find the maximum frequency f_{max} where $|S_{11\_\text{X-Pol.}}| \leq r_1$ by $\max_{f_{max}} |S_{11\_\text{X-Pol.}}|_{(f_{max})} \leq r_1$;
(d) Set r_{new} as the maximum $|S_{11\_\text{X-Pol.}}|$ between f_{min} and f_{max};
(e) If $r_{new} > r_1$, update r_1 by $r_1 = r_{new}$, and repeat steps (b)-(e);
(f) Set r_2 as the average value of $\arg(S_{11\_\text{Co-Pol.}})$ between f_{min} and f_{max};
(g) Set $\mathbf{Q}_i = [f_{min}, f_{max}, r_1, r_2]$.

For the normalization of $\mathbf{Q}_i = [\bar{f}_{min}, \bar{f}_{max}, \bar{r_1}, \bar{r_2}]$,

$$\bar{f}_{min} = \frac{f_{min} - 5\,\text{GHz}}{25\,\text{GHz} - 5\,\text{GHz}}, \tag{27}$$

$$\bar{f}_{max} = \frac{f_{max} - 5\,\text{GHz}}{25\,\text{GHz} - 5\,\text{GHz}}, \tag{28}$$

$$\bar{r_1} = \frac{\max\{-10\,\text{dB},\ \max|S_{11\_\text{X-Pol.}}|_{(f_{min},\ f_{max})}\} + 10\,\text{dB}}{10\,\text{dB}}, \tag{29}$$

$$\bar{r_2} = \text{mean}(\mathbf{R}_{2,i_{(f_{min},\ f_{max})}}). \tag{30}$$

The values of the tuning variables are normalized and converted into $\mathbf{P}_i = [\bar{l}, \bar{m}, \bar{g}, \bar{w}, \bar{r}, \bar{l_r}, \bar{\beta}]_i, i \in [1, 500]$. For example, $\bar{l} = \frac{l - \min(l)}{\max(l) - \min(l)}$. The values of $|S_{11\_\text{X-Pol.}}|$ are transformed from decibel (dB) format to linear scale between 0 and 1 and converted into $\mathbf{R}_{1,i} = 10^{\frac{|S_{11\_\text{X-Pol.}}|}{20}}$. The values of $\arg(S_{11\_\text{Co-Pol.}})$ are normalized and converted into

$$\mathbf{R}_{2,i} = \frac{\arg(S_{11\_\text{Co-Pol.}}) - \min(\arg(S_{11\_\text{Co-Pol.}}))}{\max(\arg(S_{11\_\text{Co-Pol.}})) - \min(\arg(S_{11\_\text{Co-Pol.}}))}. \tag{31}$$

The end-to-end model in Implementation II is also optimized using Bayesian optimization with the same search space as described in Implementation I: Linear-to-Circular Polarization Converter Design. The optimal architectures for the encoder, decoder, and forward model are determined after 1000 iterations. The encoder has 5 hidden layer with 16, 256, 64, 32, and 32 neurons. The modified decoder has 2 hidden layers with 64 and 256 neurons. The dimensionality of the Gaussian noise vector is 10. The batch size is 5. The number of epochs is 800. The learning rate is 0.001. The optimizer is Adam. The activation function is the Linear function for the output layer and ReLU for other layers. The optimized forward model has 4 hidden layers of 256, 32, 256, and 256 neurons. The batch size, number of epochs, and learning rate are 15,

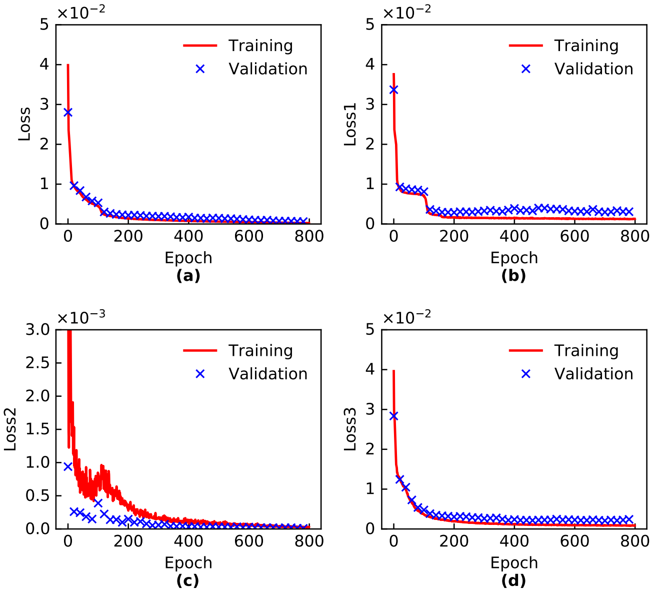


Fig. 8. Training and validation losses of the proposed model over epochs during the training process in Implementation II: Fourier-phased Metasurface Design. (a) $Loss$. (b) $Loss1$. (c) $Loss2$. (d) $Loss3$.

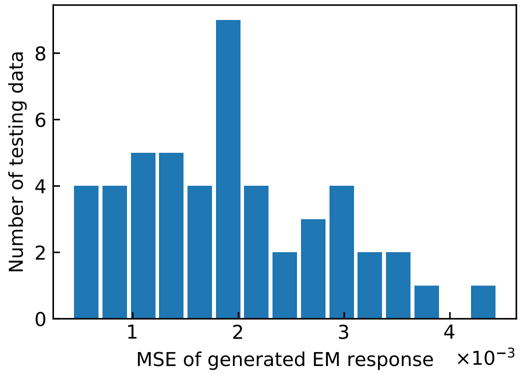


Fig. 9. Test loss of the proposed model in Implementation II: Fourier-phased Metasurface Design.

900, and 0.0001, respectively. The remaining hyperparameters are the same as the encoder and decoder.

After optimization, the forward model is pre-trained using the same datasets, and its accuracy is considered sufficient. The pre-trained forward model is fixed in the following training stage for the proposed end-to-end model. The training and validation losses of the proposed end-to-end model over epochs, as shown in Fig. 8, prove that the model converges well. The final training and validation losses are 1.9×10^{-3} and 4.2×10^{-3}, respectively. Fig. 9 shows the MSE distribution of the generated EM responses on the test dataset, with an average MSE of 2.1×10^{-3} and a maximum MSE of 4.4×10^{-3}. The model accuracy can be improved by increasing the dataset size to $N_{data} > 500$. As a result, model training and, particularly, data collection require more time and incur higher computational costs. Note that the improvement trend weakens gradually.

TABLE V
GEOMETRIC VARIABLES OF DESIGNED UNIT CELLS IN IMPLEMENTATION II: FOURIER-PHASED METASURFACE

Parameter	l (mm)	m (mm)	g (mm)	w (mm)	r (mm)	l_r (mm)	β (degrees)
II-1	1.46	3.34	0.58	0.14	0.82	2.55	31.60
II-2	1.43	3.15	0.60	0.21	0.79	2.57	33.97
II-3	1.49	3.16	0.59	0.18	0.75	2.52	33.76
II-4	1.44	3.17	0.60	0.14	0.75	2.51	32.35

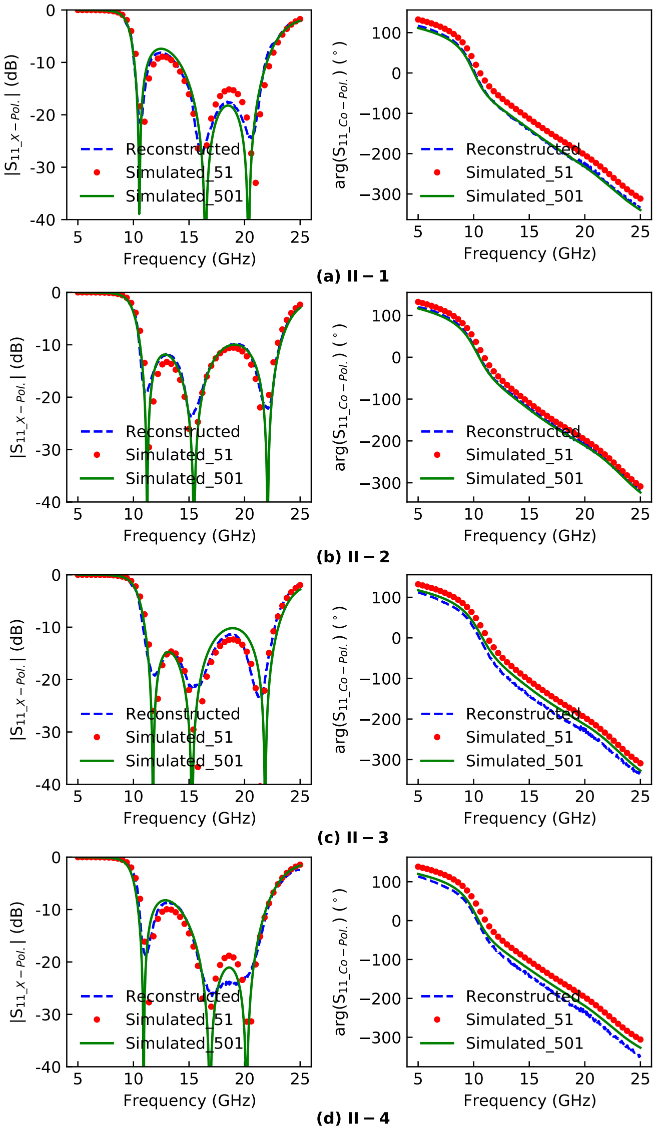


Fig. 10. Reconstructed and Simulated EM responses in Implementation II: Fourier-phased Metasurface Design. (a) II-1. (b) II-2. (c) II-3. (d) II-4. ("Simulated_51" and "Simulated_501" represent results using 51 and 501 frequency points, respectively. The reconstructed results are the same for both 51 and 501 frequency points and are denoted as "Reconstructed".)

