

Invertible Neural Networks for Design of Broadband Active Mixers

Oluwaseyi Akinwande

3D Systems Packaging Research Center
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia, USA
oakinwande3@gatech.edu

Xingchen Li

3D Systems Packaging Research Center
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia, USA
xingchen.li@gatech.edu

Osama Waqar Bhatti

3D Systems Packaging Research Center
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia, USA
osamawaqarbhatti@gatech.edu

Madhavan Swaminathan

3D Systems Packaging Research Center
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia, USA
madhavan.swaminathan@ece.gatech.edu

ABSTRACT

In this work, we present the invertible neural network for predicting the posterior distributions of the design space of broadband active mixers with RF from 100 MHz to 10 GHz. This invertible method gives a fast and accurate model when investigating crucial properties of active mixers such as conversion gain and noise figure. Our results show that the response generated by the invertible neural network model has close correlation with the output response from the circuit simulator.

CCS CONCEPTS

• **Hardware** → **Electronic design automation**; *Modeling and parameter extraction.*

KEYWORDS

invertible neural networks, neural networks, inverse design, broadband, RF front end, active mixers, design space exploration

ACM Reference Format:

Oluwaseyi Akinwande, Osama Waqar Bhatti, Xingchen Li, and Madhavan Swaminathan. 2022. Invertible Neural Networks for Design of Broadband Active Mixers. In *Proceedings of the 2022 ACM/IEEE Workshop on Machine Learning for CAD (MLCAD '22)*, September 12–13, 2022, Snowbird, UT, USA. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3551901.3556491>

1 INTRODUCTION

In electronic design automation (EDA), circuit and system designers often iterate through copious amounts of design variables, a process termed design space exploration, in an attempt to find the optimal solution that satisfies a target performance. This process is

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MLCAD '22, September 12–13, 2022, Snowbird, UT, USA.

© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9486-4/22/09...\$15.00
<https://doi.org/10.1145/3551901.3556491>

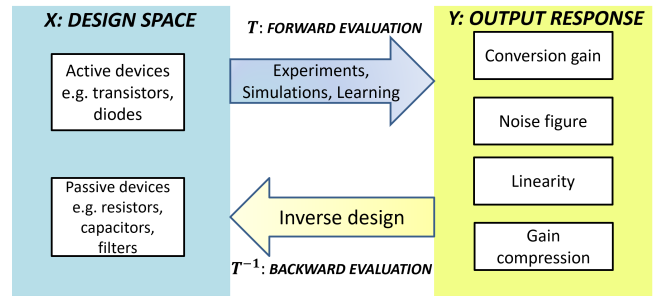


Figure 1: Model-based design framework that offers a custom solution.

usually time-consuming and cost-intensive because a lot of evaluations have to be carried out. To tackle this task more effectively, optimization methods [16], and surrogate modeling [18], [19] have been proposed. However, the best solution may still not be achieved, or several possibilities may be ignored [6].

Recently, machine learning (ML) frameworks have taken strides in learning the forward and inverse mappings between a set of inputs and outputs [14]. Consider a design space X of a parameterized system, as succinctly shown in Fig. 1, that forms the input of the ML framework or surrogate model, with corresponding output response Y . This mapping relationship can be represented as:

$$Y = T(X) + \epsilon, \quad (1)$$

where $T(\cdot)$ is the forward mapping and ϵ models the system noise. The forward model learns the input-output relationship and predicts the output response given the input parameters. To estimate the best set of input parameters that satisfies the desired target, we find the inverse mapping:

$$\hat{X} = T^{-1}(Y), \quad (2)$$

where $\hat{X}$ is the inverse solution. This process is called inverse design. Inverse problems are often inherently ill-posed and intractable for the following reasons: (1) Existence, which asks if the inverse even exists and (2) Uniqueness, due to the ambiguity brought by

the one-to-many mapping in the inverse direction. Several architectures have been proposed to address the problem of invertibility. Artificial neural network models have been proposed to solve the inverse problem by either: (1) Evaluating models iteratively to find the optimal solutions for the specified output response, or (2) Training the input and output nodes by transposing them [10]. These methods produce deterministic and point-estimate solutions with no reliability guarantees. State-of-the-art generative models like the generative adversarial network (GAN) [8], variational auto-encoder (VAE) [11], and invertible neural network (INN) [5], [2] address this issue by generating conditional posterior distributions rather than point-estimates. One major advantage that the INN enjoys is that the forward and inverse mappings can be efficiently computed [2].

In this paper, we propose an inverse design method for microwave circuits and electronic systems using INN. With inverse design, the design parameters can be directly determined from the output objectives. This offers the benefit of reduced design-cycle time and related costs by increasing the overall efficiency of the design process. We demonstrate the validity of the proposed method with the inverse design of broadband active mixers with output dimensionality up to 200. Active mixers is one of the various sensitive circuitry in the make-up of the RF front end, hence, accurate nonlinear modeling of mixers is crucial to getting good performance.

