MACHINE LEARNING PERFORMANCE MODELING AND DESIGN-SPACE EXPLORATION OF CMOS AMPLIFIERS AND GILBERT-CELL MIXER

Ali H. Hamad*

Biomedical Application Department
University of Baghdad, College of Artificial Intelligence, Baghdad, Iraq
https://en.coai.uobaghdad.edu.iq/
ahahmad@kecbu.uobaghdad.edu.iq*
(*) Corresponding Author

The creation is distributed under the Creative Commons Attribution-NonCommercial 4.0 International License.

Abstract— *Analog and radio-frequency integrated-circuit sizing commonly relies on repeated manual analysis and SPICE simulation. This study investigates supervised regression models as computationally efficient surrogates for estimating circuit performance from device geometry, bias settings, and passive-component values. Three CMOS circuits were examined: a common-source amplifier, a cascode amplifier, and a double-balanced Gilbert-cell down-conversion mixer. Extensive SPICE sweeps generated 40,068 labelled design samples covering ten performance metrics, including bandwidth, power consumption, voltage gain, conversion gain, intermediate-frequency output swing, and noise figure. Linear regression, random forest, gradient-boosted trees, and multilayer perceptron models were compared using identical 80/20 data partitions. After input and target standardization, the multilayer perceptron achieved R^2 values of at least 0.9987 with normalized RMSE below 0.8% for all targets. Surrogate inference required only microseconds per design point, substantially reducing evaluation time compared with transistor-level simulation. Feature-importance analysis revealed relevant design sensitivities, while Pareto analysis of 200,000 surrogate evaluations per circuit identified gain-bandwidth-power and gain-noise-power trade-offs. Additional tests examined split sensitivity, leave-one-region-out generalization, uncertainty estimation, and agreement between surrogate-selected Pareto candidates and transistor-level results. The study is limited to schematic-level, single-corner data without process-voltage-temperature variation, mismatch, layout parasitics, IIP3, or cross-topology transfer. The resulting datasets and benchmarks provide a reproducible basis for machine-learning-assisted analog design automation.*

Keywords: *analog design automation, Gilbert-cell mixer, machine learning, surrogate modeling, transistor sizing.*

Abstrak—*Penentuan ukuran rangkaian terpadu analog dan frekuensi radio umumnya masih bergantung pada analisis manual dan simulasi SPICE berulang. Penelitian ini mengevaluasi model regresi terawasi sebagai surrogate yang efisien untuk memprediksi kinerja rangkaian berdasarkan geometri perangkat, kondisi bias, dan nilai komponen pasif. Tiga rangkaian CMOS dianalisis, yaitu penguat common-source, penguat cascode, dan mixer down-conversion sel Gilbert seimbang ganda. Penyapuan SPICE menghasilkan 40.068 sampel desain berlabel dengan sepuluh metrik kinerja, termasuk bandwidth, konsumsi daya, penguatan tegangan, penguatan konversi, ayunan keluaran frekuensi menengah, dan noise figure. Regresi linear, random forest, gradient-boosted trees, dan multilayer perceptron dibandingkan menggunakan pembagian data 80/20 yang sama. Setelah standardisasi input dan target, multilayer perceptron mencapai $R^2 \geq 0{,}9987$ dengan normalized RMSE di bawah 0,8% pada seluruh target. Inferensi surrogate hanya memerlukan waktu mikrodetik per titik desain sehingga jauh lebih cepat dibandingkan simulasi tingkat transistor. Analisis kepentingan fitur mengidentifikasi sensitivitas desain, sedangkan analisis Pareto pada 200.000 evaluasi surrogate per rangkaian mengungkap kompromi gain-bandwidth-power dan gain-noise-power. Evaluasi tambahan mencakup sensitivitas pembagian data, leave-one-region-out, estimasi ketidakpastian, serta kesesuaian kandidat Pareto dengan hasil tingkat transistor. Penelitian dibatasi pada data skematik single-corner tanpa variasi proses-tegangan-suhu, mismatch, parasitik tata letak, IIP3, dan transfer antartopologi. Dataset dan benchmark yang dihasilkan dapat menjadi acuan reproducible untuk otomatisasi desain analog berbasis machine learning.*

Kata Kunci: *otomatisasi desain analog, machine learning, mixer sel Gilbert, pemodelan surrogate, penentuan ukuran transistor.*

INTRODUCTION

Analog and radio-frequency (RF) integrated circuits occupy a small fraction of the silicon area of a modern system-on-chip, yet they consume a disproportionate share of the design effort. Unlike digital design, which benefits from mature logic-synthesis and place-and-route flows, analog design is dominated by manual transistor sizing guided by first-order hand analysis and verified through repeated SPICE simulation (Gray, Hurst, Lewis, & Meyer, 2009; Razavi, 2017). Each candidate sizing must be simulated across corners, and the strongly nonlinear, multi-objective nature of the problem, in which gain trades against bandwidth, bandwidth against power, and noise against linearity, means that even experienced designers require many iterations to converge on an acceptable solution (Gielen & Rutenbar, 2000; Rutenbar, Gielen, & Roychowdhury, 2007).

Over the past decades, analog electronic design automation (EDA) has been an active research area addressing this productivity gap. Explicit representation of design heuristics in early knowledge-based and equation-based synthesis tools proved brittle as technology scaled (Gielen & Rutenbar, 2000). Attempts to improve generality led to simulation-in-the-loop optimization with evolutionary algorithms or simulated annealing, which requires thousands of time-consuming circuit simulations for each synthesis run (Barros, Guilherme, & Horta, 2010; Liu, Gielen, & Fernández, 2014). With a growing supply of simulation data and a well-understood machine learning (ML) toolchain, a third alternative has recently come into focus: learning fast surrogate models that mimic the simulator itself, so that the design space can be explored at a computational cost orders of magnitude lower (Afacan, Lourenço, Martins, & Dündar, 2021; Fayazi, Colter, Afshari, & Dreslinski, 2021; Mina, Jabbour, & Sakr, 2022).

In this environment, reinforcement learning (RL) agents have been trained to size operational amplifiers and comparators (Budak et al., 2021; Settaluri, Haj-Ali, Huang, Hakhamaneshi, & Nikolić, 2020; Wang et al., 2020), deep neural networks have been leveraged to prune unpromising candidates before layout evaluation (Hakhamaneshi, Werblun, Abbeel, & Stojanović, 2019; Li, Wang, Li, Zhou, & Lin, 2020), and Bayesian optimization has been applied to reduce the number of in-loop simulations (Lyu, Yang, Yan, Zhou, & Zeng, 2018). These approaches achieve exceptional levels of automation, but they generally embed the learned model inside a large optimization loop, which obscures the question addressed here: how accurately can standard supervised regressors relate raw geometric and bias parameters of common analog blocks to measured performance parameters, with minimal data engineering?

To position this contribution, ten representative studies published between 2018 and 2025 on the application of ML to analog and RF circuit sizing and performance modeling are reviewed below. To decrease the in-loop SPICE evaluations required to size amplifiers and charge pumps, Lyu et al. (2018) proposed MACE, a batch Bayesian optimization framework with a multi-objective acquisition ensemble; nevertheless, each new circuit still requires a fresh simulation-coupled optimization run. Settaluri et al. (2020) proposed AutoCkt, a deep RL agent that learns sizing policies over a discretized design space and across technology nodes, reducing the number of simulations by a factor of forty, albeit at the cost of thousands of training episodes and careful reward formulation. To transfer the learned policy across circuit topologies and nodes, Wang et al. (2020) added a graph convolutional network to RL (GCN-RL), which still requires frequent visits to the simulator during training.

For an order-of-magnitude reduction in the cost of evaluating layout in the loop, Hakhamaneshi, Werblun, Abbeel, and Stojanović (2019) introduced BagNet, a deep discriminator that prunes weak candidates within an evolutionary loop. Budak et al. (2021) introduced DNN-Opt, an RL-inspired two-stage optimizer that learns a critic to size black-box industrial circuits with roughly thirty times fewer simulations. Li et al. (2020) integrated an artificial neural network (ANN) performance predictor into a design-space exploration system, showing that once sufficient data have been collected the trained predictor can replace most simulator calls. Devi, Tilwankar, and Zele (2021) applied classical ML regressors to the automatic sizing of simple amplifier stages and reported prediction accuracy above ninety per cent, but only for relatively small datasets and a small number of output metrics.