The proposed framework is validated in four design scenarios, where the design requirements are predefined as $\mathbf{Q}^*_{1,2,3,4}$,

$$\mathbf{Q}^*_1 = [10.1\,\text{GHz},\ 21.5\,\text{GHz},\ -8\,\text{dB},\ -144^\circ], \tag{32}$$

$$\mathbf{Q}^*_2 = [10.9\,\text{GHz},\ 22.6\,\text{GHz},\ -9\,\text{dB},\ -151^\circ], \tag{33}$$

$$\mathbf{Q}^*_3 = [11.3\,\text{GHz},\ 22.3\,\text{GHz},\ -10\,\text{dB},\ -160^\circ], \tag{34}$$

$$\mathbf{Q}^*_4 = [10.9\,\text{GHz},\ 21.5\,\text{GHz},\ -8.5\,\text{dB},\ -155^\circ]. \tag{35}$$

TABLE VI

COMPARISON BETWEEN THE PROPOSED, PSO-BASED, AND FORWARD-MODEL-BASED METHODS IN IMPLEMENTATION II: FOURIER-PHASED METASURFACE DESIGN (UNIT: HOURS)

	Proposed	**PSO-Based**	**Forward-model-based**
End-to-End	Yes	No	No
Design Performance	Offset $\leq \pm 0.2\,\text{GHz}$	Offset $\leq \pm 0.2\,\text{GHz}$	Offset $\leq \pm 0.2\,\text{GHz}$
Data collection	4.55	-	4.55
Training	0.07	-	0.01
II-1	0.03	1.64	0.033
II-2	0.03	0.54	0.033
II-3	0.03	0.57	0.033
II-4	0.03	0.44	0.033
...	...	...	...
II-n	0.03	...	0.033
Total	$0.03n + 4.62$	$0.517n + 1.123$	$0.033n + 4.56$

The normalized $\mathbf{Q}^*_{1,2,3,4}$ are combined with arbitrarily initialized Gaussian noise vectors $\mathbf{N}^*_n$ and fed into the proposed model to determine four settings of geometric variables, leading to four designed unit cells. Specifically, the $\mathbf{Q}^*_{1,2,3,4}$ are combined with new initialized Gaussian noise vectors $\mathbf{N}^*_n$ ($n \in N_n$) and fed into the decoder to reconstruct $|S_{11\_\text{X-Pol.}}|_n$ and $\arg(S_{11\_\text{Co-Pol.}})_n$ first. Afterward, the reconstructed $|S_{11\_\text{X-Pol.}}|_n$ and $\arg(S_{11\_\text{Co-Pol.}})_n$ are fed into the encoder to generate the geometric variables. N_n is set as 3. For each $\mathbf{Q}^*_{1/2/3/4}$, 3 optimized structures are generated and simulated using CST to evaluate the generated $|S_{11}|_n$s. An optimal structure is determined for each $\mathbf{Q}^*_{1/2/3/4}$. The predicted geometric variables of the optimal structure are listed in Table V. The designed unit cells are simulated via CST to evaluate their $|S_{11\_\text{X-Pol.}}|$ and $\arg(S_{11\_\text{Co-Pol.}})$. The simulated $|S_{11\_\text{X-Pol.}}|$ or $\arg(S_{11\_\text{Co-Pol.}})$ are represented as red dots in Fig. 10, matching well with the predefined constraints.

When increasing the frequency points from 51 to 501, the simulated results, denoted as green solid lines, align better with the predicted results and the predefined practical constraints, reducing the frequency offset from $0.4\,\text{GHz}$ to $0.2\,\text{GHz}$. Better design quality may be realized by further increasing the training dataset size to enhance the model's accuracy. The implementation results show that the proposed model achieves an effective end-to-end design for the Fourier-phased metasurface from practical constraints to satisfactory structures.

3) Comparison with Existing Methods: As discussed in Section III-A3, the existing ML-based methods are not developed for practical end-to-end implementation in Implementation II: Fourier-phased Metasurface Design. To ensure a fair comparison, the PSO-based method is implemented across the four design scenarios $\mathbf{Q}^*_{1/2/3/4}$, using the optimized design for the first design task $\mathbf{Q}^*_1$ as the starting point for the following three tasks $\mathbf{Q}^*_{2,3,4}$. The design process is conducted using the built-in PSO optimizer in CST. The population size is 30, requiring 30 simulation runs per iteration. It follows similar stopping criteria as in Implementation I: Linear-to-Circular Polarization Converter Design.

Table VI compares the time needed in the PSO-based and the proposed methods for the four design tasks. While the proposed method takes $4.55\,\text{hours}$ to prepare the dataset and $0.07\,\text{hours}$ to train the model, the time needed for each design process is reduced to around $0.03\,\text{hours}$ for both design and validation. The PSO-based methods require a long iterative simulation time for each design task, even though the required time is reduced for the following design tasks with the first optimized design as the starting point. By comparison, the proposed method significantly saves overall time and computational costs as the number of design tasks increases. For more complex design tasks requiring longer simulation time, the time of data collection for the proposed method increases. However, the required simulation time during each design task for the PSO-based methods increases faster. In the long term, the proposed method still significantly improves the design efficiency.

The proposed method outperforms forward-model-based methods by enabling end-to-end design. In contrast, forward-model-based methods require an additional iterative optimization process during the design phase. Although this process is relatively quick ($0.003\,\text{hours}$), it restricts the use of forward-model-based methods in real-time scenarios. In these design tasks, environments change unpredictably and frequently, necessitating immediate adjustments to antenna parameters. Real-time applications demand a reduced design time, even by a small margin, highlighting the exceptional capability of the proposed end-to-end methods.

C. Implementation III: Slotline Filter Design

1) Structure of the Slotline Filter Design: The slotline filter described in [54] has been selected as the design target for Implementation III. This filter operates over a frequency range from $1\,\text{GHz}$ to $15\,\text{GHz}$, requiring specific performance in its passband and stopband. Its 3D and top-view representations are depicted in Fig. 11. According to Fig. 11(a), the filter comprises three metallic and two dielectric layers. The top and bottom layers feature two open-ended stepped microstrip lines, as illustrated in Fig. 11(b). The central layer serves as a common ground plane and contains a folded slot resonator, as shown in Fig. 11(c). *Roger's RO4003C* is used as the substrate material, characterized by a dielectric constant of $\varepsilon_r = 3.38$ and a thickness of $0.813\,\text{mm}$. The filter's geometric parameters are detailed in Table VII. Among these parameters, six are chosen as tuning variables for the design process: l_m, w_m, l_{s1}, l_{s2}, l_{s3}, and w_s. The remaining parameters are held constant.

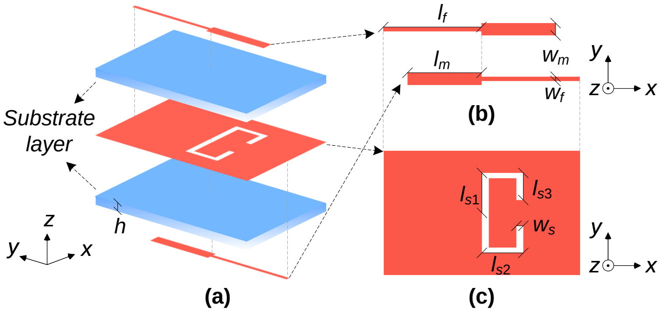


Fig. 11. Structure of the filter.