The rest of this paper is organized as follows: section II delves into the invertible neural network framework; sections III and IV explore different applications of the INN method to electronic design, and present the results; we provide a concluding remark in section V.

2 INVERTIBLE NEURAL NETWORK (INN)

The INN is a flow-based generative model [5]. It is based on the concept of normalizing flows which utilizes the change-of-variable law of probabilities to compute posterior probability distributions. The INN possesses the properties of bijectivity to allow for efficient computation of forward and inverse mappings. It guarantees the existence of the inverse solution. Consider a sample x from design space X and its probability density $p_X(x)$, with its corresponding y from the response space Y and its unknown probability density $p_Y(y)$ with a transformation $Y = f(X)$, we can form a relationship between their probability densities through the change-of-variables method [5], [9]:

$$p_Y(y|\theta) = p_X(f_\theta^{-1}(y)) \cdot \left| \det \left(\frac{df_\theta^{-1}}{dx} \right) \right|, \quad (3)$$

where we define all the composition of the INN architecture in a single function f_θ , and θ is the set of all network parameters. The INN comprises stacks of invertible blocks, known as affine coupling blocks, that can be trained simultaneously in either direction, as shown in Fig. 2. Latent variables z (usually Gaussian) use the latent space to learn the nonlinear transformation between this known distribution and the original data distribution [20]. In Fig. 3, the input vector to the affine coupling block is halved into $[x_1, x_2]$, and they are transformed by an affine function with coefficients e^s and t , given by [5], [2]:

$$y_1 = x_1, \quad y_2 = x_2 \circ e^{s(x_1)} + t(x_1). \quad (4)$$

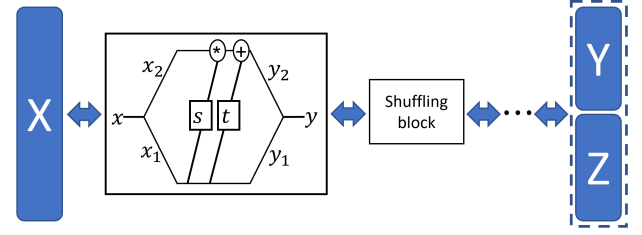


Figure 2: Architecture of the Invertible Neural Network (X: input, Y: output, Z: latent variable) [4].

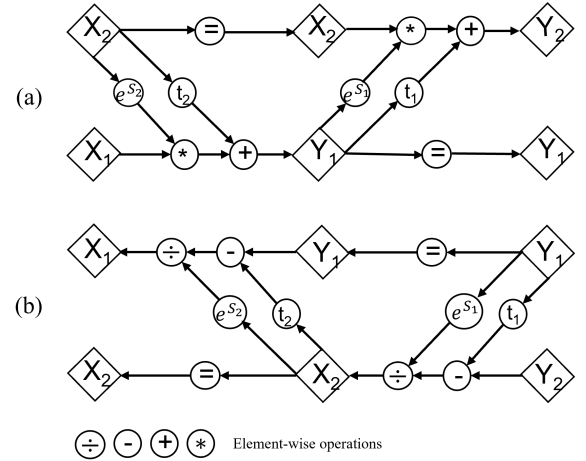


Figure 3: Computational graphs for (a) forward and, (b) inverse propagation [5].

Given the block's output $[y_1, y_2]$, these expressions are invertible through [5], [2]:

$$x_1 = y_1, \quad x_2 = (y_2 - t(y_1)) \circ e^{-s(y_1)}. \quad (5)$$

(4) represents the forward mapping while (5) represents the inverse mapping (see Fig. 3 for a graphical illustration). The transformations in (4) and (5) use element-wise additive (+) and multiplicative ($\circ$) operations which allows the inverse to be computed without requiring the scale $s(\cdot)$ and translation $t(\cdot)$ networks to be inverted. The bijectivity of the INN model allows for bi-directional operation and training, and therefore both forward and inverse processes can be well learned [20], [1]. We present the pseudocode for the INN training in Algorithm 1 which is adapted from the source code [3].

3 APPLICATION I: INVERSE DESIGN OF LOW-POWER SINGLE TRANSISTOR ACTIVE MIXER

To demonstrate the effectiveness of the proposed method, we apply the INN method for the design of a low-power active mixer, which is a nonlinear device used for summing and subtracting frequencies. They are characterized by their conversion gain or loss and how much noise they introduce in the circuit. Consequently, accurate nonlinear modeling of mixers is crucial to getting good performance.

