# Improving Generative Inverse Design of Rectangular Patch Antennas with Test-Time Optimization

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

We propose a two-stage deep learning framework for the inverse design of rectangular patch antennas. Our approach leverages generative modeling to learn a latent representation of antenna frequency response curves and conditions a subsequent generative model on these responses to produce feasible antenna geometries. We further demonstrate that leveraging search and optimization techniques at test time improves the accuracy of the generated designs and enables consideration of auxiliary objectives such as manufacturability. Our approach generalizes naturally to different design criteria, and can be easily adapted to more complex geometric design spaces.

## 1 Introduction

In our increasingly wireless world, antennas serve as the fundamental link between radio frequency (RF) waves and electronic devices, and enable essential functions such as communication, navigation, and sensing across diverse systems, including GPS, Wi-Fi, Bluetooth, and cellular networks. Among these, patch antennas (metallic patches printed onto dielectric substrates) are particularly attractive due to their low cost, low profile, and ease of fabrication. Rectangular patch antennas, in particular, are widely utilized in mobile devices and other space-constrained applications [1].

For a patch antenna to be implemented for a specific use case, its behavior needs to be tuned to maximize efficiency. Specifically, the antenna should easily receive and transmit power at certain frequency ranges, while minimizing unwanted radiation or reception at other ranges. These frequency ranges, each consisting of a center frequency and bandwidth, characterize the frequency response of the antenna. Tuning the frequency response of the antenna has the effect of filtering out unwanted signals during reception, and maximizing power converted into radiation at the desired frequency during transmission.

However, designing patch antennas for optimal electromagnetic (EM) performance remains a time-consuming and iterative process. Engineers typically rely on combinations of analytical approximations and full-wave EM simulations to match desired frequency responses [3, 5]. While analytical models or lumped circuit approximations can guide preliminary geometric parameter selection, they are limited to simple geometries and often yield only coarse solutions [1]. Additionally, adjusting one geometric parameter, such as the patch width, can simultaneously affect multiple aspects of the antenna's frequency response, necessitating careful, often tedious manual tuning.

In this work, we propose a novel two-stage generative framework for the inverse design of rectangular patch antennas that leverages variational autoencoders to model distributions over feasible electromagnetic responses and corresponding antenna geometries. The framework's adversarial training process enables controlled generation while its probabilistic nature addresses the one-to-many mapping challenge inherent to inverse design. Through our experiments, we show that simple search and

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

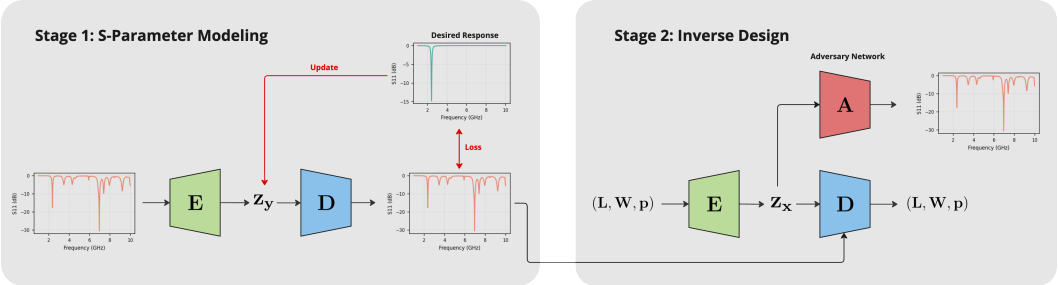



Figure 1: Overview of our two-stage generative inverse design framework.

optimization techniques employed at test time enhance design accuracy and practicability while reducing sensitivity to limited training data. Finally, while demonstrated specifically for rectangular patch antennas, we discuss how our approach naturally generalizes to arbitrary design criteria and more complex geometric design spaces.