TABLE VII
IMPLEMENTATION III: TUNABLE AND FIXED GEOMETRIC PARAMETERS FOR FILTER DESIGN

Parameter	Value (mm)	Parameter	Value (mm)
l_m	[4.7, 6.7]	w_m	[1.3, 2.8]
l_{s1}	[9.5, 11.5]	l_{s2}	[1.2, 1.7]
l_{s3}	[2.1, 2.6]	w_s	[0.15, 0.45]
l_f	12.5	w_f	1.8
l	25	w	20
h	0.813		

2) Design Using Proposed Method: We collect training, validation, and test datasets to train the proposed model. 500 sets of geometric variables ($\mathbf{P}_i$, where i ranges from 1 to 500) are defined within their respective tuning ranges, corresponding to 500 distinct structures. These structures are simulated using CST to obtain 500 reflection coefficient curves ($|S_{11}|$ values), showing their filtering performance within the passband and low/high stopband. The geometric variable values are normalized and expressed as $\mathbf{P}_i = [\bar{l_m}, \bar{w_m}, \bar{l_{s1}}, \bar{l_{s2}}, \bar{l_{s3}}, \bar{w_s}]$, where i ranges from 1 to 500. For instance, $\bar{l_m} = \frac{l_m - \min(l_m)}{\max(l_m) - \min(l_m)}$. Each $|S_{11}|$ curve is discretized into a vector of size 1001 ($\mathbf{R}_{1,i}$), containing $|S_{11}|$ values evaluated at 1001 equally spaced frequency points from 1 GHz to 15 GHz. The $|S_{11}|$ values, expressed in decibels (dB), are converted to a linear magnitude scale between 0 and 1, $\mathbf{R}_{1,i} = 10^{\frac{|S_{11}|}{20}}$.

For each $\mathbf{R}_{1,i}$, a corresponding $\mathbf{Q}_i$ is assigned, representing the distinct metrics within the passband and low/high stopband of $\mathbf{R}_{1,i}$. $\mathbf{Q}_i$ for $\mathbf{R}_{1,i}$ is defined through eight steps (a)-(h):

(a) Find the minimum frequency f_{lsb} where $\mathbf{R}_{1,i} \leq -1\,\text{dB}$ by $\min_{f_{lsb}} \mathbf{R}_{1,i(f_{lsb})} \leq -1\,\text{dB}$;
(b) Find the maximum frequency f_{hsb} where $\mathbf{R}_{1,i} \leq -1\,\text{dB}$ by $\max_{f_{hsb}} \mathbf{R}_{1,i(f_{hsb})} \leq -1\,\text{dB}$;
(c) Set $r_1 = -20\,\text{dB}$;
(d) Find the minimum frequency f_{lpb} where $\mathbf{R}_{1,i} \leq -20\,\text{dB}$ by $\min_{f_{lpb}} \mathbf{R}_{1,i(f_{lpb})} \leq r_1$;
(e) Find the maximum frequency f_{hpb} where $\mathbf{R}_{1,i} \leq -20\,\text{dB}$ by $\max_{f_{hpb}} \mathbf{R}_{1,i(f_{hpb})} \leq r_1$;
(f) Set r_{new} as the maximum $\mathbf{R}_{1,i}$ between f_{lpb} and f_{hpb};
(g) If $r_{new} > r_1$, update r_1 by $r_1 = r_{new}$, and repeat steps (c)-(g);
(h) Set $\mathbf{Q}_i = [f_{lsb}, f_{lpb}, f_{hpb}, f_{hsb}, r_1]$.

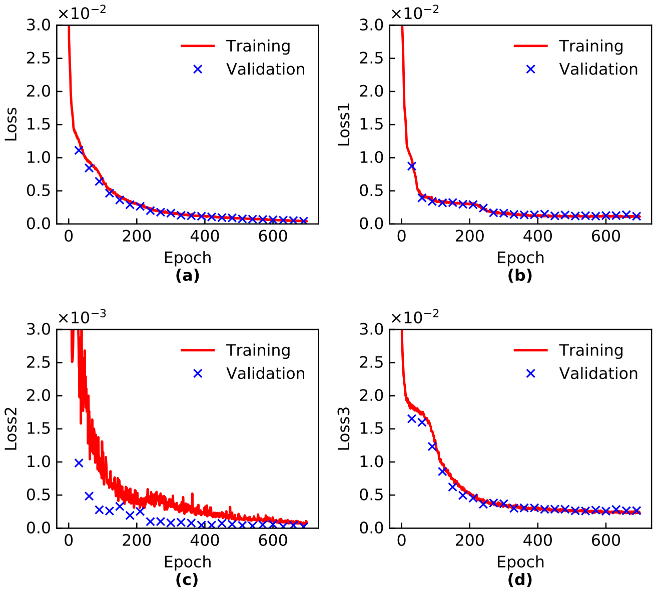


Fig. 12. Training and validation losses of the proposed model over epochs during the training process in Implementation III: Slotline Filter Design. (a) *Loss*. (b) *Loss*1. (c) *Loss*2. (d) *Loss*3.

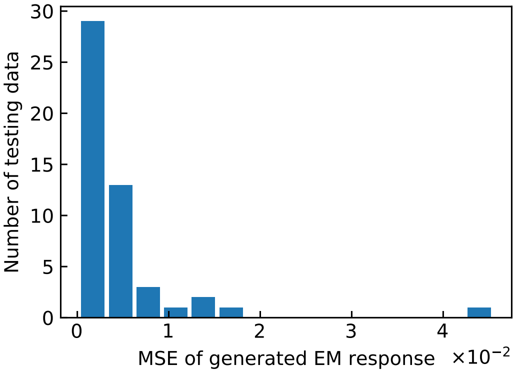


Fig. 13. Test loss of the proposed model in Implementation III: Slotline Filter Design.

The vector $\mathbf{Q}_i = [\bar{f_{lsb}}, \bar{f_{lpb}}, \bar{f_{hpb}}, \bar{f_{hsb}}, \bar{r_1}]$ is normalized using

$$\bar{f}_{lsb/lpb/hpb/hsb} = \frac{f_{lsb/lpb/hpb/hsb} - 1\,\text{GHz}}{15\,\text{GHz} - 1\,\text{GHz}}, \tag{36}$$

$$\bar{r_1} = \frac{\max\{-20\,\text{dB},\ \max|S_{11}|_{(f_{lpb},\ f_{hpb})}\} + 20\,\text{dB}}{20\,\text{dB}}. \tag{37}$$

Here, f_{lpb} and f_{hpb} refer to the frequencies where the passband ($|S_{11}|\ \leq -20\,\text{dB}$) starts and ends, respectively; f_{lsb} indicates the frequency where the low stopband ($|S_{11}| \geq -1\,\text{dB}$) ends; f_{hsb} denotes the frequency where the high stopband ($|S_{11}| \geq -1\,\text{dB}$) begins; r_1 represents the maximum reflection coefficient within the passband. Note that the insertion loss within the low/high stopbands is set at 1 dB. If distinct insertion loss metrics are needed for the low and high stopbands, additional parameters, such as r_2 and r_3, can be introduced to allow for controllable insertion loss in low and high stopbands. The 500 combinations of $\mathbf{P}_i$, $\mathbf{R}_{1,i}$, and $\mathbf{Q}_i$ are arbitrarily divided into training, validation, and test datasets in a ratio of $7:2:1$.

The end-to-end model undergoes optimization using Bayesian techniques, similar to Implementations I and II, with the same defined search space. After 1000 iterations, the optimal architectures for the encoder, decoder, and forward model are established. The encoder consists of five hidden layers with 128, 64, 64, 256, and 32 neurons, respectively. The modified decoder features two hidden layers containing 64 and 256 neurons, respectively. The dimensionality of the Gaussian noise vector is set at 9. The model employs a batch size of 10, runs for 700 epochs, uses a learning rate of 0.001, and utilizes the Adam optimizer. The activation functions are ReLU for the hidden layers and Linear for the output layer. The optimized forward model includes four hidden layers with 128, 128, 256, and 256 neurons, with a batch size of 5, epochs of 1000, and a learning rate of 0.0001, while other hyperparameters remain consistent with the encoder and decoder.

After optimization, the forward model is pre-trained using identical datasets, achieving satisfactory accuracy. This pre-trained forward model is then fixed for subsequent training of the proposed end-to-end model. The training and validation losses across epochs, illustrated in Fig. 12, confirm successful model convergence. Final training and validation losses reach values of 4.05×10^{-4} and 4.16×10^{-4}, respectively. Fig. 13 displays the MSE distribution for generated EM responses on the test dataset, noting an average MSE of 4.3×10^{-3} and a maximum MSE of 4.5×10^{-2}. Enhancing model accuracy can be achieved by expanding the dataset size to more than 500 data points, which necessitates increased time and computational resources for model training and data gathering. It is important to mention that the improvement trend diminishes progressively.