Algorithm 1: INN training

Input: training data: $\{X, Y\}$, #epochs, learning rate: α ,
 $p(z) = N(0, I_{D_z})$
Output: training losses: $\mathcal{L}$, trained model
while $i \leq \text{\#epochs}$ **do**
 for $x_{\text{batch}}, y_{\text{batch}} \in \{X, Y\}$, **do**
 $[y_{\text{pred}}, z_{\text{pred}}] = f_{\theta}(x_{\text{batch}})$
 $\mathcal{L}_y = \text{MSE}(y_{\text{pred}}, y_{\text{batch}})$
 /* MSE: mean squared error */
 $\mathcal{L}_z = \text{MMD}(q(y, z), p(y)p(z))$
 /* MMD: maximum mean discrepancy [2] */
 sample $z \sim p(z)$
 $x_{\text{pred}} = f_{\theta}^{-1}([y_{\text{batch}}, z])$
 $\mathcal{L}_x = \text{MMD}(g(x), p(x))$
 $\mathcal{L}_{\text{total}} = w_x \mathcal{L}_x + w_y \mathcal{L}_y + w_z \mathcal{L}_z$
 $p \leftarrow p - \alpha \nabla(\mathcal{L}_{\text{total}})$

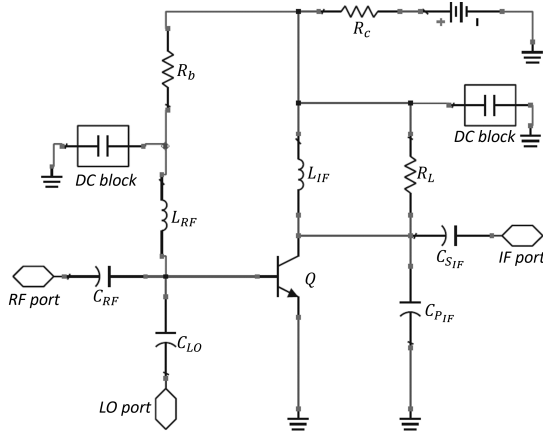


Figure 4: Low-power single transistor active mixer schematic [15].

The mixer is a down converter, with an RF of 0.1 – 10 GHz and 855 MHz LO, operating from a 1 V DC supply at 600 μ A current. This very low power consumption is typical of applications such as pagers and cellular phones, where battery lifetime is critical [15]. The objective here is to (1) obtain the mixer design parameters that correspond to a given specification of gain and noise figure, and (2) validate the design parameters through a forward evaluation.

3.1 Model Setup

The design parameters of the mixer are shown in Fig. 4, and their range of values are given in Table 1. Note that the parameters are constrained within the limits of the operating region of the transistor, and that the only 3 parameters that are varied for this example are R_L , C_{LO} and C_{RF} . The target characteristics investigated are the conversion gain G and the noise figure NF , given as [12], [13], [15]:

$$G \text{ (dB)} = P_{\text{IF}} - P_{\text{RF}} \quad (6)$$

Table 1: Design Space Parameters of Low-power Single Transistor Mixer

Parameter		Unit	Min	Max	Step
Load resistor	R_L	k Ω	1	5	0.5
LO capacitor	C_{LO}	pF	0.1	0.9	0.2
RF capacitor	C_{RF}	pF	1	2	0.2
Collector resistor	R_c	k Ω		0.47	
Base resistor	R_b	k Ω		8.2	
RF inductor	L_{RF}	nH		6.6	
IF inductor	L_{IF}	nH		270	
Parallel IF capacitor	C_{PIF}	pF		33	
Series IF capacitor	C_{SIF}	pF		13.3	
Motorola MMBR941	Q				

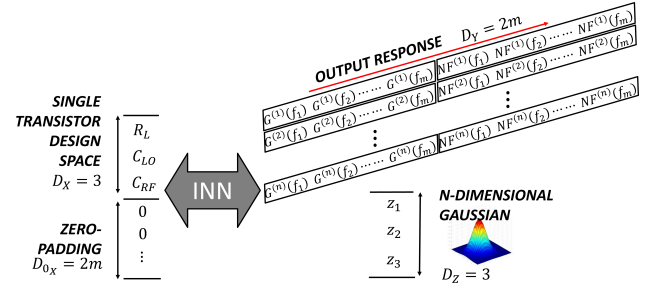


Figure 5: Proposed Invertible Neural Network (INN) model setup for low-power single transistor active mixer design. $m = 100$ frequency points.