# 2 Methodology

## 2.1 Problem Setup and Overview

We consider antennas defined by $x = (L, W, p)$, where $L$ and $W$ are the patch dimensions and $p$ is the feed position. Each design $x$ yields a frequency-dependent reflection coefficient $S_{11}(f)$, sampled at $N = 1000$ frequencies. We represent this response as a 1000-dimensional vector $y \in \mathbb{R}^N$, where we convert the response curve to dB.

We define our design criteria as a set of $K$ desired resonant frequencies $f_{\text{res}} = \{f_1, \ldots, f_K\}$ with their corresponding bandwidths $BW = \{BW_1, \ldots, BW_K\}$ and depths $d = \{d_1, \ldots, d_K\}$. A lower $|S_{11}|_{\text{dB}}$ near a desired frequency band indicates efficient power transfer and radiation, whereas higher $|S_{11}|_{\text{dB}}$ elsewhere can mitigate interference. This means we are searching for antennas that have an idealized target response curve $y^*$, modeled as a product of Lorentzian notches (Equation (1) in Appendix B). Of course, antennas with these exact frequency response curves do not exist in practice (at the very least, there will be higher harmonics). So, we only consider regions of the idealized target response relevant to our design criteria in our search.

Our inverse design goal is as follows: given an idealized target response $y^*(f_{\text{res}}, BW, d)$, find antenna design parameters $\tilde{x}$ such that $\varphi_{\text{EMS}}(\tilde{x}) \simeq y^*$, where $\varphi_{\text{EMS}}$ represents the forward EM simulation [4]. Since $\varphi_{\text{EMS}}$ is expensive to evaluate and the preimage $y^* \mapsto \tilde{x}$ is not unique, we propose a two-stage generative pipeline (see Fig. 1):

1. **Stage 1: Latent Representation of $S_{11}$-Responses.** We train a VAE that encodes $y \in \mathbb{R}^N$ into a latent vector $z_y \in \mathbb{R}^{64}$, capturing the key variations of physically realizable frequency responses. At test time, we search the latent space of this VAE to find an approximation $\tilde{y}$ of $y^*(f_{\text{res}}, BW, d)$.
2. **Stage 2: Conditional Design Generation.** We train a conditional VAE (CVAE) whose encoder maps $x \in \mathbb{R}^3$ to $z_x \in \mathbb{R}^{16}$, and a decoder that maps $z_x$ conditioned on the desired frequency response $\tilde{y}$ to a design $\tilde{x}$. To ensure $\tilde{x}$ depends both on $z_x$ and $\tilde{y}$ (rather than just $z_x$), we use an adversarial predictor that discourages $z_x$ from leaking information about $\tilde{y}$. At test time, we use this CVAE to decode $\tilde{x}$ from $\tilde{y}$ and $z_x$.

## 2.2 Latent Search for Characteristic Target Responses

Using the idealized target response $y^*$ computed from (1) in the CVAE can lead to undesirable results as it is potentially out of the distribution of realizable $y$. Instead, we search the VAE latent space to find latent variables $z_y$ that decode to a physically realizable response $\tilde{y}$. To find good latent space vectors, we perform a gradient-based search over the latent space $z_y$ and perform updates using:

$$z_y^{(t+1)} = z_y^{(t)} - \alpha \nabla_{z_y} \left( \|p_\theta(y|z_y) - y^*\|^2_{\text{masked}} + \lambda_{\text{reg}} \|z_y\|^2 \right),$$

where $\alpha$ is the learning rate. In the loss, we mask $y$ in frequency regions that are irrelevant for our desired response curves, ensuring we only penalize differences in regions of interest. We also add a regularization term $\lambda_{\text{reg}}\|z_y\|^2$ to keep the solution near the prior. This iterative optimization yields a $z_y^*$ whose decoded response $\tilde{y}$ closely approximates $y^*$ and remains on the manifold of physically realizable curves.