The proposed framework is validated in four design scenarios, where the design requirements are predefined as $\mathbf{Q}^*_{1,2,3,4}$,

$$\mathbf{Q}^*_1 = [2.8\,\text{GHz},\ 4.4\,\text{GHz},\ 8.0\,\text{GHz},\ 10.5\,\text{GHz},\ -20\,\text{dB}], \tag{38}$$

$$\mathbf{Q}^*_2 = [2.9\,\text{GHz},\ 4.5\,\text{GHz},\ 9.8\,\text{GHz},\ 11.4\,\text{GHz},\ -20\,\text{dB}], \tag{39}$$

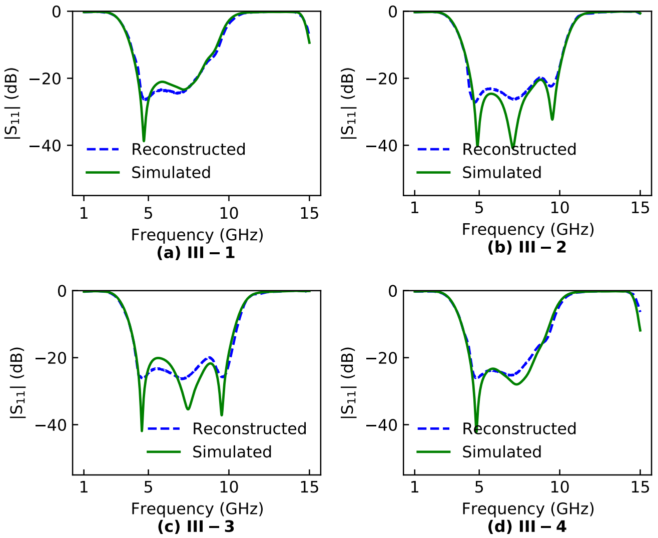
$$\mathbf{Q}^*_3 = [2.8\,\text{GHz},\ 4.3\,\text{GHz},\ 9.9\,\text{GHz},\ 11.6\,\text{GHz},\ -20\,\text{dB}], \tag{40}$$

$$\mathbf{Q}^*_4 = [2.8\,\text{GHz},\ 4.4\,\text{GHz},\ 8.5\,\text{GHz},\ 10.6\,\text{GHz},\ -20\,\text{dB}]. \tag{41}$$

The normalized $\mathbf{Q}^*_{1,2,3,4}$ are combined with arbitrarily initialized Gaussian noise vectors $\mathbf{N}^*_n$ and fed into the proposed model to determine four settings of geometric variables, leading to four designed unit cells. Specifically, the $\mathbf{Q}^*_{1,2,3,4}$ are combined with new initialized Gaussian noise vectors $\mathbf{N}^*_n$ ($n \in N_n$) and fed into the decoder to reconstruct $|S_{11}|_n$ first. Afterward, the reconstructed $|S_{11}|_n$ is fed into the encoder to generate the geometric variables. N_n is set as 3. For each $\mathbf{Q}^*_{1/2/3/4}$, 3 optimized structures are generated and simulated using CST to evaluate the generated $|S_{11}|_n$s. An optimal structure is determined for each $\mathbf{Q}^*_{1/2/3/4}$. The predicted geometric variables of the optimal structure are listed in Table VIII. The designed unit cells are simulated via CST to evaluate their $|S_{11}|$. The simulated $|S_{11}|$ is represented as green solid lines in Fig. 14, matching well with the predefined constraints and reconstructed $|S_{11}|$ denoted as blue dash lines, indicating the effectiveness of the proposed method.

TABLE VIII
GEOMETRIC VARIABLES OF DESIGNED UNIT CELLS IN IMPLEMENTATION III: SLOTLINE FILTER DESIGN

Parameter (mm)	l_m	w_m	l_{s1}	l_{s2}	l_{s3}	w_s
III-1	6.54	1.62	10.17	1.33	2.43	0.21
III-2	6.25	1.75	9.98	1.29	2.32	0.28
III-3	6.21	2.08	10.19	1.34	2.35	0.30
III-4	6.51	1.64	10.11	1.32	2.40	0.22



(a) III − 1 **(b) III − 2** **(c) III − 3** **(d) III − 4**

Fig. 14. Reconstructed and Simulated EM responses in Implementation III: Slotline Filter Design. (a) III-1. (b) III-2. (c) III-3. (d) III-4.

3) Comparison with Existing Methods: As mentioned, the existing ML-based methods have not been developed for practical end-to-end implementation in Implementation III: Slotline Filter Design. To ensure a fair comparison, the PSO-based method is implemented across the four design scenarios $\mathbf{Q}^*_{1/2/3/4}$, using the optimized design for the first design task $\mathbf{Q}^*_1$ as the starting point for the following three tasks $\mathbf{Q}^*_{2,3,4}$. The design process is conducted using the built-in PSO optimizer in CST. The population size is 30, requiring 30 simulation runs per iteration. It follows similar stopping criteria as in Implementation I and II.

Table IX compares the time needed in the PSO-based and the proposed methods for the four design tasks. As the simulation complexity of the slotline filter in Implementation III increases, the proposed method takes 56.75 hours to prepare the dataset and 0.043 hours to train the model. The time needed for each design process is reduced to around 0.34 hours for both design and validation. The PSO-based methods require a long iterative simulation time for each design task, even though the required time reduces for the following design tasks with the first optimized design as the starting point (17.02 hours for the first design task, on average 10.07 hours for the following design tasks). By comparison, the required time and computational costs of the PSO-based methods increase faster

TABLE IX
COMPARISON BETWEEN THE PROPOSED, PSO-BASED, AND FORWARD-MODEL-BASED METHODS IN IMPLEMENTATION III: SLOTLINE FILTER DESIGN (UNIT: HOURS)

	Proposed	**PSO-Based**	**Forward-model-based**
End-to-End	Yes	No	No
Design Performance	Offset $\leq \pm 0.1$ GHz	Offset $\leq \pm 0.1$ GHz	Offset $\leq \pm 0.1$ GHz
Data collection	56.75	-	56.75
Training	0.043	-	0.04
III-1	0.34	17.02	0.344
III-2	0.34	10.35	0.344
III-3	0.34	10.78	0.344
III-4	0.34	9.08	0.344
...	...	...	...
III-n	0.34	...	0.344
Total	$0.34n + 56.793$	$10.07n + 6.95$	$0.344n + 56.79$

as the number of design tasks increases, indicating the superior design efficiency of the proposed method.

The proposed method surpasses forward-model-based methods by facilitating end-to-end design. Unlike forward-model-based approaches, which require a supplementary iterative optimization process, the proposed method is better suited for real-time scenarios where environments change unpredictably and frequently, necessitating swift adjustments to antenna parameters. Real-time applications benefit from reduced design time, underscoring the superior efficiency of the proposed end-to-end method. Forward-model-based methods are better suited for sensitivity analysis and exploring a wide range of design spaces.

In the long run, the proposed design framework significantly reduces computational costs and improves design efficiency. For example, wireless communication terminal manufacturers face high costs and lengthy timelines in designing and updating antennas for various products, such as smartphones and tablets, using traditional methods that require extensive EM simulations. By implementing the proposed framework, the companies can significantly reduce these burdens. Their EM properties and past antenna designs form a comprehensive dataset to train the MCVAE model. The developed MCVAE model allows for rapid end-to-end design and update, reducing the time from weeks to mere minutes per design at a fraction of the cost. When meeting industry standards with more restrictive design specifications, significant data are required for training, and sophisticated architectures are needed. Advanced machine learning techniques and high-quality data acquisition can be integrated to ensure convergence efficiency; parallel computation, transfer learning, and small-sample machine learning can be incorporated to balance the trade-offs between maintaining the performance and reducing computational costs. Over the long term, the proposed framework saves the company substantial financial and temporal resources, enabling faster market responsiveness and fostering innovation in antenna design.

IV. DISCUSSION

We explore the approach's capability of incorporating multiple design objectives and the impact of expanded geometric parameters by conducting three additional implementations. Their implementation processes are briefly introduced for the sake of conciseness.

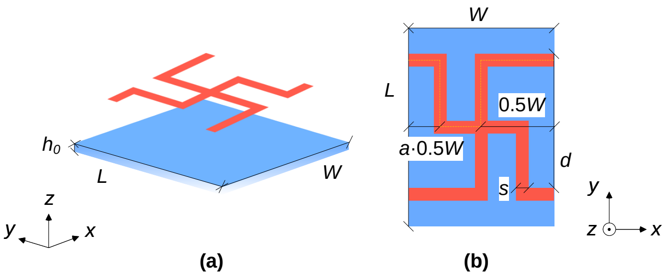


Fig. 15. Structure of the linear-to-circular loop polarization converter. ($L \in [3, 5.5]$mm, $W \in [2, 4.5]$mm, $a \in [0.2, 0.8]$, $d \in [1.5, 2.8]$mm, and $s \in [0.1, 0.5]$mm.)

TABLE X
GEOMETRIC VARIABLES OF DESIGNED LINEAR-TO-CIRCULAR LOOP POLARIZATION CONVERTER

Parameter	L (mm)	W (mm)	a	d (mm)	s (mm)
IV-1	3.89	2.13	0.22	2.52	0.12
IV-2	3.60	2.37	0.24	2.27	0.17
IV-3	3.15	2.15	0.64	2.26	0.20
IV-4	3.48	1.96	0.49	2.15	0.22

A. Multiple Independent Design Objectives

The proposed design framework is scalable to incorporate multiple independent design objectives. Significant independent design objectives can be integrated by following this procedure:

1) Collect sufficient data that necessarily represent the entire solving space;
2) Extend the feature vector $\mathbf{Q}$ to cover all the design objectives;
3) Upgrade the model's architecture by increasing the hidden layers and neurons and optimize the hyperparameters to be compatible with the increasing data;
4) Get powerful computational resources to train the model.