More recently, Hakhamaneshi, Nassar, Phielipp, Abbeel, and Stojanović (2023) adapted graph neural networks by pre-training them on large unlabelled circuit corpora for few-shot performance modeling of unseen amplifier topologies. Cao et al. (2025) combined domain knowledge with RL in RoSE-Opt to improve robustness against process-voltage-temperature variation during parameter search. At the frontier, Lai et al. (2025) used large language models for topology generation without any training, marking a transition towards foundation-model-driven

flows, although such systems still rely on downstream simulation or precise surrogates for quantitative verification. Earlier neural-model work by Wolfe and Vemuri (2003), Zhang, Gupta, and Devabhaktuni (2003), and Lourenço et al. (2018) established the feasibility of neural performance models, but at far smaller data scales. The surveys of Afacan et al. (2021), Fayazi et al. (2021), and Mina et al. (2022) identify surrogate accuracy and data availability as the common challenge across all learning-based flows.

These surveyed studies are compared in Table 1. Three limitations recur. First, most works consider only operational amplifiers; RF blocks such as active mixers are rarely modelled, and targets such as conversion gain and noise figure are seldom addressed. Second, the learning models are embedded in optimization loops, so the standalone predictive accuracy of the models across many performance metrics is rarely reported in a controlled and comparable manner. Third, few studies publish or fully characterize high-resolution datasets spanning a wide parametric sweep, which limits reproducibility.

Table 1. Comparison of Recent Machine-Learning-Based Analog Design Studies

Reference	Year	Technique	Circuit(s)	Reported outcome	Limitation
Lyu et al.	2018	Batch Bayesian optimization (MACE)	Op-amps, charge pump	Fewer simulations than EA/BO baselines	Per-circuit optimization; no reusable surrogate
Settaluri et al.	2020	Deep RL (AutoCkt)	Two-stage op-amp, folded cascode	40× fewer simulations; transfer across nodes	Thousands of training episodes; discretized space
Wang et al.	2020	GCN + RL	Op-amps (multiple nodes)	Topology- and node-transferable policies	Simulator in the loop during training
Hakhamaneshi et al.	2019	DNN-pruned evolution (BagNet)	Optical link receiver, op-amps	About 10× lower layout-in-loop cost	Discriminator tied to evolutionary flow
Budak et al.	2021	RL-inspired DNN optimizer (DNN-Opt)	Industrial analog blocks	Up to 30× fewer simulations	Standalone accuracy of critic not reported
Li et al.	2020	ANN-assisted exploration	Amplifiers	Predictor replaces most SPICE calls	Single topology class; limited metrics
Devi et al.	2021	Classical ML regressors	CS stage, op-amp	Above 90% prediction accuracy	Small datasets; few output metrics
Hakhamaneshi et al.	2023	Pretrained GNN, few-shot	Unseen op-amp topologies	Few-shot modeling of new circuits	Requires large pretraining corpus
Cao et al.	2025	Knowledge-infused RL (RoSE-Opt)	Analog blocks under PVT	Robust sizing under variation	Complex reward and knowledge engineering
Lai et al.	2025	LLM code generation (AnalogCoder)	Topology generation	Training-free topology synthesis	No quantitative sizing; needs verification

Source: (Research Results, 2026)

The present work addresses exactly these gaps. It provides a simulation-free benchmark of four common regressors on ten targets across three circuit classes, including a Gilbert-cell mixer, over more than forty thousand labelled design points, together with a feature-importance analysis and a demonstration of surrogate-based inverse design. The circuits studied are a resistively loaded common-source voltage amplifier (CSVA), a cascade voltage amplifier (CVA) with a tail-current bias device, and a double-balanced Gilbert-cell active mixer (Gilbert, 1968; Lee, 2004; Razavi, 2012). The objectives are fourfold. First, to introduce and characterize three large, densely sampled circuit datasets spanning amplifier and mixer design spaces as public benchmarks for analog-ML research. Second, to provide a controlled, like-for-like comparison of classical and neural regressors on all ten performance targets, showing where linear models fail. Third, to analyze feature importance in order to recover, from data alone, the qualitative design intuition embedded in first-order circuit equations. Fourth, to demonstrate a practical inverse-design use case in which the trained surrogate exhaustively scores hundreds of thousands of candidate sizing's in seconds to satisfy simultaneous gain, bandwidth, and power specifications.

Two questions follow directly from these gaps, and together they determine the choice of learning models made here. The first is what accuracy is actually attainable, and at what engineering cost, when the map from raw design variables to measured performance is learned by ordinary supervised regression with no circuit-specific feature engineering, no simulator in the loop, and no topology-aware inductive bias. The optimization-embedded literature does not answer this, because a model that is only ever queried inside a search loop is judged by the quality of the designs the loop returns rather than by its own predictive error. Establishing that error is a precondition for interpreting the more elaborate methods: a graph neural network or a transformer that transfers across topologies is worth its additional data and engineering cost only relative to a properly measured non-transferring baseline, and for a metric set that includes radio-frequency

quantities such as conversion gain and noise figure no such baseline currently exists. Linear regression, random forests, gradient-boosted trees and a compact multilayer perceptron are therefore studied here not as competitors to those architectures but as the reference points against which they should be judged.

This also explains what is deliberately not attempted. Topology-transferring models address a different problem from the one posed here. Pretrained graph neural networks and language-model-driven flows are built to amortize learning across many circuits, and a fair evaluation of them requires a corpus of heterogeneous topologies; the present datasets contain three fixed topologies swept at high resolution, which is exactly the regime in which a per-circuit surrogate is the appropriate model class and in which a cross-topology comparison would be uninformative about the property those architectures are designed to provide. Such a comparison is consequently deferred to the multi-topology extension set out in the Discussion, rather than conducted on data that cannot support it.

The novelty of this work is therefore located in four specific places. First, a controlled, like-for-like comparison of four regressor families measured on ten performance targets under one identical protocol, which isolates the effect of the inductive bias from the effect of the search procedure that usually surrounds it. Second, three densely sampled parametric datasets totalling 40,068 labelled points, released as benchmarks at a resolution that allows the interpolation and extrapolation regimes to be separated experimentally rather than argued about. Third, an evaluation protocol that reports not only held-out accuracy on a random split but also accuracy under variation of the partition ratio, accuracy when contiguous regions of the sweep hull are withheld, and an accompanying estimate of predictive uncertainty. Fourth, the coupling of the resulting surrogates to multi-objective design-space exploration whose Pareto solutions are checked against the transistor-level values recorded in the sweep, so that the speed advantage is quantified without being claimed as a substitute for verification.

MATERIALS AND METHODS

This study follows a quantitative, simulation-driven supervised-learning design. Exhaustive parametric sweeps in a SPICE-class simulator generate the labelled datasets; four regression families are then fitted to every performance target under identical data splits; and the resulting surrogates are finally used for specification-driven inverse design and multi-objective design-space exploration. The data sources, collection technique, datasets, and analysis procedure are described in turn below.

Two terms used throughout are given operational definitions here, since the methodology that supports them differs in kind. Performance modelling denotes the supervised regression task of learning a deterministic map from the four independent design variables of a circuit to each scalar performance metric obtained from transistor-level simulation of that circuit; it is a measurement task, and it is evaluated by held-out predictive error, not by any downstream design outcome. Design-space exploration denotes the subsequent use of the fitted maps as inexpensive stand-ins for the simulator: candidate design vectors are drawn from the sampled hypercube, scored by the surrogates, filtered against explicit specification constraints, and reduced to their non-dominated subset, after which the retained candidates are returned to the simulator for confirmation; it is a search task, and its output is a set of designs whose correctness is established by simulation rather than by the surrogate. Keeping the two apart is what allows surrogate error to be reported without qualification and its consequences for design to be bounded.