### 2.3 Controlled Design Generation

To generate antenna designs $x$ that produce a given frequency response (denoted $y$ using ground truth from the dataset during training, and $\tilde{y}$ from Stage 1 at test time), we train a conditional VAE (CVAE):

$$z_x \sim q_\varphi(z_x|x), \quad x \sim p_\psi(x|z_x, y)\,,$$

where $q_\varphi$ and $p_\psi$ are feed-forward networks mapping between design parameters and latent variables.

However, during training, $z_x$ may easily learn to encode all information about $x$ needed for reconstruction. As noted by [2], this can lead to a pathological scenario where the decoder simply ignores the conditional input $y$, since the necessary information is already present in the latent code $z_x$. In such cases, attempts to control generation by varying $y$ would have no effect. To prevent this, we follow [2] and introduce an adversarial predictor $D_\omega$ that attempts to predict $y$ directly from $z_x$. We let $\eta$ be a hyperparameter controlling the weight of the adversarial term. The predictor tries to minimize

$$\mathcal{L}_{\text{pred}} = \mathbb{E}_{q_\varphi(z_x|x)}[\|y - D_\omega(z_x)\|^2]\,.$$

The encoder tries to maximize this error (i.e., make it hard to predict $y$ from $z_x$). By this minimax interplay, the encoder removes direct correlation between $z_x$ and $y$. Since $x$ maps to a unique $y$ via the EM solver, forcing the encoder to remove $y$-information from $z_x$ ensures the decoder must rely on the explicit conditional input $y$ to reconstruct the design. This yields a controllable model: changes in $y$ at test time directly influence $x$, and $z_x$ can be adjusted to handle auxiliary objectives without altering the frequency response of $x$.

### 2.4 Test-Time Optimization

While our two-stage framework generates feasible designs directly from target specifications, additional refinement at test time can further improve accuracy and practicability. We consider two approaches: (1) generating multiple candidate designs and (2) optimizing single candidates to satisfy auxiliary constraints.

**Generating Multiple Candidates.** By sampling multiple response curves $\tilde{y}$ during latent search and multiple designs from the conditional VAE for each candidate, we obtain a pool of potential solutions.

**Optimizing a Single Candidate.** For a single candidate curve, the latent code of the conditional design decoder can be further optimized to meet auxiliary geometric criteria. For the rectangular patch antenna, we define a penalty

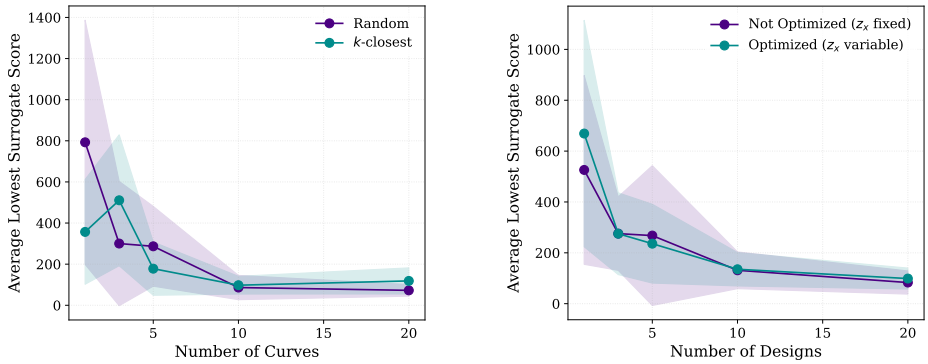
$$\mathcal{L}_{\text{penalty}}(L, W, p) = \text{ReLU}(-L)^2 + \text{ReLU}(-W)^2 + \text{ReLU}\left(-\frac{L}{2} - p\right)^2 + \text{ReLU}(p)^2\,,$$

encouraging positive dimensions and a feed position between the center and edge of the patch. We then use this penalty to conduct a gradient-based search over the latent variable $z_x$ in the same way as described in Section 2.2 for $z_y$.