Thus, the maximum number of independent design objectives that our approach can handle depends on the available computational power.

For validation, we implement this approach to design a linear-to-circular loop polarization converter in [55] with two

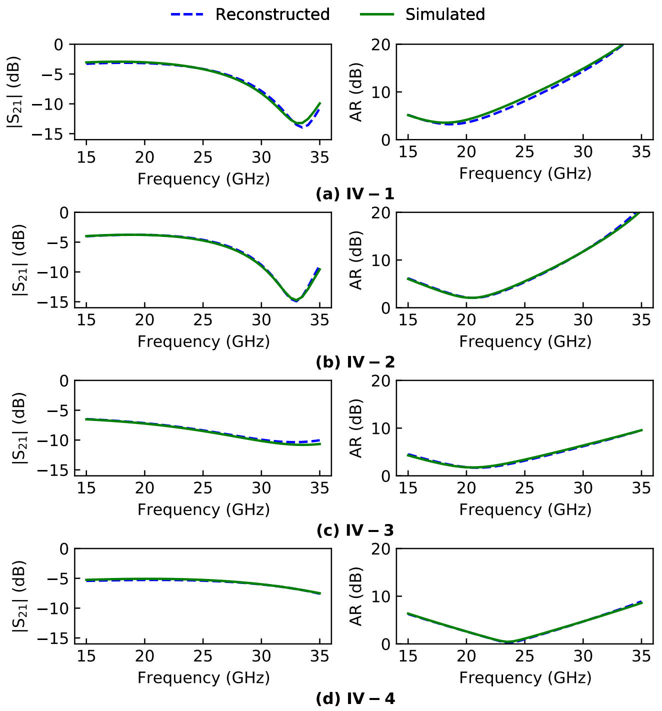


Fig. 16. Reconstructed and Simulated EM responses for the linear-to-circular loop polarization converter. (a) IV-1. (b) IV-2. (c) IV-3. (d) IV-4.

independent design objectives, which are the transmission coefficients and axial ratio from 15 GHz to 35 GHz. The tuning geometric parameters include L, W, a, d, and s, as shown in Fig. 15. 300 pairs of data are collected to form a training, validation, and test dataset in a ratio of $7:2:1$. The feature vector $\mathbf{Q}$ is defined as $\mathbf{Q} = [f_{min}, f_{max}, r_1, r_2]$, where f_{min} and f_{max} confine the operating frequency range, r_1 and r_2 are the mean transmission coefficient and maximum axial ratio over this range. The optimized model has 2 hidden layers with $[128, 256]$ neurons. The number of epochs, learning rate, and batch size are 800, 0.0001, and 10, respectively. The respective mean and max test loss are 2.6×10^{-4} and 1.1×10^{-3}. Four design specifications are initialized as $\mathbf{Q}^*_{1,2,3,4}$,

$$\mathbf{Q}^*_1 = [18.5\,\text{GHz},\ 20.0\,\text{GHz},\ -3.5\,\text{dB},\ 3\,\text{dB}], \tag{42}$$

$$\mathbf{Q}^*_2 = [18.0\,\text{GHz},\ 23.0\,\text{GHz},\ -4.0\,\text{dB},\ 3\,\text{dB}], \tag{43}$$

$$\mathbf{Q}^*_3 = [17.0\,\text{GHz},\ 25.0\,\text{GHz},\ -7.5\,\text{dB},\ 3\,\text{dB}], \tag{44}$$

$$\mathbf{Q}^*_4 = [19.0\,\text{GHz},\ 28.0\,\text{GHz},\ -5.0\,\text{dB},\ 3\,\text{dB}]. \tag{45}$$

Fig. 16 compares reconstructed and simulated EM responses of four designed structures in Table X. Their agreement validates the effectiveness of the approach in incorporating two independent design objectives.

Compared with the six-dimensional polarizer design with one objective in Implementation I, this five-dimensional design with two objectives maintains the performance using the same amount of data. It indicates that the maximum number of design objectives is inversely proportional to the geometric dimensionality at fixed computational costs.

With finite computational power, the maximum number of independent design objectives (N_{input}) that our approach can handle is inversely proportional to the geometric dimensionality (N_{output}). Geometric dimensionality (N_{output}) denotes the number of tunable geometric parameters of the desired EM structures. The geometric dimensionality (N_{output}) decides the output dimensionality, and the maximum number of independent design objectives (N_{input}) decides the input dimensionality. $N_{input} \times N_{output}$ defines the dimensionality of the design space, showing the design complexity. The maximum number of independent design objectives reduces as the geometric dimensionality increases.

TABLE XI
Implementation V: Double-Expanded Parameter Range for Linear-to-Circular Polarization Converter Design

Parameter	Value (mm)	Parameter	Value (mm)
p_{1x}	[2.10, 3.70]	p_{1y}	[0.20, 1.80]
w_{2x}	[0.10, 1.70]	w_{2y}	[0.10, 0.50]
p_{3x}	[1.70, 3.30]	p_{3y}	[0.80, 2.40]

TABLE XII
Implementation VI: Triple-Expanded Parameter Range for Linear-to-Circular Polarization Converter Design

Parameter	Value (mm)	Parameter	Value (mm)
p_{1x}	[1.70, 4.10]	p_{1y}	[0.10, 2.50]
w_{2x}	[0.10, 2.50]	w_{2y}	[0.10, 0.70]
p_{3x}	[1.30, 3.70]	p_{3y}	[0.40, 2.80]

TABLE XIII
Geometric Variables of Designed Unit Cells in Implementation V: Linear-to-Circular Polarization Converter Design with Double-Expanded Parameter Range

Parameter (mm)	p_{1x}	p_{1y}	w_{2x}	w_{2y}	p_{3x}	p_{3y}
V-1	2.49	0.62	1.24	0.24	2.22	1.58
V-2	2.65	1.16	1.34	0.30	2.58	1.54
V-3	2.50	0.86	0.78	0.25	2.64	1.39
V-4	2.68	0.77	1.57	0.38	2.86	1.64

TABLE XIV
Geometric Variables of Designed Unit Cells in Implementation VI: Linear-to-Circular Polarization Converter Design with Triple-Expanded Parameter Range

Parameter (mm)	p_{1x}	p_{1y}	w_{2x}	w_{2y}	p_{3x}	p_{3y}
VI-1	2.90	1.72	1.36	0.23	2.62	1.27
VI-2	2.83	1.79	1.68	0.29	2.42	1.28
VI-3	3.11	2.09	1.11	0.39	2.89	1.44
VI-4	2.66	1.53	2.00	0.30	2.76	1.92

B. *Wide Geometric Parameter Ranges*

To evaluate the effects of design parameter ranges on our approach, we conduct Implementation V and VI with double-expanded and triple-expanded parameter ranges, respectively, compared with Implementation I: Linear-to-Circular Polarization Converter Design. Their expanded ranges are listed in Table XI and XII, respectively. The model's architecture remains unchanged. The overall dataset size is increased to 500 and 800, respectively. Initialized design requirements for

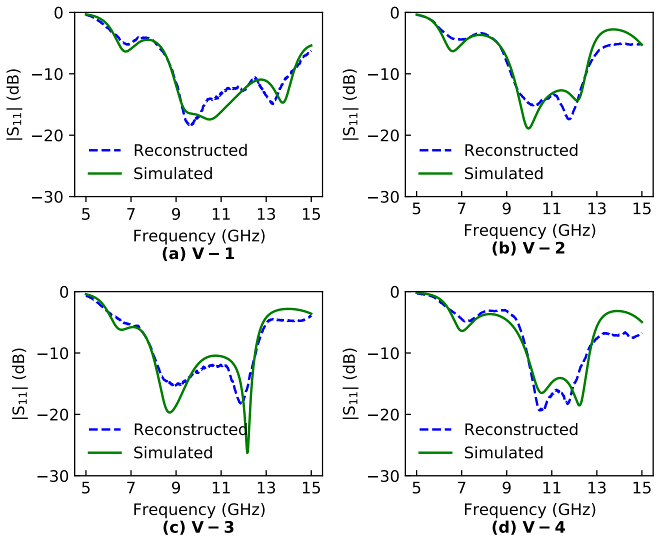


Fig. 17. Reconstructed and Simulated EM responses in Implementation V: Linear-to-Circular Polarization Converter Design with Double-Expanded Parameter Range. (a) V-1. (b) V-2. (c) V-3. (d) V-4.