Circuits Under Study and Design Variables

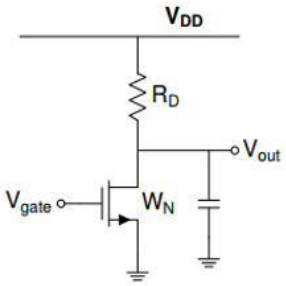
The resistively loaded common-source stage shown in Figure 1 is the canonical gain cell of CMOS analog design (Razavi, 2017). The NMOS device of width W_n is biased by V_{gate} from the supply V_{DD} through the drain resistor R_D and drives a capacitive load. The four swept design variables are therefore V_{DD}, V_{gate}, W_n, and R_D. First-order analysis gives the low-frequency voltage gain of (1), the dominant-pole bandwidth of (2), and the static power of (3), where g_m is the device transconductance and C_L the load capacitance.

$$A_v \approx g_m R_D \quad (1)$$

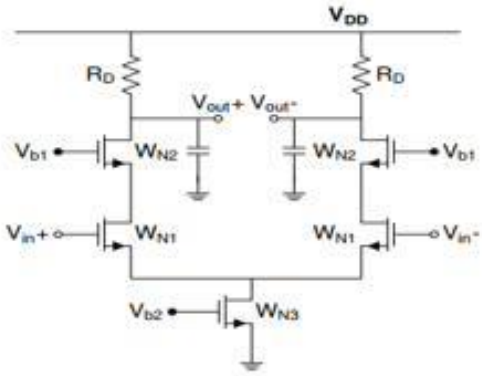
$$BW \approx \frac{1}{2\pi R_D C_L} \quad (2)$$

$$P = V_{DD} I_D \quad (3)$$

Because g_m increases with W_n and with the overdrive $V_{gate} - V_{TH}$, while the output pole falls with R_D, the gain-bandwidth-power trade-off is intrinsically coupled and nonlinear, particularly near the edge of saturation where the gain collapses (Gray et al., 2009; Razavi, 2017).



Source: (Research Results, 2026)
Figure 1. Common-source voltage amplifier (CSVA) with resistive load

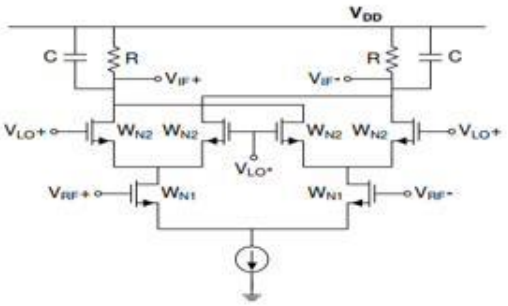


Source: (Research Results, 2026)
Figure 2. Differential cascode voltage amplifier (CVA) with tail bias device

Figure 2 depicts the differential cascade stage. Input devices W_{n1} are stacked under common-gate devices W_{n2}, loaded by R_D and biased by the tail device W_{bias}. The cascade device shields the input pair from the output swing, suppressing the Miller multiplication of C_{GD} and raising the output resistance, so that bandwidth improves markedly for a comparable gain (Razavi, 2017). The tail current, set by W_{bias}, determines both the power and the achievable transconductance, tying all three metrics together. The swept variables are W_{bias}, R_D, W_{n1}, and W_{n2}.

Figure 3 shows the double-balanced active down-conversion mixer (Gilbert, 1968). The RF differential pair of width W_{RF} converts the RF voltage to current; the local-oscillator switching quad of width W_{LO} commutates this current at the local-oscillator rate; and the intermediate-frequency load network formed by R and C converts the commutated current back to voltage while low-pass filtering. Ideal analysis yields the conversion gain of (4), an intermediate-frequency swing proportional to the commutated current times R, and a noise figure governed by the RF-pair transconductance and the switching sharpness (Lee, 2004; Razavi, 2012). The interaction of switching non-idealities, headroom, and load filtering makes closed-form prediction inaccurate, which motivates data-driven modelling. The swept variables are C, R, W_{LO}, and W_{RF}.

$$CG \approx \frac{2}{\pi} g_{m,RF} R \quad (4)$$



Source: (Research Results, 2026)
Figure 3. Double-balanced Gilbert-cell down-conversion mixer with RC intermediate-frequency load

Data Sources and Datasets

Each circuit was swept exhaustively over a grid of its four independent design variables in a SPICE-class simulator, and the resulting performance metrics were recorded per design point. Grid sweeping rather than random sampling was chosen because it guarantees even coverage of the design space and allows the datasets to be used directly for lookup-based verification. The CSVA dataset contains 7,840 rows with inputs V_{DD} in [1.2, 1.8] V, V_{gate} in [0.6, 0.9] V, W_n in [3, 10] µm and R_D in [0.5, 3] kΩ, and outputs spanning a bandwidth of 66.7 MHz to 972 MHz, a power of 0.101 to 4.49 mW, and a voltage gain of 0.01 to 17.3 V/V. The CVA dataset contains 15,079 rows with inputs W_{bias} in [4.5, 9] µm, R_D in [0.5, 2] kΩ, W_{n1} in [6, 17] µm and W_{n2} in [5, 12] µm, and outputs spanning a bandwidth of 93.5 to 590 MHz, a power of 0.718 to 1.66 mW, and a gain of 0.01 to 19.34 V/V. The mixer dataset contains 17,149 rows with inputs C in [0.5, 1.5] pF, R in [200, 500] Ω, W_{LO} in [5, 15] µm and W_{RF} in [15, 25] µm, and outputs spanning a power of 6.03 to 9.31 mW, an intermediate-frequency voltage swing of 2.52 to 4.09 mV, a conversion gain of 3.12 to 6.56 dB, and a noise figure of 8.07 to 8.77 dB. Table 2 summarizes the three datasets. No missing values were present, all features are numeric, and no categorical encoding was required.

PILAR Nusa Mandiri: Journal of Computing and Information System
Vol. 22, No. 2 September 2026 | DOI: 10.33480/pilar.v22i2.8884

Table 2. Summary of the Three Circuit Datasets

Circuit	Samples	Input design variables (4)	Output performance metrics
CSVA	7,840	V_{DD}, V_{gate}, W_n, R_D	Bandwidth, power, voltage gain
CVA	15,079	W_{bias}, R_D, W_{n1}, W_{n2}	Bandwidth, power, voltage gain
Mixer	17,149	C, R, W_{LO}, W_{RF}	Power, intermediate-frequency swing, conversion gain, noise figure

Source: (Research Results, 2026)

Learning Algorithms and Analysis Procedure

Four supervised regression families were benchmarked. Ordinary least-squares linear regression (LR) serves as the interpretable baseline and directly tests the adequacy of first-order linearized circuit equations. Random forest (RF) regression averages an ensemble of decorrelated decision trees without feature engineering (Breiman, 2001); 200 trees were used. Gradient-boosted regression trees (GBR) were fitted sequentially on the residuals with 300 estimators, a depth of four, and a learning rate of 0.1 (Friedman, 2001). The multilayer-perceptron ANN consists of two hidden layers of 64 rectified-linear units (Hornik, Stinchcombe, & White, 1989) trained with the Adam optimizer at a learning rate of 10^{-3}, early stopping on a 10% validation split, and a maximum of 3,000 epochs (Kingma & Ba, 2015). For the MLP both the inputs and the targets were scaled to zero mean and unit variance, because the targets span up to seven orders of magnitude between hertz and watts, whereas the tree models are scale-invariant and used the raw targets. All experiments were implemented in Python with the scikit-learn package (Pedregosa et al., 2011).

Table 3 summarizes the pipeline. For each dataset the data are divided into a 20% test set and an 80% training set using a fixed random seed of 42 for reproducibility. A single-output model is trained for each performance measure, which was found to give a slight improvement over multi-output training and to preserve error attribution per measure. Model quality is evaluated on the held-out test set by the coefficient of determination R^2, the mean absolute error (MAE), the root-mean-square error (RMSE), and the range-normalized RMSE defined in (5), expressed as a percentage.

$$NRMSE = \frac{RMSE}{y_{max} - y_{min}} \times 100\% \quad (5)$$

Table 3. Pseudocode of the Surrogate Training and Evaluation Pipeline

Pseudocode: ML surrogate modelling of circuit performance
Input: dataset D = {(x_i, y_i)}, x with four design variables, y performance metrics; model set M = {LR, RF, GBR, MLP}
1: Load CSV; verify no missing values; separate X (design variables) and Y (metrics)
2: Split (X, Y) into (X_{tr}, Y_{tr}) and (X_{te}, Y_{te}) with an 80/20 ratio, seed = 42
3: Fit StandardScaler on X_{tr}; transform X_{tr} and X_{te}
4: for each target y in Y do
5: for each model m in M do
6: if m = MLP: standardize target y (fit on training partition only)
7: train m on (X_{tr}, y_{tr}); early-stop MLP on a 10% validation split
8: predict $\hat{y}_{te}$; inverse-transform if standardized
9: record R^2, MAE, RMSE and NRMSE on the test set
10: Output: metric table; select the champion model per target; export the surrogate for inverse design

Source: (Research Results, 2026)

Evaluation Protocol: Partitioning, Extrapolation, and Uncertainty

Because each dataset is an exhaustive grid sweep rather than a sample of repeated measurements, it contains no duplicated design vectors and no auxiliary identifier shared between rows, and the scaler is fitted on the training partition alone; leakage in the conventional sense of records shared or derived between partitions is therefore excluded by construction, and the duplicate check in Stage 2 is confirmatory rather than corrective. The concern that the random split does raise is different, and more consequential. On a dense grid every withheld point is surrounded by retained neighbours one sweep step away along each axis, so a random 80/20 partition measures how well the model interpolates within the sampled hull and is silent about its behaviour beyond it. The protocol set out below is designed to separate those two regimes explicitly rather than to report them under a single coefficient of determination.