## 3 Experiments

### 3.1 Test-Time Compute Scaling

We conduct two investigations to determine how increasing test-time compute affects framework performance. In both investigations, we consider three distinct target response functions, and at each search configuration, record the lowest *Surrogate Scorer* score from the pool of generated antenna designs, averaged across the three targets. Results can be seen in Figure 2.




(a) Performance vs. No. of Curves

(b) Performance vs. No. of Designs per Curve

Figure 2: Scaling performance as the number of curves (left) and the number of designs per curve (right) is increased. The shaded regions indicate variability across runs.

**Multiple Candidate Curves.** To determine how sampling multiple candidate frequency response curves $\tilde{y}$ affects design accuracy, we consider a setup where one antenna design $\tilde{x}$ is sampled per candidate curve, and vary the number of candidate curves. To obtain the different curves $\tilde{y}$, we explore two initialization strategies for $z_y^{(0)}$: (i) *Random*, initializes each $z_y$ as a sample from a Gaussian prior; (ii) $k$-*closest*, initializes $z_y$ as the latent codes of the $k$ curves in the training dataset closest to the ideal curve ($k$ is the number of candidate curves).

We find that as the pool of antenna designs $\tilde{x}$ grows with the number of candidate frequency response curves, both initialization strategies yield better and more consistent predictions. Additionally, we find that early on, $k$-closest initialization may offer more stable or slightly better performance than random initialization, but as the number of curves increases, the difference in their average performance diminishes. This makes sense, since $k$-closest initialization potentially offers a more principled starting point for optimization in the one-shot case.

**Multiple Candidate Designs per Curve.** To determine how the number of generated designs per candidate curve affects design accuracy, we consider a setup where only one candidate response curve $\tilde{y}$ is sampled, and vary the number of designs sampled from this curve. Additionally, we explore the case where $z_x$ is optimized according to the auxiliary geometric criteria and compare it to the unoptimized performance.

Again, we find that the quality of the antennas improves as we sample multiple antenna designs $\tilde{x}$. Additionally, we find that the quality of the optimized and unoptimized design agrees well across search configurations, indicating that we have been successful in decorrelating $z_x$ from the physical response of $x$.

## 4 Conclusion

In this work, we presented a two-stage generative framework for the inverse design of rectangular patch antennas that effectively addresses key challenges in computational antenna design. Our approach combines the strengths of generative modeling with targeted test-time optimization to yield physically realizable antenna designs that meet desired frequency response characteristics.

Our experimental results demonstrate that test-time computation can dramatically improve design quality with minimal additional training data or model complexity.

Our approach generalizes naturally to more complex electromagnetic design tasks. While we focused on rectangular patch antennas with three design parameters, the same framework could be extended to patch antennas with arbitrary geometries or to multi-element antenna arrays with more complex frequency-domain behavior. Furthermore, our test-time optimization framework offers a flexible mechanism for incorporating auxiliary design constraints, such as fabrication limitations or size restrictions, without compromising the primary electromagnetic performance objectives.

# References


[1] Constantine A Balanis. *Antenna theory*. Wiley-Blackwell, Hoboken, NJ, 4 edition, January 2016.

[2] Jesse Engel, Kumar Krishna Agrawal, Shuo Chen, Ishaan Gulrajani, and Adam Roberts. Gansynth: Adversarial neural audio synthesis. In *ICLR*, 2019.

[3] Jianming Jin. *The Finite Element Method in Electromagnetics*. Wiley - IEEE. John Wiley & Sons, Nashville, TN, 3 edition, March 2014.

[4] Thorsten Liebig. openems - open electromagnetic field solver, accessed 2024. URL `https://www.openEMS.de`.

[5] Allen Taflove and Susan Hagness. *Computational electrodynamics*. Artech House antennas and propagation library. Artech House, Norwood, MA, 3 edition, May 2005.


# A Code and Data Availability

We have made our dataset, simulation scripts, trained models, and source code to reproduce all experiments and figures publicly available at `https://anonymous.4open.science/r/patch-antenna-tto-FBED`.