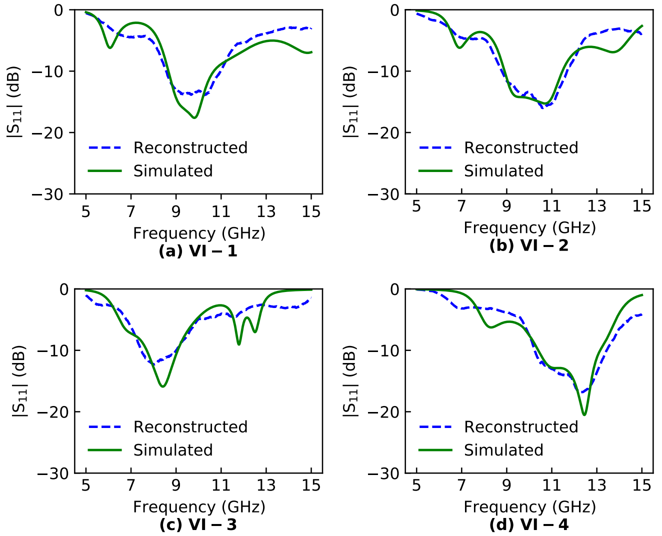


Fig. 18. Reconstructed and Simulated EM responses in Implementation VI: Linear-to-Circular Polarization Converter Design with Triple-Expanded Parameter Range. (a) VI-1. (b) VI-2. (c) VI-3. (d) VI-4.

Implementation V are

$$\mathbf{Q}_1^* = [8.9\,\text{GHz},\ 14.2\,\text{GHz},\ -10\,\text{dB}], \tag{46}$$
$$\mathbf{Q}_2^* = [9.3\,\text{GHz},\ 12.5\,\text{GHz},\ -10\,\text{dB}], \tag{47}$$
$$\mathbf{Q}_3^* = [8.0\,\text{GHz},\ 12.5\,\text{GHz},\ -10\,\text{dB}], \tag{48}$$
$$\mathbf{Q}_4^* = [9.9\,\text{GHz},\ 12.7\,\text{GHz},\ -10\,\text{dB}]. \tag{49}$$

Initialized design requirements for Implementation VI are

$$\mathbf{Q}_1^* = [8.7\,\text{GHz},\ 10.6\,\text{GHz},\ -10\,\text{dB}], \tag{50}$$
$$\mathbf{Q}_2^* = [8.9\,\text{GHz},\ 11.4\,\text{GHz},\ -10\,\text{dB}], \tag{51}$$
$$\mathbf{Q}_3^* = [7.7\,\text{GHz},\ 9.1\,\text{GHz},\ -10\,\text{dB}], \tag{52}$$
$$\mathbf{Q}_4^* = [10.4\,\text{GHz},\ 13.1\,\text{GHz},\ -10\,\text{dB}]. \tag{53}$$

Table XIII and XIV show the geometric parameters of designed structures. Fig. 17 and 18 compare their respective generated responses.

The comparative results demonstrate that our framework maintains the performance by increasing the amount of data accordingly, when the design parameter range is expanded. The simulated responses align well with the design specifications, although there is a difference between the simulated and reconstructed curves. This difference might be attributed to the increasing proportion of low-quality data occupied in the expanded parameter sampling space. Similarly, increasing the parameter dimensionality could also raise the low-quality issue and increase the number of data needed for training. Significant data are needed for an even wider parameter range and higher dimensionality, and the model should be scaled up accordingly, requiring extensive computational resources. Besides increasing the number of samples, a more efficient solution is incorporating high-quality data acquisition techniques. Our previous work [32], [33] presented its capability of maintaining the performance while greatly reducing the computational costs by leveraging a performance-oriented dynamic sampling strategy. Therefore, the proposed MCVAE integrated with high-quality data acquisition techniques can be an alternative for designing EM structures with extended parameter range and dimensionality.

V. CONCLUSION

The end-to-end electromagnetic (EM) structure design methods are preferred because they eliminate the need for iterative trial-and-error processes, showing superior efficiency and higher availability for real-time onsite design scenarios. Existing end-to-end methods either demand impractical input information or suffer from non-uniqueness effects. This paper proposes a modified conditional variational autoencoder (MCVAE) that achieves end-to-end design from practical constraints to optimized EM structures, simultaneously solving the impracticality issue and mitigating the effects of non-uniqueness through robust latent space exploration and a forward-model-based loss function. We validate its effectiveness in multiple implementations: a linear-to-circular polarization converter design, a Fourier-phased metasurface design, a slotline filter design, a loop polarization converter design, and a polarization converter design with double and triple-expanded parameter ranges. The proposed design framework can be extended to facilitate the end-to-end design of other types of EM structures, bridging diverse practical constraints to various optimized structure designs.

REFERENCES

[1] S. Cui and D. S. Weile, "Robust design of absorbers using genetic algorithms and the finite element-boundary integral method," *IEEE Trans. Antennas Propag.*, vol. 51, no. 12, pp. 3249-3258, Dec. 2003, doi: 10.1109/TAP.2003.820971.

[2] O. Ozgun, S. Mutlu, M. I. Aksun, and L. Alatan, "Design of dual-frequency probe-fed microstrip antennas with genetic optimization algorithm," *IEEE Trans. Antennas Propag.*, vol. 51, no. 8, pp. 1947-1954, Aug. 2003, doi: 10.1109/TAP.2003.814732.

[3] R. L. Haupt, "Adaptive crossed dipole antennas using a genetic algorithm," *IEEE Trans. Antennas Propag.*, vol. 52, no. 8, pp. 1976-1982, Aug. 2004, doi: 10.1109/TAP.2004.832493.

[4] F. Afshinmanesh, A. Marandi, and M. Shahabadi, "Design of a single-feed dual-band dual-polarized printed microstrip antenna using a boolean particle swarm optimization," *IEEE Trans. Antennas Propag.*, vol. 56, no. 7, pp. 1845-1852, Jul. 2008, doi: 10.1109/TAP.2008.924684.

[5] S. Chamaani, S. A. Mirtaheri, and M. S. Abrishamian, "Improvement of time and frequency domain performance of antipodal Vivaldi antenna using multi-objective particle swarm optimization," *IEEE Trans. Antennas Propag.*, vol. 59, no. 5, pp. 1738-1742, May 2011, doi: 10.1109/TAP.2011.2122290.

[6] A. A. Minasian and T. S. Bird, "Particle swarm optimization of microstrip antennas for wireless communication systems," *IEEE Trans. Antennas Propag.*, vol. 61, no. 12, pp. 6214-6217, Dec. 2013, doi: 10.1109/TAP.2013.2281517.

[7] P. Nayeri, F. Yang, and A. Z. Elsherbeni, "Design of single-feed reflectarray antennas with asymmetric multiple beams using the particle swarm optimization method," *IEEE Trans. Antennas Propag.*, vol. 61, no. 9, pp. 4598-4605, Sep. 2013, doi: 10.1109/TAP.2013.2268243.

[8] S. Koziel and M. Abdullah, "Machine-learning-powered EM-based framework for efficient and reliable design of low scattering metasurfaces," *IEEE Trans. Microw. Theory Techn.*, vol. 69, no. 4, pp. 2028-2041, Mar. 2021, doi: 10.1109/TMTT.2021.3061128.

[9] D. R. Prado, J. A. López-Fernández, M. Arrebola, M. R. Pino, and G. Goussetis, "Wideband shaped-beam reflectarray design using support vector regression analysis," *IEEE Antennas Wireless Propag. Lett.*, vol. 18, no. 11, pp. 2287-2291, Nov. 2019, doi: 10.1109/LAWP.2019.2932902.

[10] J. P. Jacobs, S. Koziel, and S. Ogurtsov, "Computationally efficient multi-fidelity Bayesian support vector regression modeling of planar antenna input characteristics," *IEEE Trans. Antennas Propag.*, vol. 61, no. 2, pp. 980-984, Feb. 2013, doi: 10.1109/TAP.2012.2220513.

[11] Y. He, J. Huang, W. Li, L. Zhang, S. Wong, and Z. Chen "Hybrid method of artificial neural network and simulated annealing algorithm for optimizing wideband patch antennas," *IEEE Trans. Antennas Propag.*, vol. 72, no. 1, pp. 944-949, Jan. 2024, doi: 10.1109/TAP.2023.3331249.

[12] J. P. Jacobs and S. Koziel, "Two-stage framework for efficient Gaussian process modeling of antenna input characteristics," *IEEE Trans. Antennas Propag.*, vol. 62, no. 2, pp. 706-713, Feb. 2014, doi: 10.1109/TAP.2013.2290121.

[13] A. Pietrenko-Dabrowska, S. Koziel, and U. Ullah, "Reduced-cost two-level surrogate antenna modeling using domain confinement and response features," *Sci. Rep.*, vol. 12, no. 1, Art. no. 1, Mar. 2022, doi: 10.1038/s41598-022-08710-2.

[14] B. Liu, H. Aliakbarian, Z. Ma, G. A. E. Vandenbosch, G. Gielen, and P. Excell, "An efficient method for antenna design optimization based on evolutionary computation and machine learning techniques," *IEEE Trans. Antennas Propag.*, vol. 62, no. 1, pp. 7-18, Jan. 2014, doi: 10.1109/TAP.2013.2283605.

[15] Q. Wu, H. Wang, and W. Hong, "Multistage collaborative machine learning and its application to antenna modeling and optimization," *IEEE Trans. Antennas Propag.*, vol. 68, no. 5, pp. 3397-3409, May 2020, doi: 10.1109/TAP.2019.2963570.