and

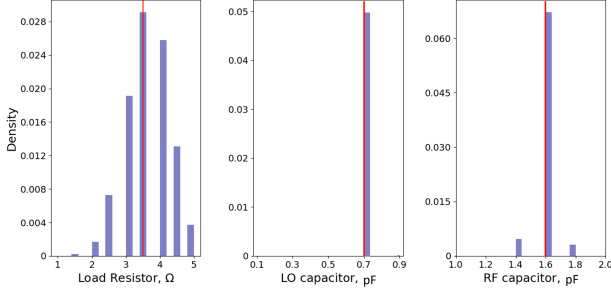
$$NF = \frac{N_i G + N_0(\text{mixer})}{N_i G}, \quad (7)$$

where P_{IF} and P_{RF} are the powers at the IF and RF ports respectively, $N_i G$ and $N_0(\text{mixer})$ are the input noise and noise added by the mixer (both referred to the IF port), respectively. The RF is swept from 0.1 – 10 GHz with steps of 100 MHz. Consequently, each tuple in the design space has the corresponding gain and noise figure with 100 frequency points. We generate 270 samples in a uniform fashion using Keysight ADS [15]. The data is divided into train and test sets.

The objective here is to determine an invertible mapping between the design space X and output response Y . The proposed model setup is shown in Fig. 5. The INN model is constructed using 8 reversible blocks with shuffling layers between them. Each reversible block contains the scale $s(\cdot)$ and shift $t(\cdot)$ networks which are constructed with fully connected neural networks with one hidden layer of 15 neurons and Rectified Linear Unit (ReLU) activation functions. On the input side of the model setup, there are 3 mixer design parameters, zero-padded to $2m$ dimensions, where $m = 100$ frequency points. The output variables consist of mixer design specifications (gain and noise figure) and 3-dimensional latent variables z sampled from a standard normal distribution, with no zero-padding. The INN model is trained for 300 epochs with 10 iterations per epoch.

Table 2: Performance of Inverse Design Tuples for Low-power Single Transistor Mixer Design

Inverse Design Tuple	Normalized Mean Squared Error (NMSE)	
	Gain, G %	Noise Figure, NF %
1 {3.5 k Ω , 0.7 pF, 1.6 pF}	0.001	0.083
2 {3.5 k Ω , 0.1 pF, 1.8 pF}	0.002	0.013

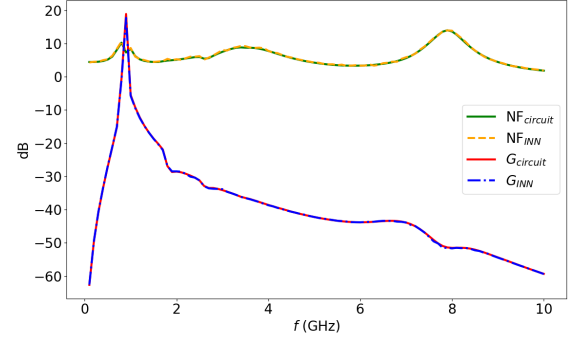
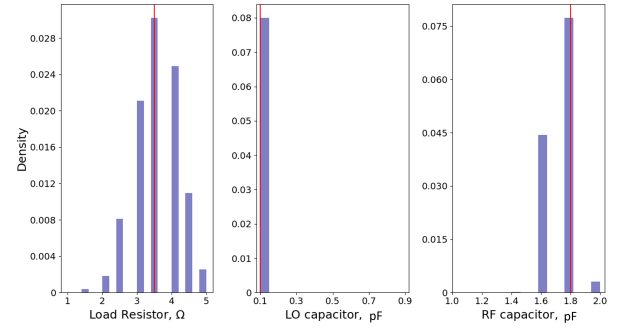
**Figure 6: Predicted conditional posterior distributions $p(x|y_{\text{target}})$ of low-power single transistor active mixer design parameters. Red vertical lines indicate the points with the highest densities. When the points corresponding to the highest densities are sampled for the design parameters, the tuple obtained is {3.5 k Ω , 0.7 pF, 1.6 pF}.**

3.2 Results

We perform 2 inferences with the trained INN model to validate the method: inverse and forward evaluations. During the inference process, we choose a random response y_{target} from the test set and we obtain the inverse solution. The INN model generates rich conditional posterior distributions of the mixer design parameters as shown in Fig. 6. Next, we obtain an inverse tuple from these distributions by sampling the points with the highest densities in the mixer design space. The tuple obtained is $\hat{x} = \{3.5 \text{ k}\Omega, 0.7 \text{ pF}, 1.6 \text{ pF}\}$. We take this tuple and perform a forward evaluation with the INN model to obtain the corresponding gain and noise figure. These are compared with the actual responses in Fig. 7. We observe that the responses from the INN model almost completely overlap the responses from the Keysight ADS circuit simulator. We repeat the inverse-forward inference above for another response y_{target} chosen randomly from the test set and present the results in Fig. 8 and 9. Similarly, the responses from the INN model almost completely overlap the responses from the Keysight ADS circuit simulator. We provide a summary of our results in Table 2.