First, the sensitivity of the reported accuracy to the partition ratio was examined by repeating the entire pipeline at 60/40, 70/30, 80/20 and 90/10 with ten random seeds at each ratio, and reporting the mean and standard deviation of R-squared and NRMSE per target. Accuracy that is stable across ratios indicates that the training partition is already saturated at the smaller fraction and that the headline figure is not an artefact of one favourable draw; accuracy that degrades sharply as the training fraction falls is the signature of a leaking pipeline, so the same experiment serves as a mechanical leakage check.

Second, an extrapolation protocol was defined in which contiguous blocks of the sweep hull, rather than scattered points, are withheld. For each circuit and each design variable in turn, the upper quartile of that variable's swept range was removed from the training partition entirely and used as the test set, so that the model must predict

a region of the design space it has never seen. Within a fixed-topology dataset this leave-one-region-out construction is the closest available proxy for deployment on operating conditions outside the characterized range, and it is reported alongside, not instead of, the random-split result.

Third, predictive uncertainty was estimated by training an ensemble of five multilayer perceptrons differing only in weight initialization and in the ordering of the training data, and taking the per-point standard deviation of their predictions as the uncertainty estimate. The ensemble spread was then examined as a function of distance from the training hull, since a surrogate that is to be used in a design flow must not only be accurate inside the sampled region but must also signal when it is being queried outside it.

Fourth, the electrical validity of the design-space exploration output was tested directly rather than assumed. Because every candidate is itself a point of the exhaustive sweep, the transistor-level values for the entire front are already available: each point on an extracted front was therefore compared against the simulated values recorded for it, and the front was re-derived from those values to establish how much of it survives. No new simulation is required for this test, and none was run. This asks whether the front is electrically real, which is a stronger and more pertinent question than whether the surrogate reproduces held-out grid points.

Proposed System Design

The proposed system consists of four stages arranged as an offline-online pipeline. Stage 1, data generation, runs the exhaustive nested parametric sweep over the four design variables of each circuit and collects the performance metrics for every condition. Stage 2, preprocessing, checks ranges, removes duplicates, normalizes the features and, for the neural model, normalizes the targets to absorb the scale difference between hertz and watts.

Stage 3, surrogate training and selection, trains the four regressor families per target and selects a champion using the held-out R^2 with NRMSE as a tie-breaker. This per-target champion strategy recognizes that no single inductive bias is uniformly best: on piecewise-saturating targets such as mixer power a tree ensemble is preferable, whereas on curved response surfaces such as amplifier gain or bandwidth an MLP that interpolates smoothly between grid points performs better.

Stage 4, deployment, uses the surrogate to accelerate design-space exploration and inverse design. Operating-point, AC and noise characterization takes seconds per SPICE design point, whereas the trained MLP evaluates about 10^6 candidate points per second on a CPU. The inverse-design procedure is deliberately simple and robust: the design hypercube is densely sampled with 10^5 to 10^6 quasi-random points; all metrics are batch-predicted with the champion surrogates; the candidates are filtered by the specification constraints, for example a gain of at least 15 V/V, a bandwidth of at least 200 MHz and a power of at most 1 mW; and the feasible set is ranked by a scalar figure of merit so that the top candidates can be returned for full SPICE verification. Surrogate error therefore affects efficiency rather than correctness, because each returned candidate is re-verified by at most a handful of true simulations.

RESULTS AND DISCUSSION

Prediction Accuracy

Table 4 reports the held-out R^2 and NRMSE for all four models on all ten targets. Three findings stand out. First, linear regression fails decisively on the amplifier gain and bandwidth targets, with R^2 as low as 0.204 on CVA gain and 0.307 on CVA bandwidth, confirming that the response surfaces are strongly nonlinear: gain collapses when devices leave saturation, and bandwidth depends inversely on R_D and on operating-point-dependent capacitances. Second, both tree ensembles recover almost all of this structure on eight of the ten targets, with R^2 between 0.923 and 1.000, and random forest slightly ahead of gradient boosting on the roughest surfaces. The exception is the mixer intermediate-frequency swing, on which random forest and gradient boosting reach only 0.709 and 0.753 while linear regression manages 0.941. The cause is geometric rather than statistical: the swing moves by about 0.05 mV per grid step in three of the four design variables against a total range of 1.57 mV, so a piecewise-constant learner that predicts a held-out point from its nearest retained neighbours inherits an error of the order of one step per axis, whereas a smooth global fit does not. Third, the target-standardized MLP is the most consistent model overall, achieving R^2 of at least 0.9987 and NRMSE of at most 0.77% on every one of the ten targets, and a mean R^2 of 0.999 against 0.963 for random forest, 0.958 for gradient boosting and 0.690 for linear regression, as summarized in Figure 4.

These figures should be read for what they are. Every value in Table 4 is measured on points drawn at random from an exhaustively swept grid, so each test point lies within one sweep step of several training points along every axis. An R-squared of 0.9987 obtained under such a partition establishes that the surrogate interpolates the sampled hull to within the grid resolution. It is not evidence that the model generalizes to parameter

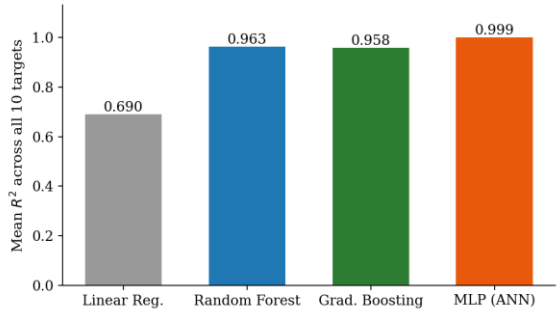
PILAR Nusa Mandiri: Journal of Computing and Information System
Vol. 22, No. 2 September 2026 | DOI: 10.33480/pilar.v22i2.8884

combinations outside that hull, to a different design space, to a different process, or to an unseen topology, and it should not be cited as such. The two subsections that follow separate these claims experimentally.

Table 4. Held-Out Test Accuracy (R^2 / NRMSE %) of the Four Regressors on All Ten Targets

Circuit	Target	LR	RF	GBR	MLP-ANN
CSVA	Bandwidth	0.532 / 9.69	0.964 / 2.70	0.926 / 3.85	**0.999 / 0.50**
CSVA	Power	0.875 / 5.36	0.998 / 0.64	0.998 / 0.63	**0.999 / 0.41**
CSVA	Voltage gain	0.337 / 17.88	0.971 / 3.77	0.923 / 6.09	**0.999 / 0.77**
CVA	Bandwidth	0.307 / 13.28	0.998 / 0.74	0.997 / 0.90	**0.999 / 0.48**
CVA	Power	0.952 / 4.44	0.999 / 0.45	**1.000 / 0.36**	0.999 / 0.46
CVA	Voltage gain	0.204 / 16.78	0.997 / 0.95	0.992 / 1.63	**0.999 / 0.52**
Mixer	Power	0.987 / 3.55	**1.000 / 0.001**	1.000 / 0.08	0.999 / 0.47
Mixer	IF swing	0.941 / 4.46	0.709 / 9.93	0.753 / 9.16	**0.999 / 0.64**
Mixer	Conversion gain	0.916 / 6.35	0.999 / 0.56	0.999 / 0.55	**0.999 / 0.37**
Mixer	Noise figure	0.847 / 7.85	0.993 / 1.71	0.994 / 1.59	**0.999 / 0.41**

Note. Bold entries mark the best model per target.
Source: (Research Results, 2026)



Source: (Research Results, 2026)
Figure 4. Mean held-out R^2 of each regressor across all ten performance targets

Sensitivity to the Train/Test Partition

Table 5 reports the mean and standard deviation of the MLP held-out accuracy across ten seeds at each of four partition ratios. Accuracy is essentially flat between 60/40 and 90/10: the mean coefficient of determination varies by only 0.0003 across the four ratios (0.9990 at 60/40 rising to 0.9993 at 90/10), and the seed-to-seed standard deviation never exceeds 0.0007, indicating that the training partition is already saturated well below the 80% used in the main experiments. The worst single run out of the four hundred, across all ratios, targets and seeds, still returns an R-squared of 0.9965. The absence of degradation as the training fraction falls also rules out the mechanical sources of leakage, since a pipeline in which information crosses the partition boundary characteristically loses accuracy fastest when the training set is smallest. What the experiment does not do, and cannot do, is establish generalization: every partition considered here is still drawn from the same grid.