# B Idealized Target Response Model

We represent idealized target responses as a product of Lorentzian notches

$$S_{11}^{*}(f \mid f_{\text{res}}, BW, d) = \prod_{k=1}^{K} \left[ 1 - (1 - 10^{\frac{d_k}{20}}) \left( 1 - \left( 1 - \frac{(BW_k/2)^2}{(f - f_k)^2 + (BW_k/2)^2} \right) \right) \right] , \quad (1)$$

where $f_k$ denotes the resonant frequency where minimal reflection is desired, $BW_k$ specifies the width of the frequency band around $f_k$ with efficient power transfer, and $d_k$ sets the target depth (in dB) of the reflection at $f_k$.

From this response curve, we convert to idealized target dB values

$$y^{*}(f_i \mid f_{\text{res}}, BW, d) = 20 \log_{10} |S_{11}^{*}(f_i \mid f_{\text{res}}, BW, d)|, \quad i = 1, \ldots N . \quad (2)$$

# C Comparison of Target vs. Simulated Responses

Table 1: Patch antenna geometries corresponding to the frequency responses in Figure 3.

| Target $f_r$ | Budget (# curves $\times$ # designs) | $L$ [mm] | $W$ [mm] | $p$ [mm] |
|---|---|---|---|---|
| 2.40 GHz | $1 \times 1$ | 30.0 | 40.5 | $-7.4$ |
| | $10 \times 20$ | 31.3 | 45.8 | $-7.2$ |
| 4.00 GHz | $1 \times 1$ | 36.0 | 53.6 | $-2.6$ |
| | $10 \times 20$ | 37.8 | 52.7 | $-2.7$ |
| 5.00 GHz | $1 \times 1$ | 15.7 | 25.5 | $-2.1$ |
| | $10 \times 20$ | 29.0 | 46.9 | $-1.7$ |

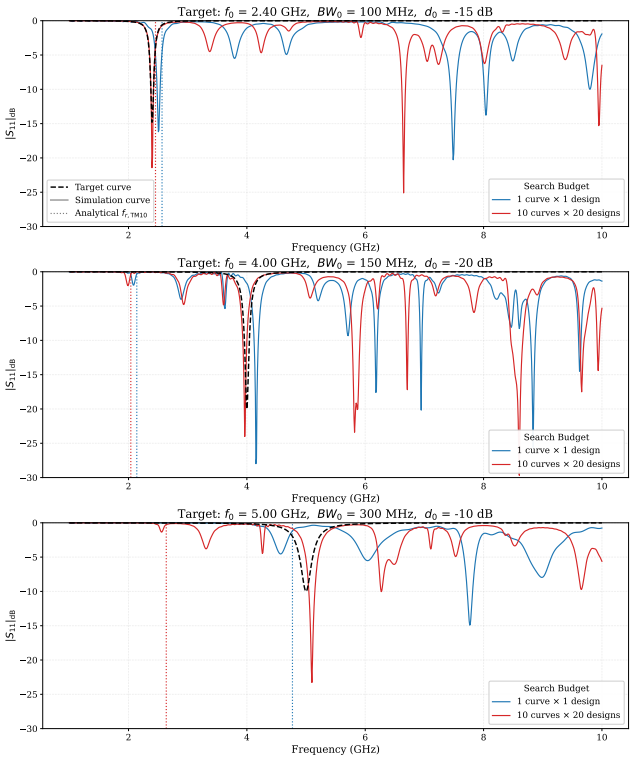



Figure 3: Comparison of the idealized target $S_{11}$ curve $y^*$ (black dashed), the dominant-mode analytical resonance $f_{r,\mathrm{TM10}}$ (dotted vertical), and the simulated $S_{11}$ of designs $\tilde{x}$ generated with two test-time compute budgets. Blue = 1 curve $\times$ 1 design, red = 10 curves $\times$ 20 designs. *Note:* Generated geometries may exploit higher-order or coupled modes, so exact agreement with the analytical reference is not expected.