4 APPLICATION II: INVERSE DESIGN OF GILBERT CELL MIXER

We further validate the INN method with another mixer design. Consider the IAM-81018 Gilbert cell mixer [7], [17], [15], shown in Fig. 10. The mixer is a down converter, with an RF of 0.1 – 10 GHz and 1.75 GHz LO, operating from a 5V DC supply. The double balanced structure has the advantage of better isolation between all ports and increased linearity. The objective here is two-fold: (1)

**Figure 7: Forward evaluation showing gain G and noise figure NF for low-power single transistor active mixer, with the trained INN model and Keysight ADS circuit simulator for the INN-generated design tuple {3.5 k Ω , 0.7 pF, 1.6 pF}. The responses from the trained INN model and the circuit simulator almost completely overlap.****Figure 8: Predicted conditional posterior distributions $p(x|y_{\text{target}})$ of low-power single transistor active mixer design parameters. Red vertical lines indicate the points with the highest densities. When the points corresponding to the highest densities are sampled for the design parameters, the tuple obtained is {3.5 k Ω , 0.1 pF, 1.8 pF}.**

to obtain the mixer design parameters that correspond to a given specification of gain and noise figure, and (2) to validate the design parameters through a forward evaluation.

Table 3: Design Parameters of Gilbert Cell Mixer

Parameter		Unit	Min	Max	Step
Collector resistor	R_c	Ω	50	450	50
Tail resistor	R_t	Ω	20	220	50
Base resistor (top)	R_{b1}	Ω	10	160	30
Base resistor (bottom)	R_{b2}	Ω	10	90	20
Coupling capacitor	C_c	μF	0.4	1.2	0.2
Bypass capacitor	C_b	μF	0.4	1.2	0.2

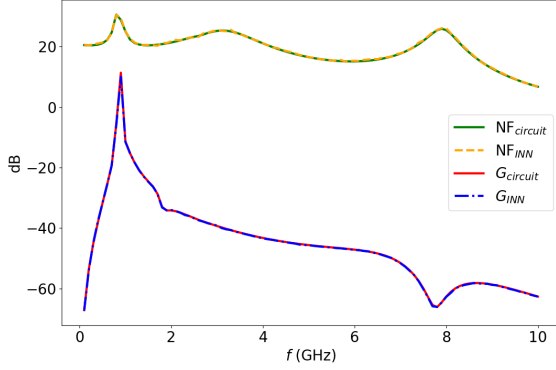


Figure 9: Forward evaluation showing gain G and noise figure NF for low-power single transistor active mixer, with the trained INN model and Keysight ADS circuit simulator for the INN-generated design tuple $\{3.5 \text{ k}\Omega, 0.1 \text{ pF}, 1.8 \text{ pF}\}$. The responses from the trained INN model and the circuit simulator almost completely overlap.

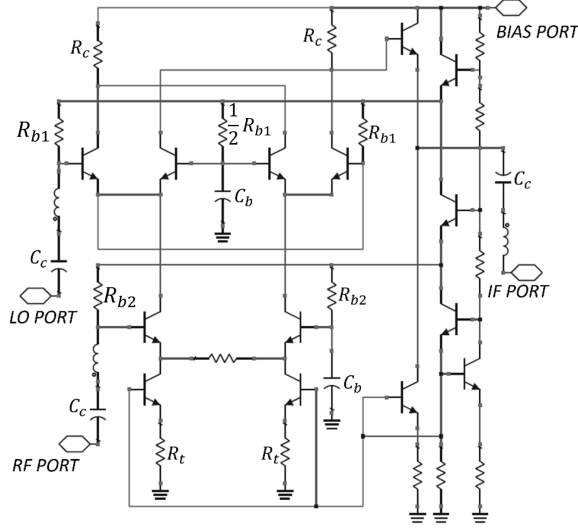


Figure 10: IAM-81018 Gilbert cell mixer schematic [7], [17], [15].

4.1 Model Setup

The design parameters of the mixer are the passive components as shown in Fig. 10, and their range of values are given in Table 3. Note that the parameters are constrained within the limits of the operating region of the transistor. The target characteristics investigated are the conversion gain G and the noise figure NF , as described in (6) and (7), respectively. We perform a parametric sweep of the mixer design space, with the RF being swept from 0.1 – 10 GHz with steps of 100 MHz for each combination of the design parameters. Consequently, each tuple in the design space has the corresponding gain and noise figure with 100 frequency

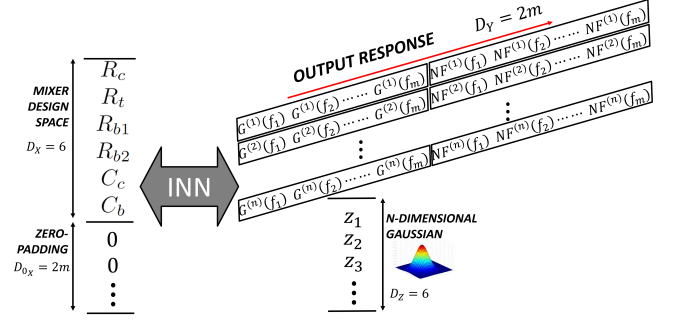


Figure 11: Proposed Invertible Neural Network (INN) model setup for Gilbert cell mixer design. $m = 100$ frequency points.