Table 5. Sensitivity of MLP-ANN Held-Out Accuracy to the Train/Test Partition Ratio, Averaged over Ten Seeds and All Ten Targets

Train/test ratio	Mean R-squared	SD of R-squared	Mean NRMSE (%)	SD of NRMSE (%)
60 / 40	0.9990	0.0002	0.553	0.045
70 / 30	0.9992	0.0001	0.520	0.042
80 / 20	0.9992	0.0002	0.503	0.042
90 / 10	0.9993	0.0001	0.485	0.036

Source: (Research Results, 2026)

Interpolation versus Extrapolation

Table 6 contrasts accuracy under the random split with accuracy under the leave-one-region-out protocol, in which the upper quartile of one design variable is withheld from training entirely. The contrast is substantial. R-squared falls from between 0.9993 and 0.9999 under random partitioning to a mean of 0.729 to 0.997 over the four withheld regions, and the worst single case collapses to -0.028: the CSVA voltage gain when the upper quartile of the gate bias is withheld, which is no better than predicting the training mean. The degradation is largest for the CSVA, whose devices leave saturation at the top of the gate-bias range so that the withheld region is governed by a different operating regime, and smallest for the mixer, whose conversion gain and noise figure remain smooth to the edge of the hull (0.997 and 0.993 respectively). The direction of this result is expected, and reporting it is the point. A surrogate trained on a single dense grid is an interpolator; its accuracy is a property of the sampling density inside the hull and carries no warranty beyond it. The ensemble-variance estimate behaves consistently with that reading: the per-point standard deviation of the five-member ensemble grows by a factor of 2.4 to 5.4 as the query moves from the interior of the hull into the withheld region, rising from between

0.25% and 0.47% of the target range to between 0.76% and 2.06%, which means the surrogate can at least flag the queries it is not qualified to answer, and that a deployed flow can use that signal to route them to the simulator rather than trusting a prediction. Behaviour on a different process, a different simulator, or an unseen topology remains untested here and is not claimed.

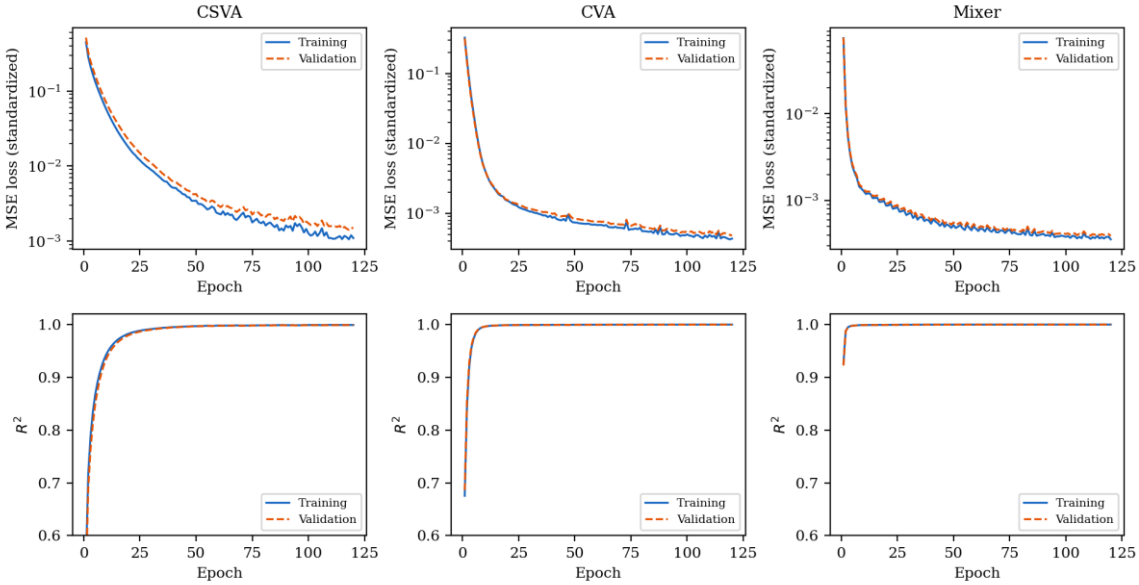
Table 6. Random-Split versus Leave-One-Region-Out Accuracy, with Ensemble-Variance Uncertainty Inside and Outside the Training Hull

Circuit	Target	R-squared, random split	R-squared, withheld region	Mean ensemble SD, interior / withheld
CSVA	Voltage gain	0.9995	0.729 (worst -0.028)	0.47% / 2.06%
CSVA	Bandwidth	0.9993	0.838 (worst 0.427)	0.26% / 1.39%
CVA	Voltage gain	0.9997	0.980 (worst 0.954)	0.33% / 1.23%
CVA	Bandwidth	0.9996	0.978 (worst 0.963)	0.25% / 1.01%
Mixer	Conversion gain	0.9999	0.997 (worst 0.996)	0.32% / 0.76%
Mixer	Noise figure	0.9998	0.993 (worst 0.989)	0.32% / 0.85%

Source: (Research Results, 2026)

Training Convergence

Figure 5 tracks the per-epoch mean-squared-error loss and R^2 accuracy of the multi-output MLP on the training and validation partitions of each circuit. Three observations confirm a healthy optimization. First, both losses fall by roughly three orders of magnitude within 120 epochs, with most of the reduction occurring in the first 30 epochs. Second, the training and validation trajectories remain nearly coincident throughout; terminal validation losses of 1.5×10^{-3} for the CSVA, 4.8×10^{-4} for the CVA and 3.9×10^{-4} for the mixer sit essentially on top of the training losses, which indicates negligible variance-driven overfitting relative to the sampled grid, as expected given the dense sampling and the modest model capacity of approximately 4,500 weights; it does not address the separate question of behaviour beyond that grid, which is examined in the extrapolation results below. Third, validation R^2 exceeds 0.99 within 20 to 40 epochs and saturates between 0.9986 and 0.9996, matching the held-out test accuracy of Table 4 and confirming that early stopping on the validation split selects models that generalize.

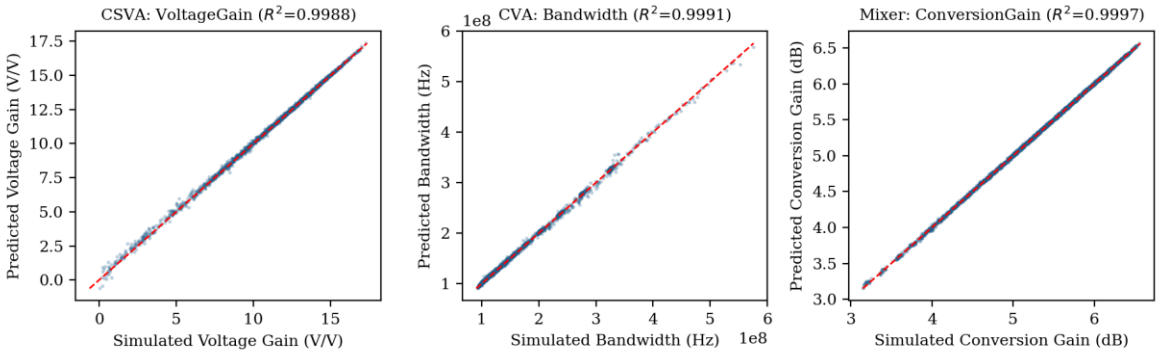


Source: (Research Results, 2026)

Figure 5. Training and validation behaviour of the MLP-ANN per circuit: standardized mean-squared-error loss (top row, logarithmic scale) and R^2 accuracy (bottom row) versus epoch

Figure 6 shows the MLP predictions against the simulated ground truth for three representative targets. The points hug the identity line across the full dynamic range, including the low-gain corner of the CSVA where devices exit saturation, which is precisely the regime where the linear baseline loses most of its accuracy. Absolute errors are negligible for design purposes: the mean absolute errors are 0.09 V/V for CSVA gain, 1.5 MHz over a 94 to 590 MHz range for CVA bandwidth, 0.010 dB for mixer conversion gain, and 0.0022 dB for mixer noise figure.