[16] L.-Y. Xiao, W. Shao, X. Ding, and B.-Z. Wang, "Dynamic adjustment kernel extreme learning machine for microwave component design," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 10, pp. 4452-4461, Oct. 2018, doi: 10.1109/TMTT.2018.2858787.

[17] W. Zhang, F. Feng, V. Gongal-Reddy, J. Zhang, S. Yan, J. Ma, and Q. Zhang, "Space mapping approach to electromagnetic centric multiphysics parametric modeling of microwave components," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 7, pp. 3169-3185, Jul. 2018, doi: 10.1109/TMTT.2018.2832120.

[18] D. R. Prado, J. A. López-Fernández, M. Arrebola, and G. Goussetis, "Support vector regression to accelerate design and crosspolar optimization of shaped-beam reflectarray antennas for space applications," *IEEE Trans. Antennas Propag.*, vol. 67, no. 3, pp. 1659-1668, Mar. 2019, doi: 10.1109/TAP.2018.2889029.

[19] Q. Wu, W. Chen, C. Yu, H. Wang, and W. Hong, "Multilayer machine learning-assisted optimization-based robust design and its applications to antennas and array," *IEEE Trans. Antennas Propag.*, vol. 69, no. 9, pp. 6052-6057, Sep. 2021, doi: 10.1109/TAP.2021.3069491.

[20] W. T. Li, H. S. Tang, C. Cui, Y. Q. Hei, and X. W. Shi, "Efficient online data-driven enhanced-XGBoost method for antenna optimization," *IEEE Trans. Antennas Propag.*, vol. 70, no. 7, pp. 4953-4964, Jul. 2022, doi: 10.1109/TAP.2022.3157895.

[21] Z. Zhou, Z. Wei, J. Ren, Y. Yin, G. F. Pedersen, and M. Shen, "Representation learning-driven fully automated framework for the inverse design of frequency-selective surfaces," *IEEE Trans. Microw. Theory Techn.*, vol. 71, no. 6, pp. 2409-2421, Jun. 2023, doi: 10.1109/TMTT.2023.3235066.

[22] J. Li, W. Qiu, P. Xiao, Z. Liu, G. Li, W. T. Joines, and S. Liao, "An adaptive evolutionary neural network-based optimization design method for wideband dual-polarized antennas," *IEEE Trans. Antennas Propag.*, vol. 71, no. 10, pp. 8165-8172, Oct. 2023, doi: 10.1109/TAP.2023.3301882.

[23] Q. Wu, W. Chen, C. Yu, H. Wang, and W. Hong, "Machine learning-assisted optimization for antenna geometry design," *IEEE Trans. Antennas Propag.*, vol. 72, no. 3, pp. 2083-2095, Mar. 2024, doi: 10.1109/TAP.2023.3346493.

[24] Z. Zhou, Z. Wei, J. Ren, Y. Yin, G. F. Pedersen, and M. Shen, "Two-order deep learning for generalized synthesis of radiation patterns for antenna arrays," *IEEE Trans. Artif. Intell.*, vol. 4, no. 5, pp. 1359-1368, Oct. 2023, doi: 10.1109/TAI.2022.3192505.

[25] J. A. Hodge, K. V. Mishra, and A. I. Zaghloul, "RF metasurface array design using deep convolutional generative adversarial networks," in *IEEE Int. Symp. Phased Array Syst. Technol.*, Oct. 2019, pp. 1-6, doi: 10.1109/PAST43306.2019.9020794.

[26] Z. Wei, Z. Zhou, P. Wang, J. Ren, Y. Yin, G. F. Pedersen, and M. Shen, "Equivalent circuit theory-assisted deep learning for accelerated generative design of metasurfaces," *IEEE Trans. Antennas Propag.*, vol. 70, no. 7, pp. 5120-5129, Feb. 2022, doi: 10.1109/TAP.2022.3152592.

[27] P. Naseri and S. V. Hum, "A generative machine learning-based approach for the inverse design of multilayer metasurfaces," *IEEE Trans. Antennas Propag.*, vol. 69, no. 9, pp. 5725-5739, Feb. 2021, doi: 10.1109/TAP.2021.3060142.

[28] P. Naseri, G. Goussetis, N. J. G. Fonseca, and S. V. Hum, "Synthesis of multi-band reflective polarizing metasurfaces using a generative adversarial network," *Sci. Rep.*, vol. 12, no. 1, pp. 1-14, Oct. 2022, doi: 10.1038/s41598-022-20851-y.

[29] Z. Wei, Z. Zhou, P. Wang, J. Ren, Y. Yin, G. F. Pedersen, and M. Shen, "Fully automated design method based on reinforcement learning and surrogate modeling for antenna array decoupling," *IEEE Trans. Antennas Propag.*, vol. 71, no. 1, pp. 660-671, Jan. 2023, doi: 10.1109/TAP.2022.3221613.

[30] Z. Wei, Z. Zhou, P. Wang, J. Ren, Y. Yin, G. F. Pedersen, and M. Shen, "Automated antenna design via domain knowledge-informed reinforcement learning and imitation learning," *IEEE Trans. Antennas Propag.*, vol. 71, no. 7, pp. 5549-5557, Jul. 2023, doi: 10.1109/TAP.2023.3266051.

[31] Z. Wei, Z. Zhou, P. Wang, J. Ren, Y. Yin, G. F. Pedersen, and M. Shen, "Fast and automatic parametric model construction of antenna structures using CNN–LSTM networks", *IEEE Trans. Antennas Propag.*, vol. 72, no. 2, pp. 1319-1328, Feb. 2024, doi: 10.1109/TAP.2023.3346050.

[32] Z. Zhou, Z. Wei, A. Tahir, J. Ren, Y. Yin, G. F. Pedersen and M. Shen, "A high-quality data acquisition method for machine learning-based design and analysis of electromagnetic structures," *IEEE Trans. Microw. Theory Techn.*, vol. 71, no. 10, pp. 4295-4306, Mar. 2023, doi: 10.1109/TMTT.2023.3235066.

[33] Z. Zhou, Z. Wei, J. Ren, Y. Yin, G. F. Pedersen and M. Shen, "Bayesian-inspired sampling for efficient machine-learning-assisted microwave component design," *IEEE Trans. Microw. Theory Techn.*, vol. 72, no. 2, pp. 996-1007, Aug. 2023, doi: 10.1109/TMTT.2023.3298194.

[34] Y.-F. Liu, L. Peng, and W. Shao, "An efficient knowledge-based artificial neural network for the design of circularly polarized 3-D-printed lens antenna," *IEEE Trans. Antennas Propag.*, vol. 70, no. 7, pp. 5007-5014, Jul. 2022, doi: 10.1109/TAP.2022.3140313.

[35] L.-Y. Xiao, W. Shao, F.-L. Jin, B.-Z. Wang, and Q. H. Liu, "Inverse artificial neural network for multiobjective antenna design," *IEEE Trans. Antennas Propag.*, vol. 69, no. 10, pp. 6651-6659, Oct. 2021, doi: 10.1109/TAP.2021.3069543.

[36] M. Sedaghat, R. Trinchero, Z. H. Firouzeh, and F. G. Canavero, "Compressed machine learning-based inverse model for design optimization of microwave components," *IEEE Trans. Microw. Theory Techn.*, vol. 70, no. 7, pp. 3415-3427, Jul. 2022, doi: 10.1109/TMTT.2022.3166151.

[37] Z. Zhou, Z. Wei, J. Ren, Y. Yin, G. F. Pedersen, and M. Shen, "Transfer learning assisted multi-element calibration for active phased antenna arrays," *IEEE Trans. Antennas Propag.*, vol. 71, no. 2, pp. 1982-1987, Oct. 2022, doi: 10.1109/TAP.2022.3216548.

[38] T. Brown, C. Narendra, Y. Vahabzadeh, C. Caloz, and P. Mojabi, "On the use of electromagnetic inversion for metasurface design," *IEEE Trans. Antennas Propag.*, vol. 68, no. 3, pp. 1812-1824, Mar. 2020, doi: 10.1109/TAP.2019.2944544.

[39] Z. Gu, D. Li, Y. Wu, Y. Fan, C. Yu, and H. Chen, "A solution to the dilemma for FSS inverse design using generative models," *IEEE Trans. Antennas Propag.*, vol. 71, no. 6, pp. 5100-5109, Jun. 2023, doi: 10.1109/TAP.2023.3266053.

[40] C. C. Nadell, B. Huang, J. M. Malof, and W. J. Padilla, “Deep learning for accelerated all-dielectric metasurface design,” *Opt. Express*, vol. 27, no. 20, pp. 27523-27535, Sep. 2019, doi: 10.1364/OE.27.027523.

[41] C. Zhang, J. Jin, W. Na, Q. J. Zhang, and M. Yu, “Multivalued neural network inverse modeling and applications to microwave filters,” *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 8, pp. 3781-3797, Aug. 2018, doi: 10.1109/TMTT.2018.2841889.