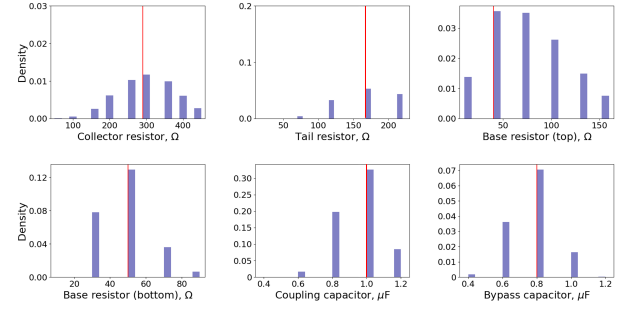


Figure 12: Predicted conditional posterior distributions $p(x|y_{\text{target}})$ of Gilbert cell mixer design parameters. Red vertical lines indicate the points with the highest densities. When the points corresponding to the highest densities are sampled for the design parameters, the tuple obtained is $\{300 \Omega, 170 \Omega, 40 \Omega, 50 \Omega, 1 \mu\text{F}, 0.8 \mu\text{F}\}$.

points. We generate 33750 samples using Keysight ADS [15], and the dataset is divided into train and test sets.

The objective here is to determine an invertible mapping between the design space X and output response Y . The proposed model setup is shown in Fig. 11. The INN model is constructed using 8 reversible blocks with shuffling layers between them. Each reversible block contains the scale $s(\cdot)$ and shift $t(\cdot)$ networks which are constructed with fully connected neural networks with one hidden layer of 15 neurons, 70% dropout regularization, batch normalization layers and Leaky Rectified Linear Unit (LeakyReLU) activation functions. On the input side of the model setup, there are 6 mixer design parameters, zero-padded to $2m$ dimensions, where $m = 100$ frequency points. The output variables consist of mixer design specifications (gain and noise figure) and 6-dimensional latent variables z sampled from a standard normal distribution, with no zero-padding. The INN model is trained for 300 epochs with 10 iterations per epoch.

4.2 Results

We perform a two-step inference process with the trained INN model: the inverse evaluation to obtain the inverse solution, then, the forward evaluation to validate the method. First, we choose a

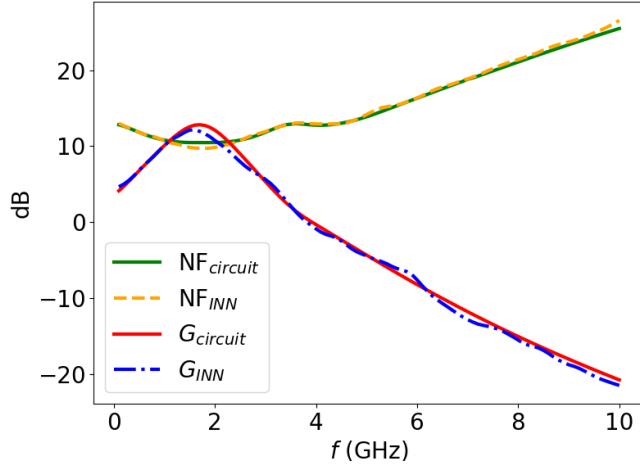


Figure 13: Forward evaluation showing gain G and noise figure NF for the Gilbert cell mixer, with the trained INN model and Keysight ADS circuit simulator for the INN-generated design tuple $\{300 \Omega, 170 \Omega, 40 \Omega, 50 \Omega, 1 \mu F, 0.8 \mu F\}$. The responses from the trained INN model and the circuit simulator have close correlation.