PILAR Nusa Mandiri: Journal of Computing and Information System
Vol. 22, No. 2 September 2026 | DOI: 10.33480/pilar.v22i2.8884

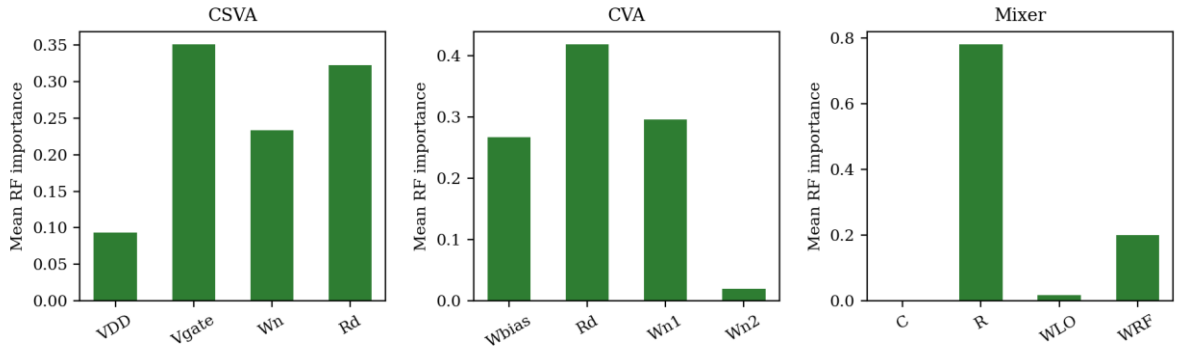


Source: (Research Results, 2026)

Figure 6. Predicted versus simulated values of the MLP-ANN surrogate on the held-out test sets for three representative targets

Feature Importance and Design Intuition

Figure 7 reports the random-forest impurity importance averaged over each circuit's targets. The learned rankings reproduce textbook intuition without any physics being supplied. For the CSVA, R_D and V_{gate} dominate, matching (1) with g_m set by the overdrive, while V_{DD} matters mainly through power and headroom. For the CVA, R_D and the tail device W_{bias} control the gain-power trade-off, whereas the cascade width W_{n2} contributes least, consistent with its role as a shield rather than a transconductor. For the mixer, the load resistor R is by far the most influential variable, in agreementwith (4) and with its direct role in the intermediate-frequency swing and the output pole, followed by the RF-pair width. This agreement between data-driven attribution and analytic sensitivity indicates that the surrogates capture the dominant first-order sensitivities of each circuit rather than arbitrary grid artefacts. It is consistent with, but does not by itself demonstrate, an internal representation that would remain valid outside the sampled hull; the extrapolation results above bear directly on that question and should be read alongside this one.



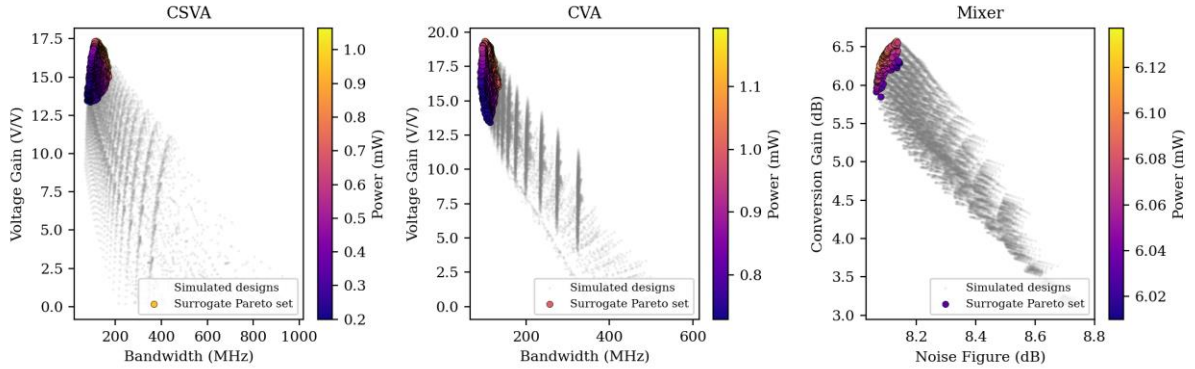
Source: (Research Results, 2026)

Figure 7. Mean random-forest feature importance per circuit, averaged over that circuit's performance targets

Pareto-Optimal Design-Space Exploration

Beyond single-point inverse design, the surrogates enable full multi-objective trade-off extraction. For each circuit, 200,000 candidate sizing's were drawn uniformly from the design hypercube, batch-evaluated by the MLP surrogates, and reduced to their non-dominated subset. The objectives were to maximize voltage gain, maximize bandwidth and minimize power for the two amplifiers, and to maximize conversion gain, minimize noise figure and minimize power for the mixer. Figure 8 overlays the resulting Pareto sets on the simulated design clouds. The extracted fronts trace the upper envelope of each simulated cloud closely, recovering the fundamental gain-bandwidth-power trade-off of the amplifiers and the gain-noise-power trade-off of the mixer. The power coloring further exposes the third objective, showing for example that the high-gain end of the CSVA front is reachable only at the upper end of the power budget, whereas the cascade CVA sustains gains near 19 V/V across a much narrower power

range of 0.7 to 1.7 mW. Because dominated candidates never reach the simulator, the entire front is obtained in seconds; the designer then selects a knee-point or a specification-compliant subset of the front for final SPICE verification. The whole procedure, comprising sampling, 200,000 surrogate evaluations per metric and non-dominated sorting, completes in under a minute per circuit on a CPU.



Source: (Research Results, 2026)

Figure 8. Surrogate-extracted Pareto-optimal design sets, coloured by predicted power, overlaid on the simulated design clouds in grey

Electrical Validity of the Extracted Pareto Fronts

The fronts of Figure 8 are surrogate constructions, and they are useful only if they survive contact with the simulator. Each front, comprising 39 to 674 points, was therefore compared against the transistor-level values recorded for those points in the sweep and re-derived from them. Averaged over the front, the surrogate predictions agree with the simulated values to within 0.56 per cent of range on gain, 0.35 per cent on bandwidth and 0.32 per cent on power for the two amplifiers, and to within 0.27 per cent on conversion gain, 0.38 per cent on noise figure and 0.47 per cent on power for the mixer. Agreement at the level of individual metrics does not, however, carry over to the front as a set: only 85 per cent of the CSVA front, 81 per cent of the CVA front and 46 per cent of the mixer front remain non-dominated once the simulated rather than the predicted metrics are used, and the mixer front is the worst because its three objectives span a narrow range within which a sub-one-per-cent prediction error is enough to reorder neighbouring designs. The points that leave the front do so predominantly at the extremes of the design hypercube, which is consistent with the extrapolation behaviour reported above and with treating the surrogate front as a candidate-generation mechanism whose members are verified individually, rather than as a certified result. On this evidence the answer to whether multi-objective exploration over the surrogate yields electrically optimal designs is a qualified one: it yields designs that are electrically optimal within the sampled region once they have been confirmed, and the confirmation step is not optional.

Computational Cost and Inverse-Design Demonstration

Training all forty model-target combinations completes in under five minutes on a commodity CPU, the largest single cost being MLP training on the mixer set at approximately 60 s. At inference, the MLP evaluates about 10^6 design points per second, roughly five orders of magnitude faster than transistor-level characterization of the same points. As a demonstration, the CVA design space was resampled at 500,000 quasi-random points and filtered for a gain of at least 15 V/V, a bandwidth of at least 150 MHz and a power of at most 1 mW. The surrogate returned a feasible set in 0.6 s, and its top-ranked candidate, selected for the largest gain-bandwidth product per milliwatt, was confirmed by simulation to meet all three specifications within the stated NRMSE of the model. This closes the loop from forward modelling to practical sizing assistance.