[42] L. Yuan, L. Wang, X. S. Yang, H. Huang, and B. Z. Wang, “An efficient artificial neural network model for inverse design of metasurfaces,” *IEEE Antennas Wireless Propag. Lett.*, vol. 20, no. 6, pp. 2013-2017, Jun. 2021, doi: 10.1109/LAWP.2021.3069713.

[43] M. Pesteie, P. Abolmaesumi, and R. N. Rohling, “Adaptive augmentation of medical data using independently conditional variational autoencoders,” *IEEE Trans. Med. Imag.*, vol. 38, no. 12, pp. 2807–2820, Dec. 2019, doi: 10.1109/TMI.2019.2914656.

[44] X. Wang, K. Tan, Q. Du, Y. Chen, and P. Du, “CVA^{2}E: A conditional variational autoencoder with an adversarial training process for hyperspectral imagery classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 8, pp. 5676–5692, Aug. 2020, doi: 10.1109/TGRS.2020.2968304.

[45] J. Wu, F. Shen, and L. Ye, “Data augmentation using time conditional variational autoencoder for soft sensor of industrial processes with limited data,” *IEEE Trans. Instrum. Meas.*, vol. 73, pp. 2524714–2524727, Jul. 2024, doi: 10.1109/TIM.2024.3427765.

[46] G. Mu and J. Chen, “Developing a conditional variational autoencoder to guide spectral data augmentation for calibration modeling,” *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 2501008– 2501015, Jan. 2022, doi: 10.1109/TIM.2022.3142060.

[47] Y. Chen, Y. Yang, X. Pan, X. Meng, and J. Hu, “Spatiotemporal fusion network for land surface temperature based on a conditional variational autoencoder,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 5002813–5002825, Jun. 2022, doi: 10.1109/TGRS.2022.3183114.

[48] R. Jiao, K. Peng, and J. Dong, “Remaining useful life prediction of lithium-ion batteries based on conditional variational autoencoders-particle filter,” *IEEE Trans. Instrum. Meas.*, vol. 69, no. 11, pp. 8831–8843, Nov. 2020, doi: 10.1109/TIM.2020.2996004.

[49] J. Wang, R. Li, C. He, H. Chen, R. Cheng, C. Zhai, and M. Zhang, “An inverse design method for supercritical airfoil based on conditional generative models,” *Chin. J. Aeronaut.*, vol. 35, no. 3, pp. 62–74, Mar. 2022, doi: 10.1016/j.cja.2021.03.006.

[50] M. Lee and K. Min, “MGCVAE: multi-objective inverse design via molecular graph conditional variational autoencoder,” *J. Chem. Inf. Model.*, vol. 62, no. 12, pp. 2943–2950, Jun. 2022, doi: 10.1021/acs.jcim.2c00487.

[51] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Int. Conf. Learn. Repr.*, Y. Bengio and Y. LeCun, Eds., 2015, pp. 1-15.

[52] S. M. A. M. H. Abadi and N. Behdad, “Wideband linear-to-circular polarization converters based on miniaturized-element frequency selective surfaces,” in *IEEE Trans. Antennas Propag.*, vol. 64, no. 2, pp. 525-534, Feb. 2016, doi: 10.1109/TAP.2015.2504999.

[53] M. K. T. Al-Nuaimi, G. Huang, W. G. Whittow, R. Chen, S. Wong, “Realization of single-layer Fourier phased metasurfaces for wideband RCS reduction,” *IEEE Antennas Wire. Propag. Lett.*, vol. 22, no. 5, pp. 1179-1183, May 2023, doi: 10.1109/LAWP.2023.3235970.

[54] L. Yang, L. Zhu, W. Choi, and K. Tam, “Analysis and design of wideband microstrip-to-microstrip equal ripple vertical transitions and their application to bandpass filters,” *IEEE Trans. Microw. Theory Techn.*, vol. 65, no. 8, pp. 2866–2877, Aug. 2017, doi: 10.1109/TMTT.2017.2675418.

[55] P. Fei, Z. Shen, X. Wen, and F. Nian, “A Single-Layer Circular Polarizer Based on Hybrid Meander Line and Loop Configuration,” *IEEE Trans. Antennas Propag.*, vol. 63, no. 10, pp. 4609-4614, Oct. 2015, doi: 10.1109/TAP.2015.2462128.

Zhao Zhou was born in Shaanxi, China. He received the B.Sc. in electronic information engineering and M.Eng. degrees in electromagnetic field and microwave technology from Xidian University, Xi’an, China, in 2017 and 2020, respectively, and the Ph.D. degree in wireless communications from Aalborg University, Aalborg, Denmark, in 2024.

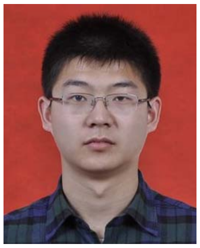
He conducted this research when he was a Postdoctoral Fellow at The Education University of Hong Kong from 2024 to 2025. He is currently a Postdoctoral Fellow at The Hong Kong Polytechnic University. His research interests include machine learning-driven electromagnetic structure design, optimization, and other applications.

Dr. Zhou has served as a reviewer for IEEE Transactions on Antennas and Propogation, IEEE Transactions on Circuits and Systems I: Regular Papers, IEEE Transactions on Artificial Intelligence, IEEE Antennas and Wireless Propogation Letters, and IEEE Microwave and Wireless Technology Letters.

Zhaohui Wei (Member, IEEE) was born in Shandong, China. He received the B.Sc. and M.Eng. degrees in electronic engineering from Xidian University, Xi’an, China, in 2017 and 2020, respectively, and the Ph.D. degree in wireless communications from Aalborg University, Aalborg, Denmark, in 2024.

He joined the School of Electronic Engineering, Xidian University, in 2024, where he is currently an Associate Professor. His research interests include filtering antennas, frequency selective surfaces, and deep learning-based methods for design and analysis of antenna systems.

Dr. Wei has served as a reviewer for IEEE Transactions on Antennas and Propagation, IEEE Transactions on Aritificial Intelligence, and IEEE Antennas and Wireless Propagation Letters.



Jian Ren (Member, IEEE) was born in Shandong, China. He received the B.Sc. and M.Eng. degrees in electronic engineering from Xidian University, Xi’an, China, in 2012 and 2015, respectively, and the Ph.D. degree in electronic engineering from the City University of Hong Kong, Hong Kong, in 2018.

In 2019, he joined the National Key Laboratory of Antennas and Microwave Technology, Xidian University, where he is currently an Associate Professor. Since 2015, he has been a Research Assistant with the City University of Hong Kong. He has authored or co-authored over 40 refereed journal articles. His current research interests include dielectric resonator antenna, millimeter-wave antennas, and metamaterials.

Dr. Ren received the Honorable Mention at the Student Best Paper Competition at the 2018 IEEE 7th Asia–Pacific Conference on Antennas and Propagation (APCAP). He has served as a reviewer for different peer-reviewed journals, including the IEEE Transactions on Antennas and Propagation, the IEEE Antennas and Wireless Propagation Letters, the Journal of Physics D: Applied Physics, and the IET Microwaves, Antennas & Propagation.

Yingzeng Yin (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electromagnetic wave and microwave technology from Xidian University, Xi'an, China, in 1987, 1990, and 2002, respectively.

From 1990 to 1992, he was a Research Assistant and an Instructor with the Institute of Antennas and Electromagnetic Scattering, Xidian University, where he was an Associate Professor with the Department of Electromagnetic Engineering, from 1992 to 1996, and has been a Professor, since 2004. His current research interests include the design of microstrip antennas, feeds for parabolic reflectors, artificial magnetic conductors, phased array antennas, base-station antennas, and computer-aided design for antennas.

Jinna Li (Senior Member, IEEE) received the Ph.D. degree in control theory and control engineering from Northeastern University, Shenyang, China in 2009.

From 2009 to 2011, she was a Post Doc Researcher with the Laboratory of Industrial Control Networks and Systems, Shenyang Institute of Automation, Chinese Academy of Sciences, Beijing, China. From 2014 to 2015, she was a Visiting Scholar with the Energy Research Institute, Nanyang Technological University, Singapore, from 2015 to 2016 with the State Key Laboratory of Synthetical Automation for Process Industries, Northwestern University, and from January to July 2017 with the School of Electrical and Electronic Engineering, the University of Manchester, U.K. She is currently a Full Professor with the School of Information and Control Engineering, Liaoning Petrochemical University, Fushun, China. Her current research interests include neural networks, reinforcement learning, and optimal operational control.

Tse-Tin Chan (Member, IEEE) received the B.Eng. (First Class Hons.) and Ph.D. degrees in information engineering from The Chinese University of Hong Kong (CUHK), Hong Kong SAR, China, in 2014 and 2020, respectively.

He is currently an Assistant Professor with the Department of Mathematics and Information Technology, The Education University of Hong Kong (EdUHK), Hong Kong SAR, China. Previously, he was an Assistant Professor with the Department of Computer Science, The Hang Seng University of Hong Kong (HSUHK), Hong Kong SAR, China, from 2020 to 2022. His research interests include wireless communications and networking, Internet of Things (IoT), age of information (AoI), and AI in wireless communications.