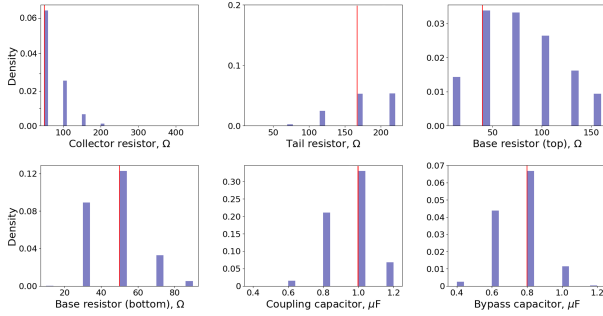


Figure 14: Predicted conditional posterior distributions $p(x|y_{\text{target}})$ of Gilbert cell mixer design parameters. Red vertical lines indicate the points with the highest densities. When the points corresponding to the highest densities are sampled for the design parameters, the tuple obtained is $\{50 \Omega, 170 \Omega, 40 \Omega, 50 \Omega, 1 \mu F, 0.8 \mu F\}$.

random response y_{target} from the test set and we obtain the inverse solution with the INN model, which generates rich conditional posterior distributions of the mixer design parameters as shown in Fig. 12. Next, we obtain an inverse tuple from these distributions by sampling the points with the highest densities in the mixer design space. The tuple obtained is $\hat{x} = \{300 \Omega, 170 \Omega, 40 \Omega, 50 \Omega, 1 \mu F, 0.8 \mu F\}$. We take this tuple and perform a forward evaluation with the INN model to obtain the corresponding gain and noise figure. These are compared with the actual responses in Fig. 13. We observe that the responses from the INN model have close correlation with the responses from the Keysight ADS circuit simulator. We repeat the inverse-forward inference above for another response y_{target} chosen randomly from the test set and present the results in Fig. 14 and

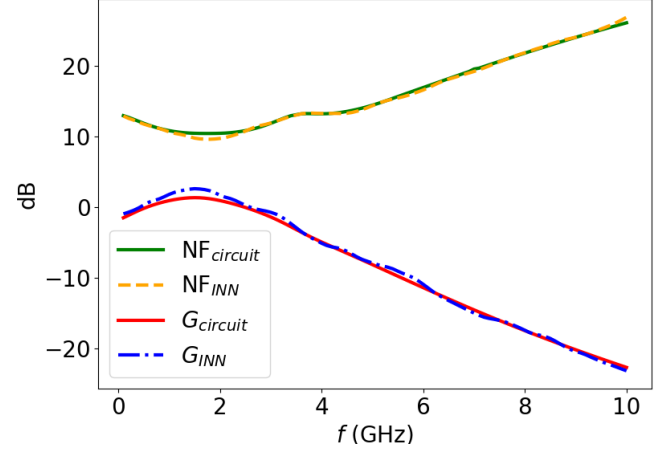


Figure 15: Forward evaluation showing gain G and noise figure NF for the Gilbert cell mixer, with the trained INN model and Keysight ADS circuit simulator for the INN-generated design tuple $\{50 \Omega, 170 \Omega, 40 \Omega, 50 \Omega, 1 \mu F, 0.8 \mu F\}$. The responses from the trained INN model and the circuit simulator have close correlation.

Table 4: Performance of Inverse Design Tuples for Gilbert Mixer Design

	Inverse Design Tuple	Normalized Mean Squared Error (NMSE)	
		Gain, G	Noise Figure, NF
		%	%
1	$\{300 \Omega, 170 \Omega, 40 \Omega, 50 \Omega, 1 \mu F, 0.8 \mu F\}$	1.427	0.325
2	$\{50 \Omega, 170 \Omega, 40 \Omega, 50 \Omega, 1 \mu F, 0.8 \mu F\}$	0.801	0.134

15. Similarly, the responses from the INN model have close correlation with the responses from the Keysight ADS circuit simulator. We provide a summary of our results in Table 4.

5 CONCLUSION

We present an inverse design method for microwave circuits and electronic systems using invertible neural network. With inverse design, the design parameters can be directly determined from the output objectives. This offers the benefit of reduced design-cycle time and related costs by increasing the overall efficiency of the design process. We demonstrate the validity of the proposed method with the inverse design of broadband active mixers where we obtain inverse posterior distributions. For the forward evaluation, our results show that the responses generated by the invertible neural network models have close correlation with the output responses from the circuit simulator.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. CNS 16-2137259 - Center for Advanced Electronics through Machine Learning (CAEML).

REFERENCES

- [1] Nikita Ambasana, Osama W. Bhatti, Majid A. Dolatsara, Madhavan Swaminathan, Xianbo Yang, Pavel R. Paladhi, and Wiren Dale Becker. 2021. Invertible Neural