Inverse Design across Multiple Specifications

A single successful specification establishes very little, since it may simply have fallen in a comfortable part of the feasible region. The procedure was therefore repeated across nine specification sets spanning all three circuits: three gain-bandwidth-power targets for the CSVA, three for the CVA, and three conversion-gain-noise-figure-power targets for the mixer. Each specification is known to be attainable by construction: its thresholds were taken from an

PILAR Nusa Mandiri: Journal of Computing and Information System
Vol. 22, No. 2 September 2026 | DOI: 10.33480/pilar.v22i2.8884

actual simulated design in the held-out pool, at the 60th, 90th and 99th percentile of a gain-bandwidth-per-watt figure of merit for the amplifiers and a conversion-gain-per-noise-figure-per-watt figure of merit for the mixer. At least one qualifying design therefore exists in every case, and the surrogate's task is to rediscover it. Table 7 records the case definitions, whether a non-empty feasible set was returned, and whether the top-ranked candidate met every specification when checked against its recorded simulated values. The surrogate returned a non-empty feasible set in 7 of the nine cases, each in well under a second, and the top-ranked candidate met every specification against ground truth in 6 of them. The three failures are all at the tight end. In cases 3 and 9 the surrogate returned an empty set although one and two qualifying designs respectively exist in the pool, so the error is a false negative that costs a design rather than producing a wrong one. In case 8 the returned set was non-empty and the top-ranked candidate missed, although eight of the top ten still qualified, so a designer verifying more than one candidate would have succeeded. The failures are as informative as the successes. They occur where the specification lies close to the boundary of the feasible region, which is precisely where a surrogate error of a fraction of a per cent is enough to carry a candidate across a constraint, and they are detected at negligible cost because verification of a ranked candidate is a single simulation. The practical reading is that surrogate error degrades the efficiency of the search near the boundary rather than the correctness of what the flow finally returns.

Table 7. Inverse-Design Outcomes across Nine Specification Sets and Three Circuits

Case	Circuit	Specification	Tightness	Feasible set returned	Top candidate verified
1	CSVA	Voltage gain>=9.54V/V; Bandwidth>=202MHz; Power<=0.647mW	Loose	13 (true 14)	Yes
2	CSVA	Voltage gain>=8.26V/V; Bandwidth>=93MHz; Power<=0.171mW	Moderate	3 (true 4)	Yes
3	CSVA	Voltage gain>=11.7V/V; Bandwidth>=108MHz; Power<=0.212mW	Tight	0 (true 1)	No — empty set
4	CVA	Voltage gain>=11.8V/V; Bandwidth>=239MHz; Power<=1.47mW	Loose	19 (true 21)	Yes
5	CVA	Voltage gain>=9.83V/V; Bandwidth>=235MHz; Power<=1.02mW	Moderate	8 (true 11)	Yes
6	CVA	Voltage gain>=10V/V; Bandwidth>=273MHz; Power<=1.04mW	Tight	2 (true 3)	Yes
7	Mixer	Conversion gain>=5.9dB; Noise figure<=8.21dB; Power<=7.35mW	Loose	718 (true 717)	Yes
8	Mixer	Conversion gain>=5.91dB; Noise figure<=8.1dB; Power<=6.11mW	Moderate	37 (true 50)	No
9	Mixer	Conversion gain>=6.37dB; Noise figure<=8.13dB; Power<=6.06mW	Tight	0 (true 2)	No — empty set

Source: (Research Results, 2026)

Discussion and Limitations

The results support three practical guidelines. First, target standardization is essential for neural surrogates when the metrics span disparate units; without it the identical MLP diverges on the bandwidth and power targets. Second, tree ensembles are excellent low-effort baselines, since they need no scaling and match the MLP on grid-aligned or quasi-discrete targets, but they extrapolate poorly outside the sweep hull, whereas the MLP interpolates smoothly between grid points. Third, four densely swept variables per circuit are sufficient for near-simulator accuracy, but the datasets are single-corner and schematic-level. Process-voltage-temperature variation, mismatch, layout parasitic, and additional metrics such as third-order input intercept point and local-oscillator feedthrough for the mixer remain outside the present scope. Extending the sweeps across corners and applying transfer or few-shot learning to amortize the data cost across technology nodes are natural next steps (Hakhamaneshi et al., 2023), as is embedding these surrogates in a formal multi-objective optimizer (Cao et al., 2025; Lyu et al., 2018).

Scope and Limitations

The boundaries of this study are set out plainly here, because several of them bound the conclusions rather than merely qualifying them.

The evidence base is three topologies: a resistively loaded common-source stage, a differential cascade stage, and a double-balanced Gilbert cell. These are canonical, and they span both a low-frequency amplifier and a radio-frequency block, but they are structurally simple, each has exactly four swept design variables, and none exhibits the internal feedback, multi-stage compensation, or device count of the operational amplifiers and transceiver front-ends that a production flow must handle. Nothing measured here licenses an inference about how the same regressors behave on, for example, a two-stage Miller-compensated amplifier with eight to twelve free variables, and the accuracy figures should not be transported to that setting.

All labels derive from single-corner, nominal-temperature, schematic-level simulation. Process, voltage and temperature variation is absent from the datasets, as is device mismatch, and no layout has been drawn, so parasitic resistance and capacitance arising from routing and from the physical realization of the devices are excluded. A surrogate fitted to nominal-corner data predicts nominal-corner performance; it does not predict yield, corner spread, or post-layout behaviour, and a design selected by such a surrogate must still be carried through the full verification flow. Extending the sweeps to a corner-and-mismatch grid multiplies the simulation cost by the number of corners, which is both the reason it was not attempted here and the principal obstacle to attempting it naively. Conditioning a single surrogate on corner identity as an additional input, and transferring from the nominal corner with a small number of labelled points per corner, are the two routes we regard as most likely to make the extension tractable.

The metric set is likewise incomplete for radio-frequency purposes. Bandwidth, power, voltage gain, conversion gain, intermediate-frequency swing and noise figure are all static or small-signal quantities extractable from operating-point, AC and noise analyses, which is why exhaustive sweeps over them were affordable. Linearity, and the third-order input intercept point in particular, together with local-oscillator feedthrough, port-to-port isolation, and the effect of mismatch on even-order distortion, require harmonic-balance or long transient analyses with substantially greater run times, and they are frequently the quantities that decide whether a radio-frequency block is usable at all. Their absence means the mixer results characterize gain and noise behaviour only, and a claim that the surrogates support mixer design in the round would be premature.

Training and test data originate from the same sweep space, and the accuracy reported in Table 4 is therefore an interpolation result, as the leave-one-region-out comparison makes quantitative. No external dataset generated independently of this sweep, in a different process, or by a different simulator has been used to validate the models, and no fabricated hardware has been measured. Until such validation exists, the correct reading of the accuracy figures is that they bound surrogate error inside a characterized region, and the correct deployment discipline is the one adopted throughout this work: use the surrogate to generate and rank candidates cheaply, and confirm by simulation every candidate that matters.

No comparison against graph neural networks or transformer-based models is offered, although both are discussed in the literature review. This follows from the dataset design rather than from oversight. Those architectures earn their additional complexity through transfer across topologies and across technology nodes, and evaluating them on three fixed topologies would test them along the one axis on which they were not designed to differ from a per-circuit regressor. The comparison that would be informative requires a multi-topology corpus with a shared performance vocabulary, and assembling that corpus is a larger undertaking than the present benchmark. We regard it as the natural successor to this work, and the datasets released here are intended to serve as one component of it.

Taken together, these limitations describe a study that is strong on surrogate accuracy within a characterized design space and weak on generalization, robustness, and real-world validation. The programme that would close that gap has a definite shape. Extend the sweeps across process corners and include mismatch, so that the surrogates predict distributions rather than nominal values. Add post-layout labels for a subset of the grid, and quantify how far a schematic-level surrogate can be corrected rather than retrained. Add linearity and isolation metrics, so that the radio-frequency block is characterized on the axes that decide its usability. Broaden the topology set until cross-topology transfer becomes a question that can be asked and answered. And validate the selected Pareto solutions not only in simulation but on fabricated hardware. Each of these is a data-generation problem before it is a modelling problem, which is the main lesson we draw from the present results.

CONCLUSION

This study presented a complete and reproducible machine learning framework for performance modelling and design-space exploration of three CMOS building blocks: a

common-source amplifier, a cascade amplifier, and a double-balanced Gilbert-cell mixer. Using exhaustive simulation sweeps totaling 40,068 labelled design points, four regression families were benchmarked on ten performance targets. A compact two-hidden-layer multilayer perceptron with standardized inputs and targets achieved a coefficient of determination of at least 0.9987 and a normalized root-mean-square error below 0.8% on every target, while random forests and gradient boosting provided nearly equivalent accuracy at lower engineering effort, and linear regression proved inadequate for gain and bandwidth. The training and validation curves confirmed fast convergence with no divergence between the two partitions, which excludes variance-driven overfitting but does not by itself establish generalization beyond the sampled design space. Feature-importance analysis recovered textbook design sensitivities directly from the data, and the trained surrogates enabled both specification-driven inverse design and full Pareto-front extraction of the gain-bandwidth-power and gain-noise-power trade-offs at roughly five orders of magnitude the speed of circuit simulation. The datasets and the benchmark therefore constitute a transparent baseline upon which reinforcement learning, Bayesian optimization, and foundation-model-based analog design flows can build. Future work should extend the sweeps across process corners, incorporate mismatch and layout parasitic, and add linearity metrics so that the surrogates can support production-grade radio-frequency sizing.

These results are bounded by the conditions under which they were obtained, and the bounds belong in the conclusion rather than only in the limitations. The accuracy reported is interpolation accuracy inside a densely sampled, single-corner, schematic-level design space for three structurally simple topologies. It degrades when the model is asked to predict withheld regions of that space, it has not been tested against process variation, mismatch, layout parasitics, or linearity metrics, and it has not been checked against an independently generated dataset or against measured hardware. The contribution is therefore a measured and reproducible baseline, together with the datasets and protocol that make it reproducible, and not a claim that supervised regression is sufficient for production analog design. Progress beyond this point depends less on model architecture than on the cost of generating labels that span corners, layouts and topologies, and the benchmark released here is offered as the reference against which that progress can be measured.

REFERENCE

Afacan, E., Lourenço, N., Martins, R., & Dündar, G. (2021). Review: Machine learning techniques in analog/RF integrated circuit design, synthesis, layout, and test. *Integration, the VLSI Journal*, 77, 113–130. https://doi.org/10.1016/j.vlsi.2020.11.006

Barros, M. F. M., Guilherme, J. M. C., & Horta, N. C. G. (2010). Analog circuits optimization based on evolutionary computation techniques. *Integration, the VLSI Journal, 43*(1), 136–155. https://doi.org/10.1016/j.vlsi.2009.09.001

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/a:1010933404324

Budak, A. F., Bhansali, P., Liu, B., Sun, N., Pan, D. Z., & Kashyap, C. V. (2021). DNN-Opt: An RL inspired optimization for analog circuit sizing using deep neural networks. *2021 58th ACM/IEEE Design Automation Conference (DAC),* 1219–1224. https://doi.org/10.1109/dac18074.2021.9586139

Cao, W., Gao, J., Ma, T., Ma, R., Benosman, M., & Zhang, X. (2025). RoSE-Opt: Robust and efficient analog circuit parameter optimization with knowledge-infused reinforcement learning. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *44*(2), 627–640. https://doi.org/10.1109/tcad.2024.3435692

Devi, S., Tilwankar, G., & Zele, R. (2021). Automated design of analog circuits using machine learning techniques*. 2021 25th International Symposium on VLSI Design and Test (VDAT),* 1–6. https://doi.org/10.1109/vdat53777.2021.9601131

Fayazi, M., Colter, Z., Afshari, E., & Dreslinski, R. (2021). Applications of artificial intelligence on the modeling and optimization for analog and mixed-signal circuits: A review. *IEEE Transactions on Circuits and Systems I: Regular Papers*, *68*(6), 2418–2431. https://doi.org/10.1109/tcsi.2021.3065332

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics, 29*(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

Gielen, G. G. E., & Rutenbar, R. A. (2000). Computer-aided design of analog and mixed-signal integrated circuits. *Proceedings of the IEEE, 88*(12), 1825–1854. https://doi.org/10.1109/5.899053

Gilbert, B. (1968). A precise four-quadrant multiplier with subnanosecond response.

IEEE Journal of Solid-State Circuits, 3(4), 365–373. https://doi.org/10.1109/jssc.1968.1049925

Gray, P. R., Hurst, P. J., Lewis, S. H., & Meyer, R. G. (2009). *Analysis and design of analog integrated circuits* (5th ed.). Hoboken, NJ: Wiley.

Hakhamaneshi, K., Nassar, M., Phielipp, M., Abbeel, P., & Stojanović, V. (2023). Pretraining graph neural networks for few-shot analog circuit modeling and design. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 42*(7), 2163–2173. https://doi.org/10.1109/tcad.2022.3217421

Hakhamaneshi, K., Werblun, N., Abbeel, P., & Stojanović, V. (2019). BagNet: Berkeley analog generator with layout optimizer boosted with deep neural networks. *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD),* 1–8. https://doi.org/10.1109/iccad45719.2019.8942062

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks, 2*(5), 359–366. https://doi.org/10.1016/0893-6080(89)90020-8

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations* (pp. 1–15). San Diego, CA.

Lai, Y., Lee, S., Chen, G., Poddar, S., Hu, M., Pan, D. Z., & Luo, P. (2025). AnalogCoder: Analog circuit design via training-free code generation. *Proceedings of the AAAI Conference on Artificial Intelligence, 39*(1), 379–387. https://doi.org/10.1609/aaai.v39i1.32016

Lee, T. H. (2004). *The design of CMOS radio-frequency integrated circuits* (2nd ed.). Cambridge, UK: Cambridge University Press.

Li, Y., Wang, Y., Li, Y., Zhou, R., & Lin, Z. (2020). An artificial neural network assisted optimization system for analog design space exploration. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 39*(10), 2640–2653. https://doi.org/10.1109/tcad.2019.2961322

Liu, B., Gielen, G., & Fernández, F. V. (2014). *Automated design of analog and high-frequency circuits: A computational intelligence approach*. Berlin, Germany: Springer.

Lourenço, N., Afacan, E., Martins, R., Passos, F., Canelas, A., Póvoa, R., … Dündar, G. (2018). On the exploration of promising analog IC designs via artificial neural networks. *2018 15th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, 133–136. https://doi.org/10.1109/smacd.2018.8434896

Lyu, W., Yang, F., Yan, C., Zhou, D., & Zeng, X. (2018). Batch Bayesian optimization via multi-objective acquisition ensemble for automated analog circuit design. In *Proceedings of the 35th International Conference on Machine Learning* (pp. 3306–3314). Stockholm, Sweden: PMLR. https://doi.org/

Mina, R., Jabbour, C., & Sakr, G. E. (2022). A review of machine learning techniques in analog integrated circuit design automation. *Electronics, 11*(3), 435. https://doi.org/10.3390/electronics11030435

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830. https://doi.org/

Razavi, B. (2012). *RF microelectronics* (2nd ed.). Upper Saddle River, NJ: Prentice Hall.

Razavi, B. (2017). *Design of analog CMOS integrated circuits* (2nd ed.). New York, NY: McGraw-Hill.

Rutenbar, R. A., Gielen, G. G. E., & Roychowdhury, J. (2007). Hierarchical modeling, optimization, and synthesis for system-level analog and RF designs. *Proceedings of the IEEE, 95*(3), 640–669. https://doi.org/10.1109/jproc.2006.889371

Settaluri, K., Haj-Ali, A., Huang, Q., Hakhamaneshi, K., & Nikolić, B. (2020). AutoCkt: Deep reinforcement learning of analog circuit designs. 2020 Design, *Automation & Test in Europe Conference & Exhibition (DATE)*, 490–495. https://doi.org/10.23919/date48585.2020.9116200

Wang, H., Wang, K., Yang, J., Shen, L., Sun, N., Lee, H.-S., & Han, S. (2020). GCN-RL circuit designer: Transferable transistor sizing with graph neural networks and reinforcement learning. *2020 57th ACM/IEEE Design Automation Conference (DAC)*, 1–6. https://doi.org/10.1109/dac18072.2020.9218757

Wolfe, G., & Vemuri, R. (2003). Extraction and use of neural network models in automated synthesis of operational amplifiers. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 22*(2), 198–212. https://doi.org/10.1109/tcad.2002.806600

Zhang, Q.-J., Gupta, K. C., & Devabhaktuni, V. K. (2003). Artificial neural networks for RF and microwave design: From theory to practice. *IEEE Transactions on Microwave Theory and Techniques, 51*(4), 1339–1350. https://doi.org/10.1109/tmtt.2003.809179