- Networks for High-Speed Channel Design & Parameter Distribution Estimation. In *2021 IEEE 30th Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS)*. 1–3. <https://doi.org/10.1109/EPEPS51341.2021.9609225>
- [2] Lynton Ardizzone, Jakob Kruse, Sebastian J. Wirkert, Daniel Rahner, Eric Pellegrini, Ralf S. Klessen, Lena Maier-Hein, Carsten Rother, and U. Köthe. 2019. Analyzing Inverse Problems with Invertible Neural Networks. *ArXiv abs/1808.04730* (2019).
 - [3] Lynton Ardizzone, Jakob Kruse, Sebastian J. Wirkert, Daniel Rahner, Eric Pellegrini, Ralf S. Klessen, Lena Maier-Hein, Carsten Rother, and U. Köthe. 2021. *Framework for Easily Invertible Architectures (FrELA)*. Retrieved June, 2022 from <https://github.com/VLL-HD/FrELA>
 - [4] Osama Waqar Bhatti, Nikita Ambasana, and Madhavan Swaminathan. 2021. Inverse Design of Power Delivery Networks using Invertible Neural Networks. In *2021 IEEE 30th Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS)*. 1–3. <https://doi.org/10.1109/EPEPS51341.2021.9609211>
 - [5] Laurent Dinh, Jascha Narain Sohl-Dickstein, and Samy Bengio. 2017. Density estimation using Real NVP. *ArXiv abs/1605.08803* (2017).
 - [6] Majid Ahadi Dolatsara, Huan Yu, Jose Ale Hejase, Wiren Dale Becker, and Madhavan Swaminathan. 2020. Invertible Neural Networks for Inverse Design of CTLE in High-speed Channels. In *2020 IEEE Electrical Design of Advanced Packaging and Systems (EDAPS)*. 1–3. <https://doi.org/10.1109/EDAPS50281.2020.9312919>
 - [7] B. Gilbert. 1968. A precise four-quadrant multiplier with subnanosecond response. *IEEE Journal of Solid-State Circuits* 3, 4 (1968), 365–373. <https://doi.org/10.1109/JSSC.1968.1049925>
 - [8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*. 2672–2680.
 - [9] John A. Gubner. 2006. *Probability and Random Processes for Electrical and Computer Engineers*. Cambridge University Press, USA.
 - [10] Humayun Kabir, Ying Wang, Ming Yu, and Qi-Jun Zhang. 2008. Neural Network Inverse Modeling and Applications to Microwave Filter Design. *IEEE Transactions on Microwave Theory and Techniques* 56, 4 (2008), 867–879. <https://doi.org/10.1109/TMTT.2008.919078>
 - [11] Diederik P. Kingma and Max Welling. 2014. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14–16, 2014, Conference Track Proceedings*. arXiv:<http://arxiv.org/abs/1312.6114v10> [stat.ML]
 - [12] David M. Pozar. 2012. *Microwave Engineering*. Wiley, Hoboken, NJ, USA.
 - [13] Behzad Razavi. 2011. *RF Microelectronics (Prentice Hall Communications Engineering and Emerging Technologies Series)* (2nd ed.). Prentice Hall Press, USA.
 - [14] Madhavan Swaminathan, Hakki Mert Torun, Huan Yu, Jose Ale Hejase, and Wiren Dale Becker. 2020. Demystifying Machine Learning for Signal and Power Integrity Problems in Packaging. *IEEE Transactions on Components, Packaging and Manufacturing Technology* 10, 8 (2020), 1276–1295. <https://doi.org/10.1109/TCPMT.2020.3011910>
 - [15] Advanced Design System. 2022. *Keysight ADS*. Retrieved June 19, 2022 from <https://www.keysight.com>
 - [16] Hakki Mert Torun and Madhavan Swaminathan. 2019. High-Dimensional Global Optimization Method for High-Frequency Electronic Design. *IEEE Transactions on Microwave Theory and Techniques* 67, 6 (2019), 2128–2142. <https://doi.org/10.1109/TMTT.2019.2915298>
 - [17] James Wholey and Issy Kipnis. Spring 1990. Silicon Bipolar Active Mixers. *Applied Microwave Journal* (Spring 1990), 287–293.
 - [18] Jianjun Xu, M.C.E. Yagoub, Runtao Ding, and Qi-Jun Zhang. 2002. Neural-based dynamic modeling of nonlinear microwave circuits. *IEEE Transactions on Microwave Theory and Techniques* 50, 12 (2002), 2769–2780. <https://doi.org/10.1109/TMTT.2002.805192>
 - [19] Huan Yu, Hemanth Chalamalasetty, and Madhavan Swaminathan. 2019. Modeling of Voltage-Controlled Oscillators Including I/O Behavior Using Augmented Neural Networks. *IEEE Access* 7 (2019), 38973–38982. <https://doi.org/10.1109/ACCESS.2019.2905136>
 - [20] Huan Yu, Hakki Mert Torun, Mutee Ur Rehman, and Madhavan Swaminathan. 2020. Design of SIW Filters in D-band Using Invertible Neural Nets. In *2020 IEEE/MTT-S International Microwave Symposium (IMS)*. 72–75. <https://doi.org/10.1109/IMS30576.2020.9223952>